European Journal of Operational Research 268 (2018) 486–503

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

An efficient matheuristic for offline patient-to-bed assignment problems

Rosita Guido*, Maria Carmela Groccia, Domenico Conforti

Laboratory of Decisions Engineering for Health Care Delivery, Department of Mechanical, Energy and Management Engineering, University of Calabria, Via Pietro Bucci Cubo 41 C, 87036 Arcavacata di Rende (CS), Italy

ARTICLE INFO

Article history:
Received 6 October 2017
Accepted 2 February 2018
Available online 6 February 2018

Keywords:
Combinatorial optimization
Scheduling
Patient bed assignment
Matheuristic

ABSTRACT

The bed assignment problem here addressed consists in assigning elective patients to beds by considering several requirements, such as patient clinical conditions and personal preferences, medical needs, bed availability in departments, length of stay and competing requests for beds. The rather complex combinatorial structure of the problem compels finding a solution for effective and efficient decision-making tools to support bed managers in making fast and accurate decisions. In this paper, we design and develop combinatorial optimization models for supporting the bed assignment decision-making process. Since the problem is $\mathcal{NP}$-hard, in order to solve the models efficiently, we propose and motivate a matheuristic solution framework based on a re-optimization approach. The matheuristic is implemented and tested on literature-based benchmark instances. It shows impressive computational performance for all the benchmark instances and the results improve all the best-known bounds of the state-of-the-art.

1. Introduction

In recent years, there has been an increasing interest of operational researchers in healthcare applications. Optimization and simulation models as well as innovative solution approaches have been devised and proposed with the aim to improve efficiency and quality of care in several complex decision-making processes (Brailsford & Vissers, 2011; Hulshof, Kortbeek, Boucherie, Hans, & Bakker, 2012). Operating room planning and management problems, for instance, have attracted a lot of attention (Cardoen, Demeulemeester, & Beliën, 2010; Guerriero & Guido, 2011). Usually, complex decision-making processes, like those concerning health care domains, are split into temporal and managerial levels, i.e. strategic, tactical and operational decision levels (Hulshof et al., 2012). The operational level refers to short-term decisions, and it is subdivided into offline and online decision-making. The offline situation reflects the in-advance decision making whereas the online one the real-time reactive decision-making in response to events that cannot be planned. For example, if each patient is classified as elective, urgent or in emergency, only admissions and discharges of elective patients can be planned and scheduled in advance.

Looking at the most difficult and challenging problems, the hospital care services delivery is of prominent significance. Within this context, the patient admission scheduling problem (PASP) and the patient bed assignment problem (PBAP) are decisional problems of notable interest at the operational decision level. The interest in these topics over the last decade is analyzed in Teixeira and De Oliveira (2015). The PASP is concerned with deciding which patient to admit and at what time (Barz & Rajaram, 2015). The PBAP, which is a sub-task of the PASP, aims to assign patients to suitable beds by taking into account of medical constraints, patient needs, and obviously bed availability. This specific problem has received minimal attention in the literature, as reported in Bachouch, Guinet, and Hajri-Gabouj (2012), Thomas et al. (2013), and usually it has been addressed only as a bed capacity problem (Hulshof et al., 2012). In the following, we first briefly review some papers that consider both the PASP and the PBAP; then we present an accurate literature review on more complex PBAPs.

1.1. Patient admission and bed scheduling problems

Bekker and Koeleman (2011) present quantitative methods to determine admission quota for scheduled admissions and analyze how variability in scheduled admissions influences on required bed capacity. Adan and Vissers (2002) develop an integer linear programming (ILP) model and test it in a pilot setting of specialty orthopaedics: the authors define an admission profile by taking into account throughput and utilization of resources while

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

https://doi.org/10.1016/j.ejor.2018.02.007

satisfying some given restrictions. Conforti, Guerriero, Guido, Cerinic, and Conforti (2011) formulate an ILP model integrating patients admission and resource scheduling to improve clinical efficiency in a hospital rheumatology department: the main goal is to select elective patients, schedule their required medical equipment/devices and define their shorter length of stay (LOS) because of scarcity of beds. Lei, Na, Xin, and Fan (2014) propose a mixed ILP model with constraints on surgery capacity for tackling deterministic and stochastic LOS in the PASP and the PBAP. The scheduling result includes admitted or rejected patients, admission day, last hospitalization day and assigned bed number (all beds are identical). In these reviewed papers, beds are a limited hospital resource and the PBAP is reduced to bed capacity constraints as well as in other papers. Only a few papers consider medical requirements, patients' room preference, bed equipment, hospital policies constraints in the formulated PBAP. Schmidt, Geisler, and Spreckelsen (2013) define a mathematical program based on up-to-date patients' LOS and take into account of bed capacity and patients' room preference (e.g., single, double); a dummy ward with high capacity is introduced to avoid infeasible solutions but its usage is penalized and interpreted as a dismissal. More complex is the decision support tool based on an ILP model of Bachouch et al. (2012), which aims to find available beds for elective and acute cases by considering patients' expected LOS and several constraints (i.e., incompatibility between pathologies, no mixed-sex rooms, continuity of care, and contagious patients): if there is no available bed for an elective patient, the physician can change his/her admission day; if a there is no available bed for acute patient, he/she is transferred to another hospital or kept in the emergency ward while waiting for a bed. Mazier, Xie, and Sarazin (2010) tackle the emergency patient waiting time reduction problem in a real-time planning: the aim is to assign patients to hospital rooms without disruption in inpatient stay. Patients already assigned could be moved into other rooms or units but transfers are minimized. Thomas et al. (2013) propose a decision support system to manage the PASP and the PBAP in real-time and flexible way at the same time: a user can specify whether bed attributes, room-type requirements, gender mismatch requirements should be hard or soft constraints; hard constraints are on isolation requirements and capacity constraints, i.e., at most one patient is assigned to one bed and vice versa.

In the following, we review more complex PBAPs. In particular, we focus on the PBAP formalized in Demeester, Souffriau, De Causmaecker, and Vanden Berghe (2010).

1.2. Offline and online patient bed scheduling problems

Demeester et al. (2010) present a PBAP similar to that addressed in Thomas et al. (2013) but formalized as an offline challenging problem, which hospitals with a central planning unit have to manage. This combinatorial optimization problem aims to find optimal bed assignments for elective patients by considering hospital departments, specialties, bed equipment, quality of care and patient characteristics such as pathology, mandatory bed equipment, preferred bed equipment, gender, age and room preferences. Vancroonenburg, Della Croce, Goossens, and Spieksma (2014) demonstrate that the PBAP defined in Demeester et al. (2010) is $\mathcal{NP}$-hard. The best patient bed assignment, which consists in matching patients' characteristics and room characteristics among all possible alternatives, becomes difficult to determine and hard to solve manually. Hence, the interest in developing efficient quantitative approaches to support hospital admission administrators. The ILP model formulated in Demeester et al. (2010) assigns patients to beds and is a soft version of the original problem because violations on mandatory bed equipment, age policy and gender policy, which are hard constraints in the PBAP definition, are penalized. Transfers in and out of rooms are allowed but highly penalized. The authors constructed six PBAP instances, related to Belgian hospitals, and developed a hybrid tabu-search method for solving them because of the high computational time with common solvers (in some cases more than 24 hours for finding a feasible solution). Bilgin, Demeester, Misir, Vancroonenburg, and Vanden Berghe (2012) in their follow up paper design several heuristic approaches for improving the results on the above six benchmark instances and report that no optimal solution was found after a week of computations with integer programming approaches. They provide also seven new and more complex benchmark instances. The contribution of Ceschia and Schaerf (2011) to this problem is notable because they reduce the number of decision variables by formulating the PBAP as a patient-to-room assignment problem, compute different kinds of lower bounds by removing some constraints, and improve considerably the best-known upper bound values by a simulated annealing approach. These new best values are found by solving the same soft version of the original PBAP formulated in Demeester et al. (2010) and allowing at most only one transfer. Misir, Verbeeck, Causmaecker, and Berghe (2013) analyze several hyper-heuristics with varying characteristics on the first six instances and their best results are similar or worse than those in Ceschia and Schaerf (2011). Range, Lusby, and Larsen (2014) tackle the problem with a Dantzig–Wolfe decomposition in a set-partitioning problem as master problem, and a set of room scheduling problems as pricing problem. They are the first authors who present computational results on the original PBAP (i.e., hard constraints on mandatory bed equipment, age and gender policies) and improve some of the known upper bounds but only of the small sized benchmark instances because of the increased complexity and size of the model with the larger instances. As reported in the literature, some of the benchmark instances require more than 24 hours for finding a feasible solution. Turhan and Bilgen (2017) decompose the problem in subproblems by employing time and patient decomposition approaches, and the feasible solutions, found in low computational times, have a gap from the best-known values in the range 15–21%.

Successive developments on PBAP are due to Ceschia and Schaerf (2012) and Vancroonenburg, De Causmaecker, and Vanden Berghe (2016) who extend the original PBAP formulation to an online PASP to manage some dynamic aspects such as arrival of urgent patients and uncertainty of LOS. Patients admission and discharge dates are not fixed a priori but they have to be defined in order to improve bed utilization and minimize penalty costs. Both papers propose two-index patient/room formulations. Vancroonenburg et al. (2016) test their models on four benchmark instances of the PBAP by adding a random registration date and an expected departure date; Ceschia and Schaerf (2012) propose a simplified optimization model formulation for the PASP under uncertainty, that they named PASU, introduced a binary matrix for feasible/infeasible assignments, and provided benchmark instances for PASU. Lusby, Schwierz, Range, and Larsen (2016) solve these instances by an efficient and flexible approach based on a simulated annealing framework and an adaptive large neighborhood search procedure. They report improvements in the range of 3–14% for the large instances; improvements in schedule quality for the short family instances are found by setting suitable penalty values in Guido, Solina, and Conforti (2017). Finally, a more elaborate model for a PASP and PBAP with a flexible planning horizon and operating room resources and a solution approach, which explores the search space using a composite neighborhood search algorithm, are proposed in Ceschia and Schaerf (2016).

In Table 1, we summarize the main characteristics of the several PBAPs addressed in the reviewed papers, i.e., online PBAP, offline PBAP and whether constraints on gender policies, mandatory equipment, preferred equipment, age policies, required unit are only hard, only soft, or both hard and soft. We also detail if there

Table 1
Overview of the PBAP addressed in the reviewed papers.

Paper	PBAP		PASP	Static	Dyn.	Dummy	Constraints				
	Online	Offline		LOS	LOS	room/refused	Gender policies	Man. equip.	Pref. equip.	Age policies	Req unit
Bachouch et al. (2012)	✓	✓	✓	✓		✓	Hard				Hard
Thomas et al. (2013)	✓		✓			✓	Hard/soft	Hard/soft			Hard/soft
Lei et al. (2014)	✓		✓	✓	✓	✓					
Schmidt et al. (2013)	✓		✓		✓	✓					
Mazier et al. (2010)	✓		✓	✓		✓	Hard				Soft
Ceschia and Schaerf (2012)	✓		✓	✓	✓		Soft	Hard	Soft	Hard	Hard/soft
Vancroonenburg et al. (2016)	✓		✓		✓	✓	Soft	Soft	Soft	Soft	Soft
Ceschia and Schaerf (2016)	✓		✓	✓	✓		Soft	Hard	Soft	Hard	Hard/soft
Vancroonenburg et al. (2013)		✓	✓	✓			Soft	Soft	Soft	Soft	Soft
Demeester et al. (2010)		✓		✓			Soft	Soft	Soft	Soft	Soft
Bilgin et al. (2012)		✓		✓			Soft	Soft	Soft	Soft	Soft
Ceschia and Schaerf (2011)		✓		✓			Soft	Soft	Soft	Soft	Soft
Misir et al. (2013)		✓		✓			Soft	Soft	Soft	Soft	Soft
Range et al. (2014)		✓		✓			Hard	Hard	Soft	Hard	Soft
Turhan and Bilgen (2017)		✓		✓			Soft	Soft	Soft	Soft	Soft
This paper		✓		✓			Hard/soft	Hard/soft	Soft	Hard/soft	Soft

is a solved PASP, static LOS, dynamic LOS, and a dummy room. The check and hyphen symbols denote the presence and absence, respectively, of a given characteristic.

The last six papers address the problem defined in Demeester et al. (2010) even though they refer to it as PASP, and present results on the benchmark instances.

1.3. Outline of the paper

The contribution of this study is fourfold: (1) we address operational bed management challenges of the PBAP defined in Demeester et al. (2010) by developing specific ILP models, (2) owing to the multiobjective nature of the problem, we give some tips about how to set suitable penalty values, (3) we design a matheuristic solution approach based on solving a sequence of hierarchical optimization subproblems, (4) we test our approach on the PBAP benchmarks and improve all the best-known bounds. Some optimal value are also found.

The paper is organized as follows: Section 2 presents the PBAP definition, its main features, and our ILP models; Section 3 discusses the main differences between our optimization models and those reported in the literature and provides useful tips to set penalty values; Section 4 presents a short review on heuristic and hybrid approaches relevant to our solution approach; then we introduce our matheuristic approach, which is a combination of fix and optimize heuristic, neighborhood searches, and ILP solvers. We evaluate it on the benchmark datasets of PBAP by carrying out computational experiments based on the default penalty values and on suitable setting values. Results are reported and compared with those of the literature in Section 5; conclusions are drawn and future directions outlined in Section 6.

2. Problem definition and mathematical models

In this section, we present the model developments of the PBAP. For better understanding, we report the main features and the used notation.

Hospital rooms are available for elective patients in different departments, where there are medical specialties with different levels of expertise in treating patients. Hospital rooms differ according to the department where they are located, gender policy, age policy, number of beds, and bed equipment, which can be mandatory and/or desired for patients. Three gender policies are distinguished: (1) rooms reserved only for men, (2) rooms only for women, and (3) the dependent gender policy (DGP), that is, only patients of the same gender as those who already occupy a room can be assigned. Each patient has proper characteristics and has to be assigned to the most appropriate bed/room/department over his/her stay. His/her admission and discharge dates have been already planned and patient's LOS does not change.

Elective patients have to be assigned to specific beds in specific wards and there is a penalty cost whenever a patient is assigned to a room that differs from a suitable one. Transfers between rooms are penalized because of extra work for a staff and discomfort for patients (Demeester et al., 2010). The hard constraints on the DGP add complexity and make the PBAP $\mathcal{NP}$-hard, as demonstrated in Vancroonenburg et al. (2014): the gender defined by patients occupying a room changes dynamically and a patient assignment affects resulting schedules. We define departments, specialties, rooms and beds as resources; patients with specific characteristics as demand. These are described in the following, and Table 2 summarizes the notation.

2.1. Resource

Departments and specialties : a hospital has a given number of departments. Each department has a given number of specialties and different levels of expertise in treating pathologies. A department should have age restrictions (e.g. paediatric and geriatric departments). Let Dep be the set of departments and $\mathcal{S}$ be the set of specialties. Every department is highly qualified to treat some diseases: let Exp be the set of levels of expertise and $exp_{ds} \in Exp$ be the level of expertise of department $d \in Dep$ in treating diseases of specialty $s \in \mathcal{S}$.

Rooms and beds : let $\mathcal{R}$, indexed by r, be the set of all available rooms. Every room has a number of beds; C_r denotes the capacity of room $r \in R$. Room capacities (e.g., single, double) define the set of room categories $\mathcal{RC}$. Some beds have equipment (e.g. oxygen) and, without loss of generality, it is assumed that beds in a room have the same equipment. Let E be the set of several bed equipment, indexed by e; $eq_{re} = 1$ if bed equipment $e \in E$ is in room $r \in \mathcal{R}$, 0 otherwise. $\mathcal{R}_1 \subseteq \mathcal{R}$ denotes the subset of rooms with at least one type of bed equipment, and $\mathcal{R}_2 = \mathcal{R} \setminus \mathcal{R}_1$ the subset those without bed equipment. Rooms have a ranked list of specialties for which they are suitable; let RS be the set of levels of room suitability. Rooms can be subject to a gender policy: some rooms can be restricted to only male or female patients (restricted gender policy, RGP); other rooms are characterized by the DGP, that is only patients with the same gender of patients occupying a room can be assigned to. Rooms without a gender policy can be assigned both

Table 2
Sets and parameters.

$\mathcal{D}$	Planning horizon in days indexed by d
Hospital resources	
Dep	Departments
S	Specialties
Exp	Set of levels of expertise in treating a disease
$\mathcal{R}$	Rooms
$\mathcal{R}_1$	Rooms with at least one type of bed equipment
$\mathcal{R}_2$	Rooms without bed equipment
$\mathcal{RC}$	Set of room categories
$\mathcal{RS}$	Set of levels of suitability of a room for a given specialty
$E = \{1, \ldots, m\}$	Set of types of bed equipment
$\mathcal{GP} = \{1, \ldots, 4\}$	Set of gender policies
$\mathcal{AP} = \{1, \ldots, n\}$	Set of age policies
For all rooms $r \in \mathcal{R}$:	
$C_r \in \mathcal{RC}$	Room capacity
eq_{re}	$= 1$ if the bed equipment $e \in E$ is in room r, 0 otherwise
gp_r	$= k$, i.e., the gender policy $k \in \mathcal{GP}$ is applied to room r
ap_{rj}	$= 1$ if the age policy $j \in \mathcal{AP}$ is applied to room r, 0 otherwise
Patients	
$\mathcal{P}_F$	Set of female patients
$\mathcal{P}_M$	Set of male patients
$\mathcal{P} = \mathcal{P}_F \bigcup \mathcal{P}_M$	Set of all patients
$\mathcal{P}_1$	Set of patients with at least one mandatory bed equipment
$\mathcal{P}_2 = \mathcal{P} \setminus \mathcal{P}_1$	Set of patients without mandatory bed equipment
For all patients $p \in \mathcal{P}$:	
$sp_p \in S$	Specialty of patient
a_p	Admission day
d_p	Discharge day
$\mathcal{D}_p = \{a_p, \ldots, d_p - 1\}$	Stay as a set of consecutive nights
$L_p = d_p - a_p$	Length of stay (in nights)
$rc_p \in \mathcal{RC}$	Preferred room category
$gen_p \in \{1, 2\}$	Patient gender
a_{pj}	$= 1$ if the age of patient p is consistent with the age policy $j \in \mathcal{AP}$, 0 otherwise
me_{pe}	$= 1$ if bed equipment $e \in E$ is mandatory, 0 otherwise
pr_{pe}	$= 1$ if bed equipment $e \in E$ is preferred, 0 otherwise

to women and men at the same time. More formally, we define the set of gender policies as $\mathcal{GP} = \{1, 2, 3, 4\}$, where value 1 and 2 denote the RGP to only male and female patients, respectively; value 3 denotes the DGP, and value 4 means no gender policy. For each room $r \in \mathcal{R}, gp_r = k, k \in \mathcal{GP}$ defines the adopted gender policy. Moreover, some rooms are subject to age policies (e.g. paediatrics, gerontology): let $\mathcal{AP}$ be the set of age policies.

2.2. Demand

Patients : let $\mathcal{P}$ be the set of elective patients, indexed by p, that have to be assigned on a defined planning period $\mathcal{D}$ to hospital rooms $\mathcal{R}$. Patients are characterized by an admission date and a discharge date, gender, age, specialty, and health status, which defines his/her mandatory bed equipment and/or preferred bed equipment. Admission dates and discharge dates, denoted by a_p and d_p, respectively, are known for all patients $p \in \mathcal{P}$. The patient stay is given as consecutive nights between admission and discharge date and it is denoted by $\mathcal{D}_p = \{a_p, \ldots, d_p - 1\}$. The length of stay is $L_p = d_p - a_p$. Patient's specialty is denoted by $sp_p \in \mathcal{S}$, and gender by $gen_p = 1$ for men and $gen_p = 2$ for women. Let $\mathcal{P}_F \subseteq \mathcal{P}$ and $\mathcal{P}_M \subseteq \mathcal{P}$ be the sets of women and men, respectively. Some patients' pathologies may require specific bed equipment: $me_{pe} = 1$ if bed equipment $e \in E$ is mandatory, 0 otherwise. $\mathcal{P}_1 \subseteq \mathcal{P}$ is the set of patients with mandatory bed equipment and $\mathcal{P}_2 = \mathcal{P} \setminus \mathcal{P}_1$ is the set of remaining patients. If a bed equipment is only preferred, the parameter $pr_{pe} = 1$, otherwise $pr_{pe} = 0$. Patients may also express some preferences concerning the room category they wish to stay in. This preference is denoted by $rc_p \in \mathcal{RC}$.

To reduce the search space of the problem, we solve the patients-to-rooms assignment problem firstly and then the patient-bed assignment problem. This approach has been already used in Ceschia and Schaerf (2011). In what follows, we refer thus to room equipment instead of bed equipment because of the assumption that beds in a room are identical. We formulate three ILP models. The first patient-room assignment model formulation has the original hard constraints on mandatory equipment, age and gender policies. Some of these constraints are relaxed to avoid infeasibility and this is the second model. We refer to these two models as HM_{PBA} and SM_{PBA}, respectively, to distinguish that they are the hard and the soft version of the PBAP. The third optimization model is for patients with co-morbidity and overall stay as consecutive periods. We refer to this model as $SM\text{-}CLOS_{PBA}$. If a patient has co-morbidity and has to be treated by multiple specialties, two cases can occur:

Co-mor (1) There is no inconsistency for required equipment, departments and levels of expertise in treating patient. Multiple specialties can be thus reduced to only one for scheduling purposes and patient' LOS is the maximum LOS among those defined for each specialty if they differ;

Co-mor (2) There is inconsistency among specialties' requirements. Two cases are distinguishable: Co-mor (2.a), that is the overall patient LOS is given as consecutive single LOS defined by involved specialties. A suitable bed should be assigned in each period by satisfying specific requirements with a minimum number of transfers between rooms; Co-mor (2.b), that is co-morbidity illnesses have to be treated in parallel and the specific requirements should be ranked both to ensure the best possible level of expertise and assign mandatory and preferred equipment.

Co-mor (1) and Co-mor (2.b) can be tackled by HM_{PBA} and SM_{PBA}; Co-mor (2.a) only by $SM\text{-}CLOS_{PBA}$.

Five hard constraints characterize the original PBAP:

1. every patient has to be assigned to a bed over his/her stay
2. patient mandatory equipment requirement has to be satisfied by assigning him/her to a well-equipped room
3. patients have to be assigned consistently to rooms with defined age policies
4. each patient has to be assigned to a room such that the defined gender policy (restricted or dependent) is satisfied
5. room capacity constraint.

Every patient should be assigned to a suitable room in correspondence with his/her characteristics: e.g., a department with high level of expertise in treating patient's pathology, preferred equipment, and preference on room category; penalty values are defined in order to limit violations of these constraints. A patient could change room every day and transfers between rooms are allowed but highly penalized to reduce patient's discomfort. The PBAP of Demeester et al. (2010) can be stated as follows: find the best feasible assignment of all patients $p \in \mathcal{P}$ to rooms $r \in \mathcal{R}$ at minimum penalty cost. In this paper, each patient is assigned to only one room during his/her LOS and transfers are allowed only in a particular case of co-morbidity.

Table 3
Constraint violations.

v_1	Violations on a patient not being assigned to a room of the correct specialty: a patient should be assigned to a room that matches his/her specialty otherwise the assignment is penalized by w^1_{pr}
v_2	Violations on patient not being assigned to a room with preferred equipment: patients should be assigned to a room with preferred equipment otherwise every missing equipment is penalized. The overall penalty is w^2_{pr}
v_3	Violations on patient not being assigned to a department with the right specialty: a patient should be placed in a department with high level of expertise according to the specific pathology of the patient otherwise the penalty is w^3_{pr}
v_4	Violations on room category preference: every patient has a preferred room category where he/she would stay. If the capacity of the assigned room is greater than that preferred there is a penalty w^4_{pr}

2.3. An optimization model for the patient bed assignment problem

Before describing the optimization model, let us introduce the following additional notation.

Constraint violations, reported in Table 3, are penalized per night by the corresponding penalty cost.

Violations v_1, v_2, v_3 are on quality of medical care; v_4 is on preferred room category. The overall penalty assignment w_{pr} is computed for each couple patient-room as sum of the above four violations, $w_{pr} = w^1_{pr} + w^2_{pr} + w^3_{pr} + w^4_{pr}$.

The decision variables are $x_{pr} = 1$ if patient $p \in \mathcal{P}$ is assigned to room $r \in \mathcal{R}$, 0 otherwise; $f_{rd} = 1$ and $m_{rd} = 1$ if at least one woman and one man, respectively, is in room $r \in \mathcal{R} \mid gp_r = 3$ on day $d \in \mathcal{D}$, 0 otherwise. Recall that $gp_r = 3$ denotes the dependent gender policy.

The binary integer programming model is the following.

$$\min \sum_{p\in\mathcal{P}} \sum_{r\in\mathcal{R}} w_{pr} x_{pr} L_p \tag{1}$$

$$\sum_{r\in\mathcal{R}_1} x_{pr} = 1 \qquad \forall p \in \mathcal{P}_1 \tag{2}$$

$$\sum_{r\in\mathcal{R}} x_{pr} = 1 \qquad \forall p \in \mathcal{P}_2 \tag{3}$$

$$x_{pr} me_{pe} \le eq_{re} \qquad \forall p \in \mathcal{P}_1, \forall r \in \mathcal{R}_1, \forall e \in E \tag{4}$$

$$x_{pr} \le a_{pj} \qquad \forall p \in \mathcal{P}, \forall j \in \mathcal{AP}, \forall r \in \mathcal{R} \mid ap_{rj} = 1 \tag{5}$$

$$x_{pr} = 0 \qquad \forall p \in \mathcal{P}_{\mathcal{F}}, \forall r \in \mathcal{R} \mid gp_r = 1 \tag{6}$$

$$x_{pr} = 0 \qquad \forall p \in \mathcal{P}_{\mathcal{M}}, \forall r \in \mathcal{R} \mid gp_r = 2 \tag{7}$$

$$f_{rd} \ge x_{pr} \qquad \forall p \in \mathcal{P}_{\mathcal{F}}, \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D}_p \tag{8}$$

$$m_{rd} \ge x_{pr} \qquad \forall p \in \mathcal{P}_{\mathcal{M}}, \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D}_p \tag{9}$$

$$f_{rd} + m_{rd} \le 1 \qquad \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D} \tag{10}$$

$$cap_{rd} = C_r - \sum_{p\in\mathcal{P} \mid d\in\mathcal{D}_p} x_{pr} \qquad \forall r \in \mathcal{R}, \forall d \in \mathcal{D} \tag{11}$$

$$cap_{rd} \ge 0 \qquad \forall r \in \mathcal{R}, \forall d \in \mathcal{D} \tag{12}$$

$$x_{pr} \in \{0, 1\} \qquad \forall p \in \mathcal{P}, \forall r \in \mathcal{R} \tag{13}$$

$$m_{rd}, f_{rd} \in \{0, 1\} \qquad \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D} \tag{14}$$

The objective function (1) aims to assign patients to rooms according to high quality of care and patient preferences. The penalty room assignment w_{pr} is based on the compatibility with specialty of the department where room r is located, preferred equipment, and patient preference of room category. For each patient, the overall penalty assignment is given as w_{pr}, computed per night, multiplied by the number of nights (i.e, patient LOS). Constraints (2) and (3) define that only one room is assigned to a patient. Constraints (2) ensure that patients characterized by mandatory equipment are assigned only to equipped rooms (subset $\mathcal{R}_1$), and these rooms are equipped at least with that is mandatory for them, i.e., Constraints (4). Constraints (5) express that only patients satisfying age policy j can be assigned to rooms with that policy. Constraints (6)–(10) are on gender policy: Constraints (6) and (7) do not allow to assign patients to rooms not matching the specific RGP, i.e., no women in rooms reserved for men and no men in rooms reserved for women, respectively; Constraints (8) and (9) capture the gender of patients assigned to a room subject to the DGP, i.e., only men or women are allotted at the same time, and Constraints (10) aim to satisfy the DGP. Constraints (11) and (12) are on room capacity, that is, the number of patients assigned to room must not be greater than the number of available beds on each day. Finally, Constraints (13) and (14) are integrity relations. We refer to this model formulation as HM_{PBA}.

In order to make the formulation stronger and thus speed up the solution approach, the constraints $x_{pr} = 0, \forall p \in \mathcal{P}_2, r \in \mathcal{R}_2$ can be added.

In case of sparse assignments it could be convenient to reduce the number of variables of HM_{PBA} by defining a set of feasible room assignments per each patient. Let $RE_r \subseteq E$ be the set of equipment available in room r, and $ME_p \subseteq E$ be the set of mandatory equipment for patient p. For each $p \in \mathcal{P}$, we define $\bar{R}^1_p = \{r \in R | gp_r \le 2 : ME_p \subseteq RE_r, ap_{rj} a_{pj} > 0, gp_r = gen_p\}$ as the set of feasible rooms subject to RGP, and $\bar{R}^2_p = \{r \in R | gp_r \ge 3 : ME_p \subseteq RE_r, ap_{rj} a_{pj} > 0\}$ as the set of feasible rooms subject to DGP or without gender policy. The set of feasible room assignments is thus $\bar{R}_p = \bar{R}^1_p \bigcup \bar{R}^2_p$, and a sparse version of HM_{PBA} is:

$$\min \sum_{p\in\mathcal{P}} \sum_{r\in\bar{\mathcal{R}}_p} w_{pr} x_{pr} L_p \tag{15}$$

$$\sum_{r\in\bar{\mathcal{R}}_p} x_{pr} = 1 \qquad \forall p \in \mathcal{P} \tag{16}$$

$$cap_{rd} = C_r - \sum_{p\in\mathcal{P} \mid d\in\mathcal{D}_p, r\in\bar{\mathcal{R}}_p} x_{pr} \qquad \forall r \in \mathcal{R}, \forall d \in \mathcal{D} \tag{17}$$

$$x_{pr} \in \{0, 1\} \qquad \forall p \in \mathcal{P}, \forall r \in \bar{\mathcal{R}}_p \tag{18}$$

Constraints (8)–(10), (12), (14)

2.4. A patient bed assignment model with soft constraints

It could happen that demand for mandatory equipment and/or rooms with a defined gender policy or/and an age policy is greater than availability and this causes infeasibility. The constraints on mandatory equipment, gender policy and age policy can be relaxed but their violation has a penalty cost. The following violations are allowed in the second model:

v_5: violations on no matched mandatory equipment, penalized by w_m

v_6: violations on the dependent gender policy, penalized by w_g

v_7: violations on age policies, penalized by w_a

New binary decision variables are introduced. Their value is 1 when the corresponding hard constraint is violated, 0 otherwise: $vm_{pe} = 1$ if patient $p \in \mathcal{P}$ is assigned to a room without mandatory equipment $e \in E$; $va_{pj} = 1$ if the age of patient $p \in \mathcal{P}$ is not consistent with age policy j, $j \in \mathcal{AP}$; $b_{rd} = 1$ if both women and men are in room $r \in R \mid gp_r = 3$ on day $d \in \mathcal{D}$.

The soft version of HM_{PBA} can be defined as the following model, which is referred as SM_{PBA}:

$$\min F_p + F_m + F_a + F_g \tag{19}$$

$$\sum_{r \in \mathcal{R}} x_{pr} = 1 \qquad \forall p \in \mathcal{P} \tag{20}$$

$$x_{pr} m e_{pe} \le eq_{re} + vm_{pe} \qquad \forall p \in \mathcal{P}_1, \forall r \in \mathcal{R}, \forall e \in E \tag{21}$$

$$x_{pr} \le a_{pj} + va_{pj} \qquad \forall p \in \mathcal{P}, \forall j \in \mathcal{AP}, \forall r \in \mathcal{R} \tag{22}$$

$$b_{rd} \ge m_{rd} + f_{rd} - 1 \qquad \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D} \tag{23}$$

$$b_{rd} \in \{0, 1\} \qquad \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D} \tag{24}$$

$$va_{pj} \in \{0, 1\} \qquad \forall p \in \mathcal{P}, \forall j \in \mathcal{AP} \tag{25}$$

$$vm_{pe} \in \{0, 1\} \qquad \forall p \in \mathcal{P}, \forall e \in E \tag{26}$$

Constraints (6)–(9) and (11)–(14)

The objective function (19) has four components:

$F_p = \sum_{p \in \mathcal{P}} \sum_{r \in \mathcal{R}} w_{pr} x_{pr} L_p$ overall penalty owing to violations on quality of care and patient preferences

$F_m = w_m \sum_{p \in \mathcal{P}} \sum_{e \in E} vm_{pe} L_p$ overall penalty owing to violations on mandatory equipment

$F_a = w_a \sum_{p \in \mathcal{P}} \sum_{j \in \mathcal{AP}} va_{pj} L_p$ overall penalty owing to violations on age policies

$F_g = w_g \sum_{r \in \mathcal{R} | gp_r = 3} \sum_{d \in \mathcal{D}} b_{rd}$ overall penalty owing to violations of the dependent gender policy.

Constraints (20) ensure that only one room is assigned to a patient. Constraints (21) state that an assigned room could not be suitable because of mandatory equipment for patients in $\mathcal{P}_1$. Constraints (22) express that the age of a patient could not be consistent with the age policy defined in the assigned room. Constraints (23) capture the presence of women and men at the same time in rooms with DGP. Constraints (24)–(26) are integrity relations.

Note that only violations on the DGP are allowed in this optimization model. If there is infeasibility due to the hard constraints on RGP, Constraints (6) and (7) have to be removed, and Constraints (8), (9), Constraints (23) and the term F_g defined for all rooms with gender policies (i.e., for $gp_r \neq 4$ because every gender policy could be violated and then penalized accordingly).

2.5. Patients with composite length of stay

As already introduced, the third optimization model addresses patients with Co-mor (2.a). Without loss of generality, we assume that some patients with co-morbidity need only two medical specialties. Let P' be the set of these patients. Each patient has to be treated by a first specialty from day a_p to day d_p, and by a second specialty from day $a'_p = d_p$ to his/her discharge day d'_p. The overall stay is composite by two consecutive periods: $a_p, \ldots, d_p$ and $a'_p, \ldots, d'_p$ where $d_p = a'_p$. The second stay (in nights) is $\mathcal{D}'_p = \{a'_p, \ldots, d'_p - 1\}$ and the second LOS is $L'_p = d'_p - a'_p, \forall p \in \mathcal{P}'$. A patient $p \in P'$ is assigned to a room during his/her first LOS and can be transferred to another room on day a'_p accordingly to his/her changed characteristics of second stay (e.g., specialty, mandatory equipment). We introduce the binary decision variables $x'_{pr} = 1$ if patient $p \in P'$ is assigned to room $r \in R$ on day a'_p, and $t_{pr} = 1$ if he/she is transferred on day a'_p (otherwise $t_{pr} = 0$). The goal is to assign patients to appropriate rooms by considering all characteristics of the two stays (L_p and L'_p) and minimize transfers. The following two terms $F'_p = \sum_{p \in \mathcal{P}'} \sum_{r \in \mathcal{R}} w_{pr} x'_{pr} L'_p$ and $F_t = w_t \sum_{p \in \mathcal{P}'} \sum_{r \in \mathcal{R}} t_{pr}$, where w_t is a penalty cost for transfers, have to be minimized and added to objective function (19).

The optimization model formulation for patients with composite length of stay is constructed by adding Constraints (6)–(9), (13), (14), (20)–(26), to the following optimization model:

$$\min F_p + F_m + F_a + F_g + F'_p + F_t \tag{27}$$

$$\sum_{r \in \mathcal{R}} x'_{pr} = 1 \qquad \forall p \in \mathcal{P}' \tag{28}$$

$$t_{pr} \ge x'_{pr} - x_{pr} \qquad \forall p \in \mathcal{P}' \tag{29}$$

$$x'_{pr} m_{pe} \le eq_{re} + vm_{pe} \qquad \forall p \in \mathcal{P}_1 \cap \mathcal{P}', \forall r \in \mathcal{R}, \forall e \in E \tag{30}$$

$$x'_{pr} \le a_{pj} + va_{pj} \qquad \forall p \in \mathcal{P}', \forall j \in \mathcal{AP}, \forall r \in \mathcal{R} \tag{31}$$

$$\sum_{r \in \mathcal{R} | gp_r = 1} x'_{pr} = 0 \qquad \forall p \in \mathcal{P}_{\mathcal{F}} \cap \mathcal{P}' \tag{32}$$

$$\sum_{r \in \mathcal{R} | gp_r = 2} x'_{pr} = 0 \qquad \forall p \in \mathcal{P}_{\mathcal{M}} \cap \mathcal{P}' \tag{33}$$

$$f_{rd} \ge x'_{pr} \qquad \forall p \in \mathcal{P}_{\mathcal{F}} \cap \mathcal{P}', \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D}'_p \tag{34}$$

$$m_{rd} \ge x'_{pr} \qquad \forall p \in \mathcal{P}_{\mathcal{M}} \cap \mathcal{P}', \forall r \in \mathcal{R} \mid gp_r = 3, \forall d \in \mathcal{D}'_p \tag{35}$$

$$cap_{rd} = C_r - \sum_{p \in \mathcal{P} | d \in \mathcal{D}_p} x_{pr} - \sum_{p \in \mathcal{P}' | d \in \mathcal{D}'_p} x'_{pr} \qquad \forall r \in \mathcal{R}, \forall d \in \mathcal{D} \tag{36}$$

$$x'_{pr} \in \{0, 1\} \qquad \forall p \in \mathcal{P}', \forall r \in \mathcal{R} \tag{37}$$

$$t_{pr} \in \{0, 1\} \qquad \forall p \in \mathcal{P}', \forall r \in \mathcal{R} \tag{38}$$

We refer to this model formulation as $SM\text{-}CLOS_{PBA}$.

3. Discussion on the optimization models proposed in the literature

In this section, we underline the main differences between our optimization models and those proposed in the literature for the PBAP. As mentioned in Section 2, the ILP model of Demeester et al. (2010) assigns patients to beds and penalizes violations on all the original hard constraints. The search space is very huge because of patient-bed-day assignment variables. Transfers are allowed but highly penalized with respect to all other violations. Ceschia and Schaerf (2011) reduce considerably the search space by assigning patients to rooms and introducing room capacity constraints. These authors perform a preprocessing step aiming at computing a penalty matrix W to assess the assignments when patients' needs and preferences are not satisfied. We report, consistently with our notation, their ILP model (39)–(47), to underline the main differences with respect to our optimization models. An overall penalty cost $\bar{w}_{pr}$, due to the violations $v_1 - v_5$, v_7, and RGP, is computed for each couple patient-room. The decision variable x_{prd} has value 1 if patient p is assigned to room r on day d, 0 otherwise; t_{prd} has value 1 if patient p, assigned to room r on day d is moved to another room on day $d+1$, 0 otherwise. The binary decision variables f_{rd}, m_{rd}, b_{rd} have the same meaning above defined.

$$\min \sum_{p\in\mathcal{P}} \sum_{d\in\mathcal{D}_p} \sum_{r\in\mathcal{R}} \bar{w}_{pr} x_{prd} + \sum_{p\in\mathcal{P}} \sum_{d\in\mathcal{D}} \sum_{r\in\mathcal{R}} w_t t_{prd} + \sum_{r\in\mathcal{R}|gp_r=3} \sum_{d\in\mathcal{D}} w_g b_{rd} \tag{39}$$

$$\sum_{r\in\mathcal{R}} x_{prd} = 1 \qquad \forall p\in\mathcal{P}, \forall d\in\mathcal{D}_p \tag{40}$$

$$t_{prd} \geq x_{prd} - x_{prd+1} \qquad \forall p\in\mathcal{P}, \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{41}$$

$$f_{rd} \geq x_{prd} \qquad \forall p\in\mathcal{P}_{\mathcal{F}}, \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{42}$$

$$m_{rd} \geq x_{prd} \qquad \forall p\in\mathcal{P}_{\mathcal{M}}, \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{43}$$

$$b_{rd} \geq f_{rd} + m_{rd} - 1 \qquad \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{44}$$

$$C_r \geq \sum_{p\in\mathcal{P}} x_{prd} \qquad \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{45}$$

$$x_{prd}, t_{prd} \in \{0,1\} \qquad \forall p\in\mathcal{P}, \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{46}$$

$$b_{rd}, m_{rd}, f_{rd} \in \{0,1\} \qquad \forall r\in\mathcal{R}, \forall d\in\mathcal{D} \tag{47}$$

The objective function (39) has three terms: the first term penalizes those assignments not respecting RGP, patients' need and preference; the second one penalizes transfers, and the third term penalizes violations on the DGP. Constraints (40) ensure that each patient is assigned to only one room on every day over D_p. Constraints (41) define transfers between rooms. Constraints (42)–(44) capture both genders in a room. Constraints (45) are on room capacity, and Constraints (46) and (47) are integrity relations.

It is straightforward to observe four main differences between Model (39)–(47) and our models:

1. The decision variables x have three indices in Model (39)–(47) because they are defined over the set $\mathcal{D}_p$ for all rooms $r \in \mathcal{R}$; moreover, transfers are allowed. In our models, the overall number of variables is reduced because decision variables x are patient-to-room assignments and transfers are allowed only in $SM\text{-}CLOS_{PBA}$.
2. Model (39)–(47) is an optimization model for the PBAP with only room assignment and capacity constraints as hard; violations of mandatory equipment, age policies, and RGP are considered in $\bar{w}_{pr}$, whereas they are hard constraints in the problem statement and in our model HM_{PBA}.
3. SM_{PBA} is a soft version of the PBAP but is more flexible than Model (39)–(47) because of the variables vm_{pe} and va_{pj}, which refer to unfulfilled original hard constraints on mandatory requirements and age policy. If an instance is infeasible because of scarcity of mandatory equipment, it is possible to (i) analyze alternative schedules by setting $vm_{pe} = 0$ for some patients and equipment, (ii) relax constraints on mandatory equipment only for a subset of patients, (iii) penalize no matched equipment with different penalty costs.
4. Model (39)–(47) must be modified if cases Co-mor (2.a) there are. Indeed, the first term of the objective function must be split in several terms in order to take into account of a composite length of stay, and all constraints and variables defined appropriately.

Table 4
Hard and soft constraints in the model formulations with respect to the original problem statement.

	Mandatory equipment	Age policy	Restricted room gender policy	Dependent room gender policy
HM_{PBA}	Hard	Hard	Hard	Hard
SM_{PBA}	Soft	Soft	Hard	Soft
Model (39)–(47)	Soft	Soft	Soft	Soft

Table 4 summarizes the main differences among HM_{PBA}, SM_{PBA} and Model (39)–(47) in terms of hard and soft constraints with respect to the original problem statement. We underline that HM_{PBA} models the original PBAP and that constraints on the RGP are hard constraints in our optimization models with the aim to minimize discomforts of patients. Similar to model Model (39)–(47) is that proposed for the static PBAP in Vancroonenburg et al. (2016). Finally, as already stated in Section 1, optimization models with two-index decision variables x are developed in Ceschia and Schaerf (2012), Vancroonenburg et al. (2016). A basic model formulation is presented for PASU problem in Ceschia and Schaerf (2012) and a binary matrix is introduced for feasible assignments even if it is not explicitly defined. On the basis of the set of feasible assignments introduced in Section 2.3, this matrix has entry 1 for rooms $r \in \bar{R}_p$, 0 otherwise. Vancroonenburg et al. (2016) formulate two optimization models to address the online version of the PBAP. These models, which are distinguished in reactive and anticipatory, generate a schedule per each day of the planning horizon: the patients that have to be admitted on each day are known and they have to be assigned to the most suitable rooms by taking into account room capacity and gender policies. Only room capacity constraints are hard and they are constructed on an interval graph to generate maximal cliques. The objective function penalizes room assignments, both genders in rooms, and transfers, which occur whenever a room different from that occupied on the last night is assigned to a patient.

In the next session, we show that the optimization models have different search spaces and they are not equivalent when no suitable values are used to penalize constraint violations.

3.1. Effects of penalty values in resulting schedules

The PBAP is a multi-objective problem in nature and more complex in its soft version because a decision maker would like to optimize the several terms related to constraint violations at the same time. Lexicographic approaches (Castro-Gutierrez, Landa-Silva, & Pérez, 2010) or genetic algorithms (Guido & Conforti, 2017)

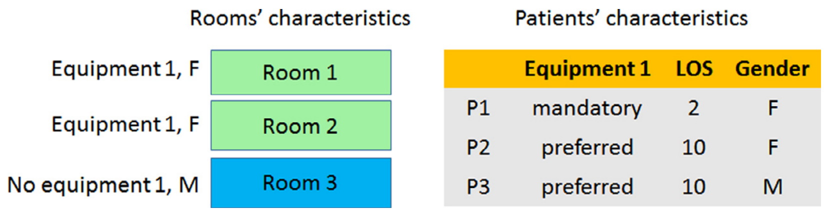


Fig. 1. Toy example. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

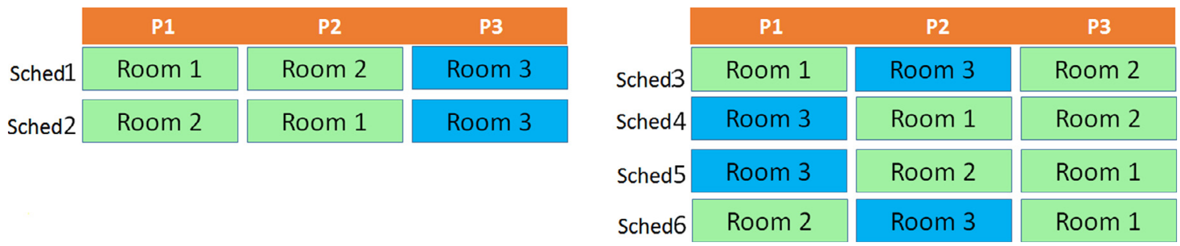


Fig. 2. The feasible patient room assignments for the toy example. On the left, the only two feasible schedules common to the three models. On the right, the schedules feasible only for Model 3.

are solution approaches that aim to optimize sequentially or simultaneously more than one objective. As described before, the PBAP is converted to a single-objective. In particular, both the objective function of SM_{PBAP} and $SM-CLOS_{PBAP}$ minimize a weighted sum of the violated hard constraints, and the penalty values should be carefully chosen.

We show the main differences among HM_{PBA}, SM_{PBA} and Model (39)–(47) in terms of search space and also of optimal solutions if no suitable penalty values are used. We refer to them as $Model_i, i = 1, 2, 3,$ respectively. Let S_i be the search space of $Model_i$. The relationship $S_1 \subseteq S_2 \subset S_3$ holds among the three search spaces. Indeed, all patient-room couples define the search space S_3 whereas the search space S_1 is more restricted than S_2 because of the hard constraints; then S_2 is more restricted than S_3 because of RGP constraints. This is illustrated in the following toy example characterized by only rooms subject to the RGP because the DGP adds complexity. This simple instance consists of three single rooms (r_1, r_2, r_3) located in a given department: the first and second room have the same bed equipment and are restricted to women. There is no bed equipment in the third room, which is restricted to men. Three patients have to be assigned to these rooms; they have the same admission date and differ only in mandatory and preferred bed equipment, LOS, and gender. All these characteristics are shown in Fig. 1, where the two rooms reserved for women are in green, and that reserved for men is in blue. Observe that

- Model 1 has only two feasible schedules: patient p_1 can be assigned only to room r_1 and r_2 because of mandatory equipment; patient p_2 to the remaining room reserved for women. Patient p_3 is assigned to room r_3 in both cases.
- Model 2 has only two feasible schedules, which are the same schedules of Model 1, because of hard constraints on the RGP.
- Model 3 has only six feasible schedules if transfers are not allowed. The number of feasible schedules increases if transfers are allowed because room assignments could change every day. This model allows violations on the RGP and mandatory equipment.

All the feasible schedules, represented as patient-to-room assignments, are reported in Fig. 2: the two schedules on the left are common to the three search spaces whereas the four schedules on the right belong only to S_3. Thus $S_1 = S_2$, $|S_1| = 2$ and $|S_3| = 6$. Another important fact is that a resulting optimal schedule strongly depends on penalty values. We show that the three models have different optimal schedules if we set $w_m = 5$ and $w^2_{pr} = 2$, which penalizes the no matched mandatory equipment and preferred equipment, respectively. These values are the default ones defined in Demeester et al. (2010). A resulting optimal schedule is dependent by the used model, that is:

- the optimal schedule of Models 1 and 2 is one of the two feasible schedules reported on the left of Fig. 2. Only the constraint on preferred equipment is not met for patient p_3, then the two schedules are equivalent in terms of objective function value, which is $w^2_{33} x_{33} LOS_3 = 2 * 10 = 20$.
- the optimal schedule of Model 3 assigns patients p_2 and p_3 to equipped rooms, and p_1 to the non-equipped room r_3 although p_1 is the only one requiring mandatory equipment. The fourth and fifth schedules reported in Fig. 2 are both optimal for Model 3 and the objective function value is $w_m x_{13} LOS_1 = 5 * 2 = 10$.

The toy instance has then an optimal schedule with objective function value equal to 10 if Model 3 is solved but this schedule is not feasible for Models 1 and 2; the objective function value obtained with Models 1 and 2 is 20. An optimization model with the original hard constraints will naturally yield larger objective values both in the linear programming-relaxation as well as the integer solutions in cases like the one illustrated above, but this is not correct from an optimization point of view. Indeed, there must be no gain to violate constraints. An appropriate setting of penalty values related to constraints violations is not trivial but HM_{PBA} overcomes this drawback. In what follows, we provide tips on how to set the penalty values for SM_{PBA}, and in Section 6 we show how they affect solution quality by comparing the schedules obtained with the default penalty values and suitable penalty values.

3.2. How to set suitable penalty costs

Suitable values used to penalize violations of constraints on mandatory equipment, gender, and age policies, are crucial for a correct solution. To better model the real-life problem in cases of violations of hard constraints, we provide tips regarding how to set penalty costs. To avoid an unexpected solution like that just illustrated, the penalty weights for mandatory equipment should be set as $\hat{w}_m > \frac{\max_{p \in \mathcal{P}, r \in \mathcal{R}} w^2_{pr} L_p}{\min_{p \in \mathcal{P}_1} L_p}$. The numerator is the biggest overall cost computed for preferred equipment; the denominator takes into account of the smallest length of stay among patients requiring equipment in $\mathcal{P}_1$. The so defined $\hat{w}_m$ ensures that well-equipped rooms are assigned to patients $p \in \mathcal{P}_1$. If $\min_{p \in \mathcal{P}_1} L_p = 1$, thus $\hat{w}_m > \max_{p \in \mathcal{P}, r \in \mathcal{R}} w_{pr}^2 L_p$.

Concerning the value of the penalty cost w_g, two exclusionary situations there are:

(a) $w_g > \hat{w}_m$ if hospital strategy aims to satisfy firstly gender segregation with respect to constraints on mandatory equipment
(b) $\hat{w}_m > w_g \geq w_a > w_{pr}^2$ if hospital strategy aims to satisfy firstly constraints on mandatory equipment, then gender segregation, age policy and preferred equipment.

Finally, if transfers are allowed to improve patients assignments, we suggest setting $\hat{w}_m > w_t$ such that a patient in $\mathcal{P}_1$ is transferred if at least one item of mandatory equipment is available in another room. The default penalty values defined in Demeester et al. (2010), and used in all papers that solved the benchmark instances with a soft version of PBAP, do not respect any of the relationships we have just suggested.

4. Our matheuristic approach

One of the main characteristics of metaheuristics has been their generality but over the years the focus of many metaheuristic applications has shifted towards performance, at the cost of losing generality (Blum, Puchinger, Raidl, & Roli, 2011). There are few general MIP heuristics in the MIP literature when compared to problem-specific heuristics (Akartunali & Miller, 2009).

A matheuristic is a heuristic based on mathematical programming and its name was defined by Maniezzo, Stützle, and Voß (2009). In this section, we review briefly some heuristics useful to address $\mathcal{NP}$-hard problems. The reader is referred to Blum et al. (2011), Ball (2011), Jourdan, Basseur, and Talbi (2009), Boussaïd, Lepagnot, and Siarry (2013) for recent surveys and a taxonomy. Then, we illustrate a general framework of our matheuristic and how it is implemented to solve the PBAP.

4.1. Local search heuristics

The literature devoted to heuristic algorithms often distinguishes between constructive algorithms and improvement algorithms. A constructive algorithm builds a solution from scratch by assigning values to one or more decision variables at a time; an improvement algorithm usually starts with a feasible solution and iteratively tries to improve the solution.

Neighborhood search algorithms aim to generate an improved solution at each iteration by searching the "neighborhood" of the current solution. Very Large Scale Neighborhood (VLSN) search algorithms are defined for those neighborhoods very large with respect to the size of the input data (Pisinger & Ropke, 2010). The large neighborhood search (LNS) is a heuristic introduced by Shaw (1998) and it belongs to the VLSN class. The two main parts characterizing the LNS heuristic are a destroy method and a repair method. Given a current solution, a destroy operation creates an infeasible solution, which is brought back into a feasible form by a repair heuristic. The LNS heuristic alternates then between an infeasible solution and a feasible solution (Pisinger & Ropke, 2010). The most important choice is the destroy method because it affects both computational time and quality of solutions: it can be time consuming or yield poor quality solutions, according to how the partial solution is repaired. If only a small part of the solution is destroyed, the heuristic can have trouble exploring the search space as the effect of a large neighborhood is lost. If a very large part of the solution is destroyed, the LNS heuristic almost degrades into repeated re-optimization. The degree of destruction can be gradually increased (Shaw, 1998) or chosen from a specific range dependent on the instance size (Ropke & Pisinger, 2006). Approaches based on LNS and VLNS have recently shown outstanding results in solving various transportation and scheduling problems, like the traveling salesman problem and capacitated vehicle routing problems (Ahuja, Ergun, Orlin, & Punnen, 2002; Ribeiro & Laporte, 2012).

Another interesting approach applied successfully to important real-world problems, like production lot-scheduling problems, is the relax and fix heuristic (Akartunali & Miller, 2009; Federgruen, Meissner, & Tzur, 2007; Pedroso & Kubo, 2005), used also in combination with tabu search (Pedroso & Kubo, 2005). Two important approaches in the literature that apply the relax-and-fix idea using "time windows" are due to Belvaux and Wolsey (2010), who introduce it in a production planning tool, and to Stadtler (2003). The main advantage of this heuristic is to solve submodels that are smaller and possibly easier than the original one. Integer variables of an optimization problem are partitioned into disjunctive sets; at each iteration only the variables of one of these sets are defined as integers while the remaining variables are relaxed. The resulting submodel is solved; then a subset of variables are fixed at their current values and the process is repeated for all the remaining sets (Ferreira, Morabito, & Rangel, 2010). Danna, Rothberg, and Pape (2005) propose the Relaxation Induced Neighborhood Search, which combines solutions of two related but different problems: the original mixed integer programming model and its continuous relaxation.

The PBAP here addressed is $\mathcal{NP}$-hard as demonstrated in Vancroonenburg et al. (2014) and the benchmark instances are computationally hard as well as reported in the literature. Furthermore, some of them require more than 24 hours to find a feasible solution with an ILP solver. Turhan and Bilgen (2017) solve Model (39)–(47) by using fix and relax heuristic and fix and optimize heuristic with time windows decomposition to find feasible solutions in a computational time less than three minutes. Upper bounds for these benchmarks have been found in Demeester et al. (2010), Bilgin et al. (2012), Ceschia and Schaerf (2011), Range et al. (2014) through suitable heuristic approaches. To solve the PBAP, we define and implement a matheuristic, which is tested on these benchmarks. The overall approach can be interpreted as a sequential rescheduling and it was inspired by the rescheduling approaches proposed in Conforti, Guerriero, and Guido (2008), Conforti, Guerriero, Guido, and Veltri (2011).

The proposed matheuristic is extremely effective because it exploits synergies of the relax and fix heuristic, the LNS heuristic and ILP solvers at the same time. Briefly, we found a feasible solution and fix some assignments of patients to rooms. This subproblem is solved and in the related neighborhood a better solution is found with a mathematical programming approach. Basically, our destroy method removes some patient-to-room assignments from the current solution and the remaining assignments are added as constraints to the problem formulation. The repair method is based on ILP solvers. We refer to our matheuristic as FiNeMath from now on. In what follows, we describe a framework of FiNeMath and detail it for the PBAP. Following the classification reported in Jourdan et al. (2009), FiNeMath is cooperative between a metaheuristic and an exact method.

4.2. A framework of FiNeMath

The strength of FiNeMath lies in its capability of finding an optimal or nearly optimal solution in a large search space by solving a sequence of subproblems. FiNeMath performs two main steps: the first step aims to find an initial feasible solution and the second step to improve the initial solution by an iterative scheme characterized by a sequence of subproblems solved by an ILP

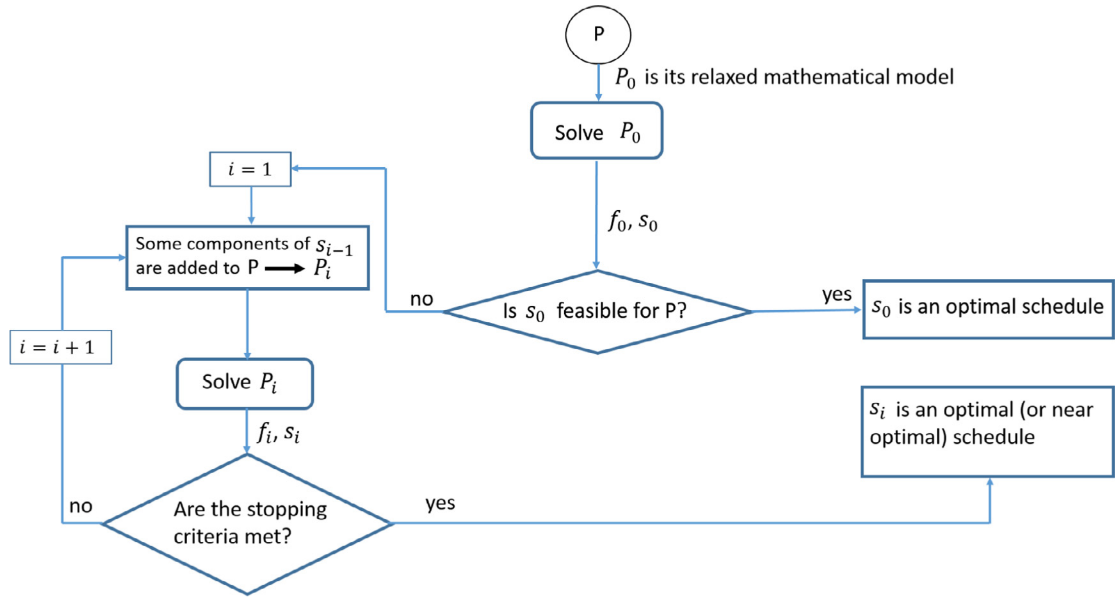


Fig. 3. Overview of the matheuristic approach.

solver. Each new subproblem is linked to its predecessor because is formulated incorporating knowledge by fixing some feasible assignments of the incumbent solution. An overview of the FiNeMath is depicted in Fig. 3.

The general idea we pursue is to explore the large search space of an ILP problem P by moving to best/good solutions of smaller search spaces. The optimization problem P is rather hard to solve because of some constraints C_0. Let s and f be its optimal solution and the corresponding objective function value, respectively. Let P_0 be the relaxed optimization problem of P obtained by removing the constraints C_0. Solve P_0 and let s_0 and f_0 be its optimal solution and objective function value, respectively. If s_0 is feasible for P, that is, s_0 satisfies the constraints C_0, then s_0 is an optimal solution to P because it is the schedule at minimum cost. If s_0 is not feasible for P, select those assignments of s_0 that satisfy C_0 and add them to P as constraints. This ILP model is P_1 and its search space is smaller than that of P. Solve P_1 and let s_1 and f_1 be its optimal solution and objective function value, respectively. If at least one of the defined stopping criteria is satisfied, FiNeMath stops; otherwise an iterative procedure runs until at least one of the defined stopping criteria holds. The iterative procedure is based on constructing ILP submodels by adding a subset of the current feasible assignments as constraints to P.

More formally, let S be the search space of P, and N_i be the neighborhood of the current solution s_i. FiNeMath converges monotonically to an optimal or near optimal solution and the following three properties hold for a sequence of subproblems:

Property 1 $\bigcup N_i \subseteq S$

Property 2 $N_i \bigcap N_{i+1} \neq \emptyset$. Indeed, let s_i be the optimal solution in N_i; $s_i \in N_{i+1}$ by construction.

Property 3 $f_{i+1} \leq f_i, i \geq 1$ because of Property 2. Indeed, let s_i and s_{i+1} be the optimal solution in N_i and N_{i+1}, respectively. Since $s_i \in N_{i+1}$ by construction, and the new solution optimal solution s_{i+1} is found at iteration $i+1$, then $f_{i+1} \leq f_i$ and $s_{i+1} \notin N_i \bigcap N_{i+1}$.

At the current iteration $i > 1$ an optimal solution s_{i-1} of the subproblem P_{i-1} is exploited to reduce the search space of a new subproblem P_i; a subset of components of the current solution s_{i-1} is added to P as equality constraints. Property 2 states that the optimal solution s_i of P_i will get better assignments of s_{i-1} or the

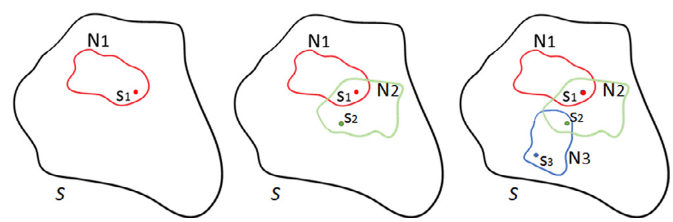


Fig. 4. Properties of FiNeMath in a search space S.

same/equivalent assignments in the worst case, that is $s_i = s_{i-1}$ and then $f_i = f_{i-1}$. Property 3 states that a sequence of objective function values decreasing monotonically characterizes the sequence of generated subproblems, i.e. $f_i \leq f_{i-1}, \forall i > 1$. Note that $f_0 \leq f \leq f_1$, that is f_0 and f_1 are a lower bound and the worst upper bound, respectively, for the objective function of P. The three properties are schematized in Fig. 4, where a big search space S is explored: s_1 is the optimal schedule in N_1; some assignments of this solution s_1 are added to P and N_2 is then defined. Observe that $s_1 \in N_1 \bigcap N_2$. The problem P_2 is solved and a schedule s_2 better than s_1 is found because $f_2 < f_1$. The current solution s_2 is exploited to define N_3 and the problem P_3 is solved.

4.3. FiNeMath for the PBAP

In this section, we detail the above framework of FiNeMath for solving the PBAP by illustrating the basic steps for the optimization problem HM_{PBA} in Algorithm 1. Then we show how to apply FiNeMath for solving SM_{PBA} and $SM\text{-}CLOS_{PBA}$. The relaxed model formulation of each optimization model presented above is defined by removing the constraints on the DGP, which make the problem $\mathcal{NP}$-hard. More specifically, Constraints (8)–(10) are removed from HM_{PBA}, and Constraints (8), (9) and Constraints (23) from SM_{PBA} and $SM\text{-}CLOS_{PBA}$. Note that in this last case the term F_g is also removed from the objective function because there is no penalty for DGP violations. Let P be the optimization model HM_{PBA} and P_0 its relaxed model. P_0 is solved and its optimal schedule s_0 consists of patient-to-room assignments at minimum cost. The schedule s_0 is optimal for P if the constraints on DGP are satisfied; otherwise s_0 is infeasible and is exploited to obtain a feasible solution for

Algorithm 1 The matheuristic approach FiNeMath for the HM_{PBA}.

Notation

Let I be an instance $(\mathcal{P}, \mathcal{R})$, where $\mathcal{P}$ is the set of patients and $\mathcal{R}$ the set of rooms.

Each patient has to be assigned to only one room $r, r \in \mathcal{R}$.

Let HM_{PBA} be the optimization model for the PBAP and C_0 be the constraints on DGP.

Warm start

Step 1. Remove C_0 from HM_{PBA}. Let P_0 be the relaxed optimization model.

Step 2. Solve (P_0, I). Let s_0 and f_0 be the optimal solution (schedule) and the optimal objective function value of P_0, respectively. Evaluate C_0 on s_0: if all constraints C_0 are feasible, then s_0 is the optimal solution of HM_{PBA}, STOP.Otherwise set $i = 1$

Step 3.

3.1 Select a subset $\bar{s}_0$ of feasible assignments for HM_{PBA} from the current solution s_0.

3.2 Formulate the problem P_1 by adding the feasible assignments $\bar{s}_0$ as constraints to HM_{PBA}.

3.3 Solve P_1. If P_1 is feasible, let s_1 and f_1 be the optimal solution and the optimal objective function value of P_1, respectively. Go to Step 5. Otherwise

3.4 Solve HM_{PBA} without its objective function. Let s_1 and f_1 be the feasible solution and the corresponding objective function value, respectively. Go to Step 5.

Iterative scheme:

Step 4. At the current iteration i:

4.1 Select a subset of feasible assignments for HM_{PBA} from the current solution s_{i-1}. Let $\bar{s}_{i-1}$ be this subset.

4.2 Formulate the problem P_i by adding the feasible assignments $\bar{s}_{i-1}$ as constraints to HM_{PBA}.

4.3 Solve P_i. Let s_i and f_i be the optimal solution and the optimal objective function value of P_i, respectively.

Step 5. If one of the stopping criteria is met, STOP: s_i is the optimal or suboptimal solution of HM_{PBA} and f_i is the corresponding objective function value.Otherwise $i = i + 1$, and go to Step 4.

P as detailed in Algorithm 1. A subset of feasible assignments of s_0 are selected and added to P. More specifically, let $\bar{\mathcal{P}}_{\mathcal{F}} \subseteq \mathcal{P}_{\mathcal{F}}$ and $\bar{\mathcal{P}}_{\mathcal{M}} \subseteq \mathcal{P}_{\mathcal{M}}$ be the subsets of women and men in s_0, respectively, such that constraints on DGP hold. The feasible assignments related to $\bar{\mathcal{P}}_{\mathcal{M}}$ and $\bar{\mathcal{P}}_{F}$ are added to HM_{PBA} as equality constraints. This subproblem is P_1 and its variables include those related to patients of the subsets $\tilde{\mathcal{P}}_{\mathcal{F}} = \mathcal{P}_{\mathcal{F}} \setminus \bar{\mathcal{P}}_{\mathcal{F}}$ and $\tilde{\mathcal{P}}_{\mathcal{M}} = \mathcal{P}_{\mathcal{M}} \setminus \bar{\mathcal{P}}_{\mathcal{M}}$, that is, for female and male patients of s_0 violating Constraints (10). The search space of P_1 is smaller than that of P. P_1 is solved, and let s_1 be its optimal schedule, now feasible for P. The aim of Step 3 is to find a good starting solution. If the added constraints slightly reduce the search space, a subset of found feasible assignments can be joined; if P_1 is infeasible due to the added constraints, a feasible solution can be found easily by solving HM_{PBA} without its objective function. The next step is to evaluate stopping criteria. We define and implement the following three stopping criteria based on a gap evaluated with respect to a computed lower bound value, computational time, and objective function improvements:

1. $f_i - LB \leq g$, where LB is a good lower bound and g is a given small constant.
2. $T \geq T_{max}$, where T and T_{max} denote the overall and the maximum computational time, respectively.
3. no improvements in the objective function values after k consecutive iterations.

The matheuristic stops if at least one of the defined stopping criteria holds otherwise a new subproblem is constructed and solved.

Fig. 5 depicts a simple example to better understand Step 3 of Algorithm 1: three men in $\mathcal{M} = \{A, B, C\}$ and two women in $\mathcal{F} = \{D, E\}$ have to be assigned to four double rooms $r_i, i = 1, \dots, 4$; two rooms are subject to DGP, one to RGP for men and one for women, denoted on the right of each room of Fig. 5 as DGP, M, and F, respectively. HM_{PBA} without constraints on DGP is solved; let s_0 be the solution, depicted in Fig. 5(a). s_0 is not feasible for HM_{PBA} because constraints on DGP are violated in room r_1. Steps 3.1 and 3.2 are performed: the problem P_1 is formulated by adding the feasible assignments for DGP of s_0, that is, assignments of patients A and D to room r_2, as constraints to HM_{PBA}. P_1 is solved and its solution s_1, reported in Fig. 5(b), is now feasible for HM_{PBA}. If no one stopping criterion holds, the iterative scheme is performed.

It is worth observing that for $i > 1$, the destroy and repair method are crucial. At Step 4.1 of Algorithm 1, the subset of feasible assignments is selected randomly and such that it is stratified. The cardinality of this subset has to be a compromise between a low computational cost and a good improvement in objective function values. With the aim to decrease rapidly the objective function value, the subset of feasible assignments added as constraints at iteration $i = 2$ is related to only patients with the same gender. As a consequence, all patients of the other gender are scheduled. After this suitable warm start procedure, there is then a sequence of objective function values decreasing monotonically and converging to an optimal or near optimal solution in a reasonable computational time.

FineMath for SM_{PBA} and $SM\text{-}CLOS_{PBA}$ differs slightly: every found feasible solution s_0 at Step 2 is feasible also for P and Step 3 is thus removed from Algorithm 1.

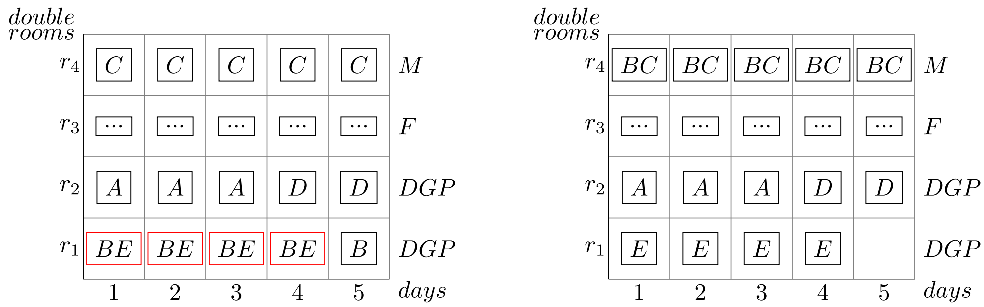
In order to test our matheuristic, we solve the thirteen PBAP benchmark instances, particularly time consuming, by carrying out several computational experiments, as described in the next section.

5. Computational experiments

In this section, we (i) summarize the main characteristics of the benchmark instances of the PBAP, (ii) report the experimental setting related to the default penalty values and propose new settings, (iii) solve the instances with FiNeMath, present our results and compare them with the best ones in the literature, (iv) analyze and discuss the results and solution approach in terms of quality and saving of computational time, respectively.

5.1. The benchmark instances of the PBAP

We solve the thirteen benchmark instances of the PBAP based on data collected of Belgian hospitals (Bilgin et al., 2012; Demeester et al., 2010). These instances are currently available at https://people.cs.kuleuven.be/~wim.vancroonenburg/pas/ (last visit August 2017). Their main characteristics are summarized as follows. The number of both departments and specialties is between 4 and 6. Every department has different levels of expertise in treating diseases of the several specialties. Three levels of expertise have been defined, i.e. high, medium, and low level, which are coded as 0, 1, and 2, respectively; the set of levels of expertise



(a) Infeasible solution s_0 for problem P

(b) Feasible solution s_1 for problem P

Fig. 5. Construction of an initial feasible solution s_1 for problem P.

Table 5
Characteristics of the instances: $|\mathcal{B}|$ =number of beds, $|\mathcal{R}|$ =number of rooms, $|\mathcal{R}_1|$=number of equipped rooms, $|\mathcal{P}|$=number of patients to be scheduled and overall number of patients (in parenthesis), $|\mathcal{P}_1|$=number of patients requiring mandatory equipment, D=planning horizon (in days), DGP=dependent gender policy (N:no, Y:yes), RGP=restricted gender policy (N:no, Y:yes), AP=age policy (N:no, Y:yes).

Instance	$\lvert\mathcal{B}\rvert$	$\lvert\mathcal{R}\rvert$	$\lvert\mathcal{R}_1\rvert$	$\lvert\mathcal{P}\rvert$	$\lvert\mathcal{P}_1\rvert$	D	DGP	RGP	AP
1	286	98	74	652 (693)	257	14	Y(98)	N	N
2	465	151	113	755 (778)	269	14	Y(151)	N	N
3	395	131	101	708 (757)	276	14	Y(131)	N	N
4	471	155	110	746 (782)	260	14	Y(155)	N	N
5	325	102	75	587 (631)	234	14	Y(102)	N	N
6	313	104	81	685 (726)	247	14	Y(104)	N	N
7	472	162	133	519 (770)	455	14	Y(39)	Y(82)	Y
8	441	148	124	895 (895)	788	21	Y(41)	Y(74)	Y
9	310	105	87	1400 (1400)	1213	28	Y(20)	Y(58)	Y
10	308	104	82	1575 (1575)	1373	56	Y(25)	Y(51)	N
11	318	107	90	2514 (2514)	2193	91	Y(26)	Y(53)	Y
12	310	105	89	2750 (2750)	2413	84	Y(33)	Y(54)	Y
13	368	125	108	907 (907)	805	28	Y(34)	Y(65)	Y

is then $Exp = \{0, 1, 2\}$. Each specialty prefers some rooms. To take into account this aspect, three levels of room suitability for specialties have been defined, i.e. high, medium, and low level, which are coded as 0, 1, and 2, respectively. The set of levels of suitability is thus $RS = \{0, 1, 2\}$. There are three categories of rooms, that is single, double, and quadruple rooms (thus $\mathcal{RC} = \{1, 2, 3\}$), and four types of bed equipment (thus $E = \{1, 2, 3, 4\}$). The planning horizon $\mathcal{D} = \{1, \dots, \hat{d}\}$ is a sequence of days and all patients must be discharged within $\hat{d}$. The patient's discharge date is set equal to $\hat{d}$ if the original discharge date is greater than $\hat{d}$, and his/her LOS is computed according to this setting. A room is assigned to a patient only if his/her LOS is at least one night. Hence, all patients whose admission and discharge are on the same day or the admission date is after $\hat{d}$ are removed. More specifically, Instances 1–6 have a planning horizon of 14 days, two types of equipment, and only the DGP. Instances 7–13 have a planning horizon between 14 and 91 days, two or four types of equipment, different age policies, and rooms are subject to the RGP, DGP or are without a gender policy. All patients of Instances 1–12 require only one specialty for his/her pathology, 202 patients of Instance 13 have a composite LOS because they require two specialties (see Co-mor(2.a) in Section 2.5).

Table 5 lists for each instance: the number of beds, the overall number of rooms, the number of equipped rooms, the number of patients, the number of patients requiring mandatory equipment, the number of days of the planning horizon, which gender policy is defined and whether there are age policies. Note that for Instances 1–7 the number of patients to be scheduled in the planning horizon is less than overall number of patients, reported in parenthesis. Moreover, all rooms of Instances 1–6 are subject to the DGP whereas Instances 7–13 have the DGP and the RGP. The number of rooms subject to the DGP and to the RGP are reported in parenthesis. For more details about the first six instances the reader is referred to Bilgin et al. (2012).

5.2. Experiment settings

We carried out several computational experiments based on two experiment settings, named ExpSet1 and ExpSet2, and described in the following.

The percentage of assignments, added as constraints to a problem formulation, affects the solution in terms of both computational time and improvement in the objective function value. A high percentage of added assignments decreases the computational time but often also an improvement in objective function values. We set this percentage between 20% and 30% of $|\mathcal{P}|$ for Instances 1–8 equivalent to a destruction between 80% and 70%, and between 25% and 55% of $|\mathcal{P}|$ for Instances 9–13, equivalent to a destruction between 75% and 45%. These percentages are a good compromise between a good improvement in objective function values and runtime. The highest percentage of destruction is used at the final runs of the matheuristic. The computational experiments have been performed using IBM ILOG CPLEX optimization Studio V15.5.1 with a free academic license on a computer equipped with Intel Xeon E5-1620, 3.60 gigahertz and 32.0 gigabytes of RAM.

5.2.1. ExpSet1

A quick way to check whether an instance is feasible for HM_{PBA} is to solve it without the objective function because the computational time is very short, that is 120 seconds for the largest instances: if a feasible solution is found, the instance is solved by HM_{PBA}, otherwise by SM_{PBA}. Instances 1–8 and 10–11 are thus solved by HM_{PBA}: patients are assigned to well-equipped rooms

Table 6
Experiment settings.

Instances	ExpSet1				ExpSet2			
	ILP model	Stopping criteria of FiNeMath			ILP model	Stopping criteria of FiNeMath		
		g	T_{max} (seconds)	k		g	T_{max} (seconds)	k
1–8	HM_{PBA}	0.01	800	3	HM_{PBA}, SM^S_{PBA}		7200	
9, 12	SM_{PBA}	0.01	1900	3	SM^S_{PBA}		10800	
10, 11	HM_{PBA}	0.01	1900	3	HM_{PBA}, SM^S_{PBA}		7200	
13	$SM\text{-}CLOS_{PBA}$	0.01	800	3				

and the gender policies and age policies are satisfied. Instances 9 and 12 are solved by SM_{PBA}, otherwise there is infeasibility due to a demand greater than supply. Instance 13 is the only one with patients requiring two specialties and it is solved by $SM\text{-}CLOS_{PBA}$.

We define three stopping criteria for FiNeMath, which iterates until at least one criterion is met: (SC1) the found schedule has an overall cost near to the computed lower bound, i.e. when the difference between the two values is at most equal to 0.01; (SC2) the overall computational time T is greater than the maximum T_{max}, which is instance size dependent. We set $T_{max} = 800$ seconds for Instances 1–8 and Instance 13, and $T_{max} = 1900$ seconds for the large Instances 9–12; (SC3) the third stop criterion aims to stop the matheuristic if there is no improvement in objective function values of three consecutive solved subproblems.

Each subproblem is solved optimally or with a gap of less than 1%; each current solution is a warm start, and every run is stopped if the gap is less than 3% and there is no improvement after 60 seconds.

5.2.2. ExpSet2

With the aim to further test FiNeMath, we set a time limit as only one criterion and (i) solve Instance 1–8 and Instance 10, 11 by HM_{PBA} and the results are compared with the UB^R values; (ii) solve all the instance by SM_{PBA} with RGP as soft constraints. We refer to this model as SM^S_{PBA}, which then differs from Model 3 only for no allowed transfers. The results are compared with the UB^C_F values, which are the lowest upper bounds found in Ceschia and Schaerf (2011) although they are reported in the table of lower bounds and no detail there is about parameter values and computational time because found throughout the whole experimentation campaign.

We fixed a time limit of three hours for I_9 and I_{12}, and two hours for the remaining instances. The two experimental settings are summarized in Table 6.

5.3. Penalty values

The best upper bounds known in the literature are found in Ceschia and Schaerf (2011), Range et al. (2014). These authors used the penalty values defined in Demeester et al. (2010). With the aim to compare our results with the best-known values, we use the default values; then we propose different penalty values for $v_5 - v_7$ to show how they affect solutions.

Default penalty values. In order to avoid misunderstanding, we report below how to set the weight values for $w^1_{pr}, w^2_{pr}, w^3_{pr}, w^4_{pr}$.

- Room-specialty suitability: patients require suitable treatment in terms of nursing and medical equipment. On the basis of the assessed preference by the specialty, the room assignment is penalized by $w^1_{pr} \in \mathcal{RS}$.
- Preferred equipment: every missed preferred equipment is penalized. The overall penalty is $w^2_{pr} = 2\sum_{e \in E | pr_{ep}=1}(pr_{pe} - eq_{re})$.
- Room department-specialty suitability: a penalty w^3_{pr} is induced when the department of a room does not have high expertise in treating patient's specialty sp_p. This penalty is set as $w^3_{pr} = 0$ if the expertise level is high or medium (codified as 0 and 1 in the datasets); otherwise $w^3_{pr} = 1$, if the expertise level is low (codified as 2 in the datasets).
- Room preference: a penalty $w^4_{pr} = 0.8$ is induced when the capacity of the assigned room is greater than the one preferred by patient ($C_r > rc_p$).

Table 7
Parameter settings.

Default penalty values for $v_1 - v_4$	Settings of penalty values for $v_5 - v_7$
$w^1_{pr} \in \{0, 1, 2\}$	*ParSet*0: $w_m = 5,\ w_g = 5,\ w_a = 10$
$w^2_{pr} = 2\sum_{e \in E \| pr_{ep}=1}(pr_{ep} - eq_{re})$	*ParSet*1: $w_m = 5,\ w_g = 100,\ w_a = 10$
$w^3_{pr} \in \{0, 1\}$	*ParSet*2: $w_m = 100,\ w_g = 5,\ w_a = 10$
$w^4_{pr} = 0.8$ if $rc_p < C_r$, 0 otherwise	*ParSet*3: $w_m = 130,\ w_g = 50,\ w_a = 50$

The values related to violations $v_5 - v_7$ are reported in Table 7 and denoted as *ParSet*0.

New settings of penalty values for $\boldsymbol{v_5 - v_7}$. We carry out further computational experiments on the instances solved by SM_{PBA} with the following settings, based on what has been discussed in Section 3.2.

- *ParSet*1: The penalty cost on gender policy violations is set as $w_g = 100$ to strongly penalize violations of the DGP.
- *ParSet*2: The penalty cost on mandatory equipment is set as $w_m = 100$. This models the situation (b) with $w_m > w_a > w_g$.
- *ParSet*3: The penalty costs are set as $w_m = 130, w_g = w_a = 50$. The aim is to assign patients with mandatory equipment requirements to suitable rooms. This models the situation (b) with no preferences between violations on gender policies and age policies, that is, $w_m > w_a = w_g$.

These new settings are tested on Instances 9 and 12.

We run FiNeMath to solve the benchmark instances by combining the two experimental settings with the four parameter settings. The results are presented and compared with the best-known values in the next section.

5.4. Computational results

In this section, we report and discuss the results of the computational experiments carried out on the benchmark instances with the experimental settings and parameter settings reported in Table 6 and Table 7, respectively. We carried out also further computational experiments in order to explore the quality of the found values and test the efficiency of FiNeMath.

5.4.1. Results with *ExpSet*1 – *ParSet*0

We run FiNeMath with *ExpSet*1 and the default penalty values *ParSet*0. Our results and the best ones reported in the literature are shown in Table 8. The first column reports the instance

Table 8
Results on the benchmark instances with $ExpSet1 - ParSet0$: I (instance number), LB (lower bound), UB (our upper bound), gap, UB^R and UB^C (the best upper bounds reported in the literature), $\Delta UB(\%)$ (percentage difference with respect to the best UB in the literature). Violation value: v_1 (correct specialty), v_2 (preferred equipment), v_3 (department with the right specialty), v_4 (room category preferences), v_5 (mandatory equipment), v_6 (dependent gender policy), v_7 (age policy). F_t (transfers), computational time (in seconds), and SC (stopping criterion).

I	LB	UB	Gap (%)	UB^R	UB^C	ΔUB (%)	v_1	v_2	v_3	v_4	v_5	v_6	v_7	F_t	Time (seconds)	SC
1	644.9	652.8	1.21	654.4	659.2	0.24	0	0	0	652.8					259	3
2	1111.4	1134.4	2.02	1130.4	1143.6	−0.35	16	0	0	1118.4					578	3
3	747.0	764.2	2.25	768.2	776.6	0.52	13	0	0	751.2					789	3
4	1141.1	1162.2	1.81	1179.0	1176	1.17	87	36	0	1039.2					848	3
5	620.8	624	0.51	624	625.6	0	0	0	0	624					173	1
6	787.0	794.2	0.90	792.6	801.2	−0.20	3	0	0	791.2					451	1
7	1160.5	1176.6	1.36	1215.0	1199	1.86	158	268	20	730.6					295	3
8	4015.3	4068	1.29	4425.0	4158.6	2.17	891	1532	213	1432					440	3
9	20356.0	20832.8	2.28		21942	5.05	1193	10,674	300	2060.8	1040	225	4720		1900	2
10	7682.7	7806.4	1.58		8146.6	4.17	447	4400	1	2958.4					1923	2
11	10946.8	11536.2	5.11		12016.8	4.00	944	6242	3	4347.2					1826	3
12	21813.0	22707.2	3.93		23758.4	4.42	1717	15,562	299	4839.2	190	100	0		1905	2
13	8775.8	9109.8	3.66		9426.8	3.36	1655	4458	610	2060.8	45	50	0	231	964	2

number; the next three columns report the best computed lower bound (LB), the found upper bound (UB) values, and the percentage gap computed as $(UB - LB)/UB$. Then, the fourth and the fifth columns show the upper bounds UB^R and UB^C found in Range et al. (2014) and Ceschia and Schaerf (2011), respectively. The sixth column reports the gap between the best value of the literature, which is underlined, and our UB value. This gap is computed as $(bestUB - ourUB)/bestUB$: A positive value is an improvement, a negative is a worsening. Note that Instances 9–13 were not solved in Range et al. (2014) due to increased complexity and size of their model; a dash line means that no upper bounds have been computed. The next seven columns show the values of the weighted violations $v_1 - v_7$: value 0.0 means that there are no violations, whereas a dash line means that violations are not allowed because of hard constraints. Only for Instance 13 is there the term F_t on the penalized transfers. The second-to-last column reports the overall computational time (in seconds), and finally, the last one specifies the satisfied stopping criterion.

We computed two LB values for each instance: the first LB value by solving the related linear optimization model, i.e., we relaxed constraints on integer variables; the second one by solving the relaxed ILP model constructed by removing the constraints on the DGP. This last LB value is of lower quality than the first one. Table 8 reports the best LB value: the LB values of Instances 1–8 and 10, 11 are related to the linear optimization model whereas those of Instances 9 and 12, 13 to the relaxed ILP model. All the values are optimal, except for Instances 12 (gap= 0.54 after 8600 seconds).

We found with the penalty setting $ParSet0$ new best values for ten out of the thirteen instances and the highest improvement is on Instance 9 ($\Delta UB(\%) = 5.05$). The most violated constraints are on preferred equipment and room category preferences, which can be explained by a high demand and scarce available resources. Indeed, a demand for single and double rooms exceeds the supply.

In what follows, we analyze the results of Table 8 in detail and discuss their quality. UB_i and LB_i denote the upper bound value and the lower bound value of instance i, respectively. Instances 1–8 and 10, 11 do not have values for $v_5 - v_7$ because these violations are not allowed in HM_{PBA}. The found UB_1 and UB_5 values are completely due to violations of room category preferences. Only bounds on Instances 2 and 6 are slightly worse than the corresponding best known values; observe the first stop condition of FiNeMath, based on a gap between UB_i and LB_i, holds only for these two instances. The third stop condition, that is any improvement after three consecutive iterations, holds for Instances 1–4, 7, 8 and 10. With the aim to analyze the quality of UB_7 and UB_8 we carried out the following computational experiments:

- We used the final schedule related to $UB_7 = 1176.6$ as a warm start to schedule optimally all patients of Instance 7 by HM_{PBA}. The optimal objective function value of Instance 7 is 1176.4, that is UB_7 differs with respect to the optimal value of only 0.2.
- The value $UB_8 = 4068$ was found by solving ten subproblems in 440 seconds. In order to explore the quality of UB_8, we add only 41 assignments of our final schedule (i.e., 4.5% of $|\mathcal{P}|$) as constraints to HM_{PBA}: the result is UB=4065.8 after 1195 seconds with a gap of 0.49. Finally, in order to test the efficiency of FiNeMath in terms of saving of computational time, we perform a further computational experiment on Instance 8 by solving HM_{PBA} with CPLEX: the incumbent objective function value is 4076.6 after 7215 seconds with a gap of 0.93, whereas the value $UB_8 = 4068.0$ was found by solving ten subproblems in 440 seconds.

FiNeMath finds in most cases better values and in a shorter computational time with respect to the reported computational times in the literature. Indeed, Ceschia and Schaerf (2011) fixed a total number of iterations to $I = 6.7 \times 10^8$, which results in a running time of about 400 seconds on their computer but they do not specify how many times each run has been performed. Some of our values close to UB^C are found in less than 300 seconds. Range et al. (2014) perform sixteen different experimental settings and impose a time limit of 18.000 seconds: there is no the best setting and the UB^R values are the best values. In Section 5.4.3 we improve our UB and the best known values reported in the literature. Fig. 6 shows the performance of FiNeMath on six benchmark instances. Each dot represents a solved subproblem with its objective function value and computational time. The sequence of dots shows how the objective function values decrease over time. An individual iteration takes at most 300 seconds. We can observe: (1) a rapid improvement in the objective function values (in only 3 and 4 iterations) and a slow improvement when the incumbent value is near to the computed LB value in every plot; (2) usually, the incumbent values close to the UB^C values are reached in only 2 and 3 iterations and in less than 300 seconds. Every solution s_0 at Step 2 of Algorithm 1 is feasible for Instances 9 and 12 because they are solved by SM_{PBA}. Step 3.3 was performed for Instance 1–8 and Instance 10; Step 3.4 only for Instance 11.

In order to explore how different penalty values affect results, we test the penalty values of Table 7 on Instances 9 and 12. The results are show and discussed in the following section.

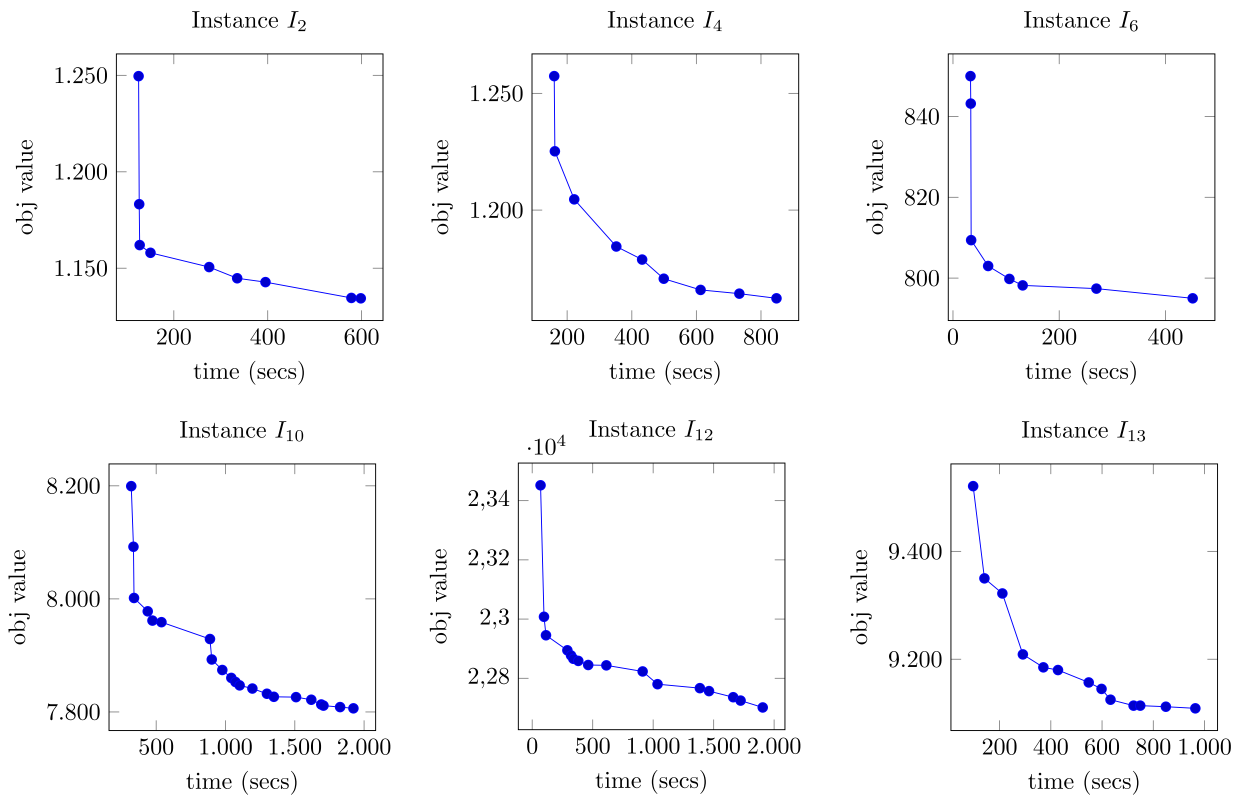


Fig. 6. Plots of the performance of our matheuristic on six benchmark instances.

Table 9
Results on the benchmark Instances 9 and 12 with the three proposed values: *I* (instance number), parameter setting, *UB* (upper bound). Violation value: v_1 (correct specialty), v_2 (preferred equipment), v_3 (department with the right specialty), v_4 (room category preferences), v_5 (mandatory equipment), v_6 (DGP), v_7 (age policy). Computational time (in seconds), and SC (stopping criterion)

I	Setting	*UB*	v_1	v_2	v_3	v_4	v_5	v_6	v_7	Time (seconds)	SC
9	*ParSet*1	20838.2	1196.0	10608.0	304.0	2715.2	1165.0	0.0	4850.0	1922	2
9	*ParSet*2	21515.8	1287.0	11069.0	353.0	2736.8	360.0	345.0	5380.0	1903	2
9	*ParSet*3	22272.8	1960.0	11854.0	635.0	2808.8	370.0	5.0	4640.0	1913	2
12	*ParSet*1	22865.2	1786.0	15885.0	311.0	4883.2	0.0	0.0	0.0	1923	2
12	*ParSet*2	22896.2	1807.0	15622.0	284.0	4843.2	0.0	270.0	70.0	1940	2
12	*ParSet*3	22852.4	1702.0	15990.0	262.0	4878.4	0.0	0.0	20.0	1930	2
12	DGP constraints	22679.6	1701.0	15570.0	264.0	4869.6	275.0	0.0	0.0	1905	2

5.4.2. Quality improvements with the proposed penalty values

As already introduced, we test the penalty values reported in Table 7 for $v_5 - v_7$ on Instances 9 and 12 in order to explore the quality of the found values. On Instance 12 we carry out a further computational experiment with *ParSet*0 and hard constraints on the DGP, i.e., we introduce Constraints (8)–(10) in SM_{PBA}; the term F_g of the objective function and Constraints (23) are then removed. The final schedules have been re-evaluated with the default values for comparative purposes. The results, in Table 9, show improvements in quality of the schedules. Indeed, to highlight the effects of the penalty values on the results of Instance 9, we display in Fig. 7 how the seven values of violations $v_1 - v_7$ change according to the used penalty values. The most remarkable result to emerge from this figure is that with:

*ParSet*1: the violations on the DGP are null but violations on the room category preferences, mandatory equipment, and age policy increase with respect to their corresponding value found with the default values;

*ParSet*2: the violations on mandatory equipment decrease (i.e., v_5) but v_2, v_4, v_7 increase with respect to the values found with the default weights. This result shows that a greater number of patients requiring mandatory equipment are assigned to suitable rooms but, on the other hand, there is a greater number of violated constraints on preferred equipment, category preferences and age policies.

*ParSet*3: v_5, v_6, v_7 have the minimum value at the expense of v_1, v_2, v_3, v_4, which increase. This schedule is thus better than the other two for that concerning quality of medical needs but it is the worst if we compare the objective function values.

Similar conclusions can be drawn with regard to the results of Instance 12: also in this case, improving schedule quality evaluated in terms of violations $v_5 - v_7$ implies a worsening of the objective function value. In addition, we observe that the best objective function value is found with the added hard constraints on DGP to SM_{PBA}, but the value $v_6 = 0$ involves a worsening of violations on mandatory equipment v_5.

5.4.3. Results with ExpSet2 – ParSet0

In this section we present the results obtained by running FiNeMath with *ExpSet*2 – *ParSet*0. Instances 1–8 and 10, 11 are solved with HM_{PBA}. Since Range et al. (2014) solved the original PBAP, we

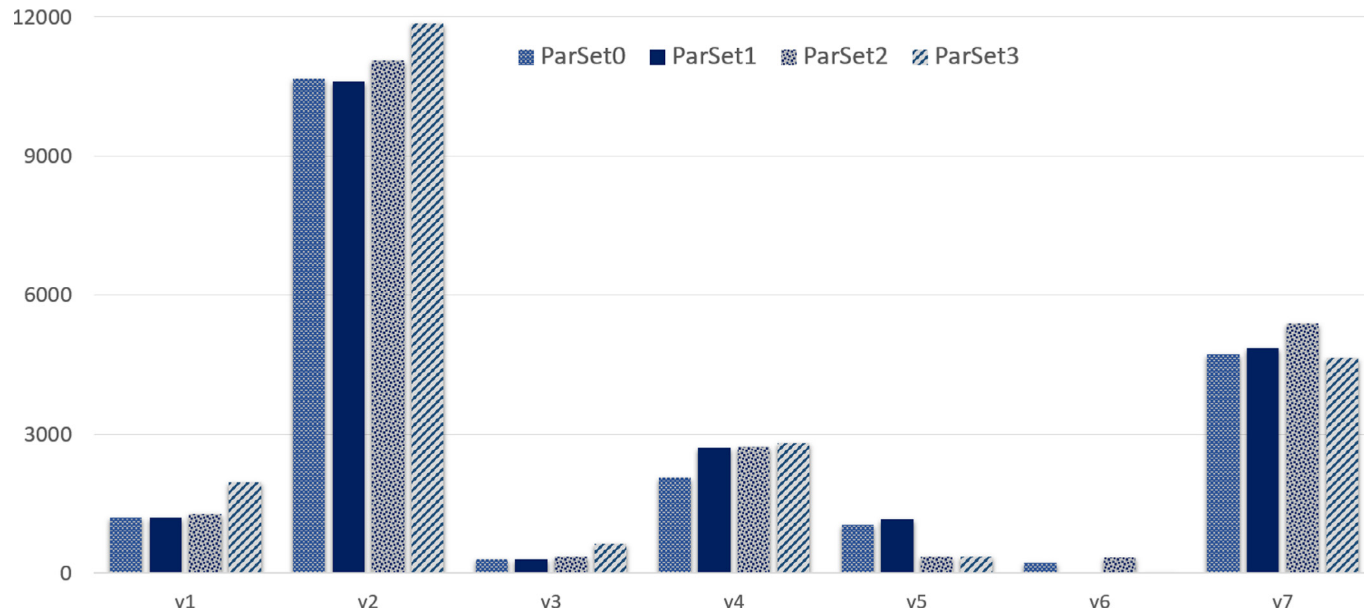


Fig. 7. Constraint violations for Instance 9 based on four settings of penalty values. v_1 (correct specialty), v_2 (preferred equipment), v_3 (department with the right specialty), v_4 (room category preferences), v_5 (mandatory equipment), v_6 (dependent gender policy), v_7 (age policy).

Table 10
Results with *ExpSet*2 – *ParSet*0: *I* (instance number), *LB* (lower bound), *UB* (our upper bound), UB^R (and UB_F^C). Gap (in %) between: *LB* and *UB*, *LB* and UB^R (and UB_F^C), UB^R (and UB_F^C) and *UB*. Violation values: v_1 (correct specialty), v_2 (preferred equipment), v_3 (department with the right specialty), v_4 (room category preferences), v_5 (mandatory equipment), v_6 (DGP and RGP), v_7 (age policy).

		Lower and upper bound values			Percentage gap			Violation values						
Opt model	*I*	*LB*	*UB*	UB^R	*Gap*	$GapUB^R$	ΔUB	v_1	v_2	v_3	v_4	v_5	v_6	v_7
HM_{PBA}	1	644.9	652.8	$\underline{654.4}$	1.21	1.45	0.24	0.0	0.0	0.0	652.8			
	2	1110.4	**1128.0**	$\underline{1130.4}$	1.56	1.76	0.21	8.0	0.0	0.0	1120.0			
	3	747.0	**761.6**	$\underline{768.2}$	1.91	2.75	0.86	8.0	0.0	0.0	753.6			
	4	1140.8	**1152.2**	1179.0	0.98	0.98	1.73	73.0	36.0	0.0	1043.2			
	5	620.8	624.0	$\underline{624.0}$	0.51	0.51	0.0	0	0.0	0.0	624.0			
	6	787.0	**792.6**	$\underline{792.6}$	0.70	0.70	0.0	0.2	3.0	0.0	791.2			
	7	1160.5	**1176.4**	1215.0	1.35	4.52	3.17	158.0	268.0	20.0	730.6			
	8	4015.3	**4065.4**	4425.0	1.23	9.25	8.12	893.0	1538.0	216.0	1418.4			
	10	7682.68	**7804.6**		1.56			447.0	4400.0	1.0	2956.6			
	11	10946.8	**11491.8**		4.74			962.0	6060.0	1.0	4348.8		120.0	
Opt model	*I*	*LB*	*UB*	UB_F^C	*Gap*	$GapUB_F^C$	ΔUB	v_1	v_2	v_3	v_4	v_5	v_6	v_7
SM_{PBA}^S	1	644.9	**652.0**	655.6	1.08	1.63	0.54	0.0	0.0	0.0	652.0	0.0	0.0	0.0
	2	1110.4	1129.6	1137.6	1.69	2.39	0.70	4.0	0.0	0.0	1125.6	0.0	0.0	0.0
	3	747.0	764.2	773.6	2.25	3.43	1.21	13.0	0.0	0.0	751.2	0.0	0.0	0.0
	4	1140.8	1154.0	$\underline{1172.2}$	1.14	2.67	1.57	78.0	0.0	0.0	1040.0	0.0	0.0	0.0
	5	620.8	624.0	625.6	0.51	0.76	0.25	0.0	0.0	0.0	624.0	0.0	0.0	0.0
	6	787.0	794.2	798.0	0.90	1.37	0.47	3.0	0.0	0.0	791.2	0.0	0.0	0.0
	7	1160.5	**1176.4**	$\underline{1193.0}$	1.35	2.72	1.39	158.0	268.0	20.0	730.6	0.0	0.0	0.0
	8	4015.3	**4065.4**	$\underline{4149.8}$	1.23	3.24	2.03	893.0	1538.0	216.0	1418.4	0.0	0.0	0.0
	9	20356.0	**20718.6**	21501.8	1.75	5.32	3.64	1205.0	10636.0	321.0	2701.6	1035.0	60.0(20.0)	4740.0
	10	7682.68	7831.0	8036.2	1.89	4.39	2.55	459.0	4376.0	1.0	2960.0	0.0	10.0 (20.0)	0.0
	11	10946.8	**11491.8**	11811.8	4.74	7.32	2.71	962.0	6055.0	1.0	4348.8	5.0	120.0	0.0
	12	21813.0	23145.2	23344.2	5.75	6.55	0.85	1891.0	15644.0	290.0	4935.2	220.0	110.0 (55)	0.0

compare our results with the UB^R values, as reported in the first part of Table 10. In this table, there are also the percentage gap between *LB* and *UB*, *LB* and UB^R, and UB^R and *UB*. In order to show how FiNeMath converges to a good solution despite a huge search space, we solve all the instances by SM_{PBA}^S. These results are in the second part of Table 10 and they are compared with the UB_F^C values. The values in parentheses on column v_6 refer to violations of RGP.

The new best values are in bold in these table and all the best-known values, which are underlined, are improved. Only the value of Instance 12 is not in bold because the best value we found is in Table 9. Finally, observe that most of our percentage gap from the *LB* are below 2% and that the values related to Instance 7 are optimal, as already found in Section 5.4.1.

5.5. Evaluation of FiNeMath and discussion

Our results suggest that FiNeMath outperforms the other proposed solution approaches in terms of quality of schedules and computational times. Turhan and Bilgen (2017) find feasible schedules with a gap less than 15% from the best known values and in low computational times for six small instances, whereas the gap is less than 21% for the rest. If we compare these performances with those reported in Fig. 6, we observe that FiNeMath finds similar or better values and in a shorter computational time. Moreover, FiNeMath depends only on the degree of solution destruction and can be considered as a general solution approach because no specific strategy problem-dependent has been designed for fixing variables.

On the basis of our best values for the benchmark instances, we can assert that HM_{PBA} is appropriate for solving feasible instances of the PBAP because the search space is the smallest and no suitable penalty costs have to be defined. If infeasibility occurs due to scarcity of resources, the instances can be solved by a soft version of HM_{PBA}. In this case, unfulfilled constraints, i.e., gender policies, age policies, and mandatory requirements have to be penalized by well-defined values and accordingly to hospital strategies to avoid an “unexpected” behavior like the one analyzed in Section 3. A soft version of HM_{PBA} can be solved even when a problem is hard to solve without objective function and then without knowing if it is feasible for HM_{PBA}. Moreover, the search spaces of the two optimization models are different: the smaller is a search space the better is the computational time to find good solutions. However, the computational results show that although SM_{PBA} and thus SM^S_{PBA} with no hard constraints on RGP have a huge search space, FiNeMath provides tighter bounds than the best-known values for all instances and results in an improved matching demand to supply.

In some hospitals, such as the Belgian ones, room category preference constraints are important because there are significant financial incentives for both hospitals and physicians when patient preference concerning room category is met. Moreover, room supplements and physician fee supplements may be charged, making the final cost for a patient preferring a single room as high as 400% over a double/communal room (Vancroonenburg, De Causmaecker, Spieksma, & Vanden Berghe, 2013).

For completeness, as our schedules consist of patient-to-room assignments, we solve the patient-bed assignment problem that requires every bed to be occupied at most by one patient and every patient is assigned to only one bed. Our optimal patient-bed assignments of each solved instance are input to the Java solution validator now available at https://people.cs.kuleuven.be/~wim.vancroonenburg/pas/. The results of the Java solution validator are available at

http://www.dehealthlab.it/en/21-papers/55-guido-etal-pbap.

6. Conclusion and future work

The patient bed assignment problem is a complex combinatorial problem. In this paper, we have presented three optimization models and shown the main differences between the hard and the soft versions in terms of search space. We have also provided further evidence that a no appropriate schedule can be found with a soft version of the PBAP when no suitable penalty values are used, and suggested some tips on how to set them. We have developed an efficient matheuristic, FiNeMath, inspired to rescheduling approaches and classified as cooperative between a metaheuristic and an exact method. The matheuristic, for its nature, allows to get to good results with a stronger model and in less time if compared to a soft version, specially when a soft version is not needed. Indeed, our results show that a hard optimization model is more effective than a soft version and all the best-known values of the benchmark instances are improved. Further computational experiments have been carried out in order to explore the quality of the results and test the efficiency of FiNeMath in huge search space.

A benefit of this approach is to support hospital assignment staff in assigning beds to patients with minimal effort and in a small amount of time. Admission systems similar to those in Belgian hospitals, where there are significant financial incentives for hospitals and physicians in meeting some preferences, could exploit this approach to match patient preference concerning room type (single or double/communal room).

The computational results show that the designed matheuristic exhibits very good performance and it might inspire novel solution approaches for problems like the $\mathcal{NP}$-hard red–blue transportation problems (Vancroonenburg et al., 2013). We are currently testing FiNeMath on the benchmark instances of the PASU problem, and PASU with surgeries scheduling problem and good preliminary results confirm that FiNeMath is a fast and efficient solution approach. Future works will focus on (1) the patient bed assignment problem as a multi-objective optimization problem, which can be solved by lexicographic approaches and genetic algorithms, (2) testing the proposed matheuristic on benchmarks of the red–blue transportation problems, (3) improving performance of FiNeMath by designing effective strategies for fixing variables.

Acknowledgment

We would like to thank Eugenio Rende who created a tool for computing input cost matrices, Professor Alessandro Agnetis and Professor Andrea Schaerf who suggested us that our proposed matheuristic is based on LNS heuristic. We thank the anonymous reviewers for their helpful comments and questions.

References

Adan, I., & Vissers, J. (2002). Patient mix optimisation in hospital admission planning: A case study. *International Journal of Operations & Production Management, 22*(4), 445–461.

Ahuja, R. K., Ergun, O., Orlin, J. B., & Punnen, A. P. (2002). A survey of very large-scale neighborhood search techniques. *Discrete Applied Mathematics, 123*(1–3), 75–102.

Akartunali, K., & Miller, A. J. (2009). A heuristic approach for big bucket multi-level production planning problems. *European Journal of Operational Research, 193*(2), 396–411.

Bachouch, R. B., Guinet, A., & Hajri-Gabouj, S. (2012). An integer linear model for hospital bed planning. *International Journal of Production Economics, 140*(2), 833–843.

Ball, M. O. (2011). Heuristics based on mathematical programming. *Surveys in Operations Research and Management Science, 16*(1), 21–38.

Barz, C., & Rajaram, K. (2015). Elective patient admission and scheduling under multiple resource constraints. *Production and Operations Management, 24*(12), 1907–1930.

Bekker, R., & Koeleman, P. (2011). Scheduling admissions and reducing variability in bed demand. *Health Care Management Science, 14*(3), 237–249.

Belvaux, G., & Wolsey, L. (2010). bc-prod: A specialized branch-and-cut system for lot-sizing problems. *Management Science, 46*(5), 724–738.

Bilgin, B., Demeester, P., Misir, M., Vancroonenburg, W., & Vanden Berghe, G. (2012). One hyper-heuristic approach to two timetabling problems in health care. *Journal of Heuristics, 18*(3), 401–434.

Blum, C., Puchinger, J., Raidl, G. R., & Roli, A. (2011). Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing, 11*(6), 4135–4151.

Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences, 237*, 82–117.

Brailsford, S., & Vissers, J. (2011). OR in healthcare: A European perspective. *European Journal of Operation Research, 212*(2), 223–234.

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

Castro-Gutierrez, J., Landa-Silva, D., & Pérez, J. M. (2010). Improved dynamic lexicographic ordering for multi-objective optimisation. In R. Schaefer, C. Cotta, J. Kołodziej, & G. Rudolph (Eds.), *Proceedings of the eleventh international conference on parallel problem solving from nature, PPSN XI, Kraków, Poland* (pp. 31–40). Berlin, Heidelberg: Springer. September 11–15, 2010, part II

Ceschia, S., & Schaerf, A. (2011). Local search and lower bounds for the patient admission scheduling problem. *Computers & Operations Research, 38*(10), 1452–1463.

Ceschia, S., & Schaerf, A. (2012). Modeling and solving the patient admission scheduling problem under uncertainty. *Artificial Intelligence in Medicine, 56*(3), 199–205.

Ceschia, S., & Schaerf, A. (2016). Dynamic patient admission scheduling with operating room constraints, flexible horizons, and patient delays. *Journal of Scheduling, 19*(4), 337–389.

Conforti, D., Guerriero, F., & Guido, R. (2008). Optimization models for radiotherapy patient scheduling. *4OR, 6*(3), 263–278.

Conforti, D., Guerriero, F., Guido, R., Cerinic, M., & Conforti, M. (2011). An optimal decision making model for supporting week hospital management. *Health Care Management Science, 14*(1), 74–88.

Conforti, D., Guerriero, F., Guido, R., & Veltri, M. (2011). An optimal decision-making approach for the management of radiotherapy patients. *OR Spectrum, 33*(1), 123–148.

Danna, E., Rothberg, E., & Pape, L. C. (2005). Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming, 102*(1), 71–90.

Demeester, P., Souffriau, W., De Causmaecker, P., & Vanden Berghe, G. (2010). A hybrid tabu search algorithm for automatically assigning patients to beds. *Artificial Intelligence in Medicine, 48*(1), 61–70.

Federgruen, A., Meissner, J., & Tzur, M. (2007). Progressive interval heuristics for multi-item capacitated lot-sizing problems. *Operations Research, 55*(3), 490–502.

Ferreira, D., Morabito, R., & Rangel, S. (2010). Relax and fix heuristics to solve one-stage one-machine lot-scheduling models for small-scale soft drink plants. *Computers & Operations Research, 37*(4), 684–691.

Guerriero, F., & Guido, R. (2011). Operational research in the management of the operating theatre: a survey. *Health Care Management Science, 14*(1), 89–114.

Guido, R., & Conforti, D. (2017). A hybrid genetic approach for solving an integrated multi-objective operating room planning and scheduling problem. *Computers & Operations Research, 87*(Suppl C), 270–282.

Guido, R., Solina, V., & Conforti, D. (2017). Offline patient admission scheduling problems. In *Proceedings of the 2017 mathematics & statistics* (pp. 129–137).

Hulshof, P., Kortbeek, N., Boucherie, R., Hans, E., & Bakker, P. (2012). Taxonomic classification of planning decisions in health care: a structured review of the state of the art in OR/MS. *Health Systems, 1*, 129–175.

Jourdan, L., Basseur, M., & Talbi, E. G. (2009). Hybridizing exact methods and metaheuristics: A taxonomy. *European Journal of Operational Research, 199*(3), 620–629.

Lei, X., Na, L., Xin, Y., & Fan, M. (2014). A mixed integer programming model for bed planning considering stochastic length of stay. In *Proceedings in the 2014 IEEE international conference on automation science and engineering* (pp. 1069–1074).

Lusby, R. M., Schwierz, M., Range, T. M., & Larsen, J. (2016). An adaptive large neighborhood search procedure applied to the dynamic patient admission scheduling problem. *Artificial Intelligence in Medicine, 74*, 21–31.

Maniezzo, V., Stützle, T., & Voß, S. (2009). *Matheuristics: Hybridizing metaheuristics and mathematical programming* (1st). Springer Publishing Company, Incorporated. ISBN 144191305X, 9781441913050.

Mazier, A., Xie, X., & Sarazin, M. (2010). Real-time patients assignment: A method for improving emergency department flow. Proceedings in the 2010 Workshop on Health Care Management IEEE, pp. 1–6.

Misir, M., Verbeeck, K., Causmaecker, P. D., & Berghe, G. V. (2013). An investigation on the generality level of selection hyper-heuristics under different empirical conditions. *Applied Soft Computing, 13*(7), 3335–3353.

Pedroso, J. P., & Kubo, M. (2005). *Hybrid tabu search for lot sizing problems* (pp. 66–77). Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-31898-9.

Pisinger, D., & Ropke, S. (2010). *Large neighborhood search* (pp. 399–419). Boston, MA, US: Springer. ISBN 978-1-4419-1665-5.

Range, T. M., Lusby, R. M., & Larsen, J. (2014). A column generation approach for solving the patient admission scheduling problem. *European Journal of Operational Research, 235*(1), 252–264.

Ribeiro, G. M., & Laporte, G. (2012). An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. *Computers & Operations Research, 39*(3), 728–735.

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science, 40*(4), 455–472.

Schmidt, R., Geisler, S., & Spreckelsen, C. (2013). Decision support for hospital bed management using adaptable individual length of stay estimations and shared resources. *BMC Medical Informatics and Decision Making, 13*, 1–19.

Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *Proceedings of the fourth international conference on principles and practice of constraint programming* (pp. 417–431). London, UK: Springer-Verlag. ISBN 3-540-65224-8.

Stadtler, H. (2003). Multilevel lot sizing with setup times and multiple constrained resources: Internally rolling schedules with lot-sizing windows. *Operations Research, 51*, 487–502.

Teixeira, A., & De Oliveira, M. J. (2015). Operations research on hospital admission systems: A first overview of the 2005–2014 decade. *Journal of Physics: Conference Series, 616*(1), 012009.

Thomas, B. G., Bollapragada, S., Akbay, K., Toledano, D., Katlic, P., Dulgeroglu, O., et al. (2013). Automated bed assignments in a complex and dynamic hospital environment. *Interfaces, 43*(5), 435–448.

Turhan, A. M., & Bilgen, B. (2017). Mixed integer programming based heuristics for the patient admission scheduling problem. *Computers & Operations Research, 80*, 38–49.

Vancroonenburg, W., De Causmaecker, P., Spieksma, F., & Vanden Berghe, G. (2013). Scheduling elective patient admissions considering room assignment and operating theatre capacity constraints. *Lecture Notes in Management Science, 5*, 153–158.

Vancroonenburg, W., De Causmaecker, P., & Vanden Berghe, G. (2016). A study of decision support models for online patient-to-room assignment planning. *Annals of Operations Research, 239*(1), 253–271.

Vancroonenburg, W., Della Croce, F., Goossens, D., & Spieksma, F. (2014). The red–blue transportation problem. *European Journal of Operational Research, 237*(3), 814–823.