Ann Oper Res (2016) 239:253–271
DOI 10.1007/s10479-013-1478-1

A study of decision support models for online patient-to-room assignment planning

Wim Vancroonenburg · Patrick De Causmaecker · Greet Vanden Berghe

Published online: 16 October 2013

Abstract The present paper studies patient-to-room assignment planning in a dynamic context. To this end, an extension of the patient assignment (PA) problem formulation is proposed, for which two online ILP-models are developed. The first model targets the optimal assignment for newly arrived patients, whereas the second also considers future, but planned, arrivals. Both models are compared on an existing set of benchmark instances from the PA planning problem, which serves as the basic problem setting. These instances are then extended with additional parameters to study the effect of uncertainty on the patients' length of stay, as well as the effect of the percentage of emergency patients. The results show that the second model provides better results under all conditions, while still being computationally tractable. Moreover, the results show that pro-actively transferring patients from one room to another is not necessarily beneficial.

Keywords Patient assignment problem · Dynamic planning · Mixed Integer Linear Programming

1 Introduction

Rooms and beds belong to the critical assets of any hospital since they account for a considerable part of a hospital's infrastructure. A large amount of financial resources are invested in equipping them with medical apparatus to facilitate patient care. Moreover, they also represent the place where most patients will spend the largest part of their stay, as they recover

W. Vancroonenburg (✉) · G. Vanden Berghe
Department of Computer Science, CODeS/iMinds, KU Leuven, Gebroeders De Smetstraat 1, 9000 Gent, Belgium
e-mail: Wim.Vancroonenburg@cs.kuleuven.be

G. Vanden Berghe
e-mail: Greet.VandenBerghe@cs.kuleuven.be

P. De Causmaecker
Department of Computer Science, CODeS/iMinds, KU Leuven, Etienne Sabbelaan 53, 8500 Kortrijk, Belgium

from surgery or treatment, wait for examinations to take place, etc. In order to improve their comfort, patients are offered a choice between single bed rooms, luxury rooms with private showers, and other conveniences. As a result, many hospitals provide a large variety of rooms in terms of capacity, medical apparatus and amenities. Assigning patients can therefore be challenging, necessitating an efficient plan for making such assignments.

Bed managers and admission officers aim at finding a solution that strikes a balance between patients' preferences and comfort, and patients' clinical conditions and the resulting required room facilities. However, the availability of rooms and equipment needs to be considered, as well as hospital policies and standards, complicating decision making. A lack of overview on occupied beds and the uncertainty on how long patients will stay in the hospital, further complicate the matter.

Demeester et al. (2010) defined and studied the patient assignment (PA) problem in the context just described. They consider a set of patients, each with individual characteristics, who arrive at a hospital over a certain period of time. The hospital comprises a set of rooms, each with a given capacity and characteristics. The problem is to find an effective assignment of patients to rooms, satisfying room capacity restrictions. Moreover, a perceived cost is associated with each patient to room assignment relating to the *appropriateness* of that assignment (which may also depend on the assignment of other patients). The objective is to minimize the total cost of these assignments. The present contribution focuses on this problem.

1.1 Related work

As pointed out by Rais and Viana (2011) in their survey on operations research in health care, a great deal of the considered literature has focused on scheduling patients and hospital resources. Notably, nurse rostering, and operating theatre (OT) planning and scheduling have quite naturally received a considerable amount of attention, given that personnel and the OT are among the most expensive resources for any hospital (see e.g. Burke et al. 2004; Cardoen et al. 2010, for a review on these respective topics).

The PA problem as considered by Demeester et al. (2010) comprises an assignment problem that occurs at the operational level of hospital admission offices. It assumes that patients have already been attributed an admission date, a decision that is made as part of either an intervention scheduling process (see e.g. Guinet and Chaabane 2003; Riise and Burke 2010) during operational surgery scheduling, or an appointment/treatment scheduling process when no surgery is required. The type of patients and the arrival pattern of patients with different pathologies is largely influenced by the Master Surgery Schedule[1] (MSS), a (typically cyclic) timetable that allocates operating rooms and operating time to different medical disciplines. Much research has focused on techniques for reorganizing this MSS in order to change this arrival pattern. For example, Beliën and Demeulemeester (2007) show how a cyclic MSS can be constructed resulting in an expected levelled bed occupancy, by minimizing a weighted sum of the maximal expected bed occupancy and the maximal expected variance of the bed occupancy, over the considered period. The main idea is thus to construct MSSs for which the peak bed occupancy is minimal and also more constant. However, a MSS or block schedule is not the only way to partition the available operating theatre capacity. Fei et al. (2008) show how to develop an open scheduling strategy using a Column-Generation-Based Heuristic procedure.

[1] Also denoted as a *block scheduling* strategy.

Demeester et al. (2010) introduced the PA problem to the academic community as a challenging combinatorial optimization problem. In a follow up paper, Bilgin et al. (2012) presented a new hyper-heuristic algorithm to the PA problem, providing new benchmark instances and reporting test results. Vancroonenburg et al. (2011) showed that the PA problem is NP-hard.

The problem was also studied by Ceschia and Schaerf (2011), who developed a Simulated Annealing algorithm that improves on the best known results for the benchmark instances. Lower bounds for these instances are provided as well. More interestingly, they argue that the problem definition only assumes patients that are planned in advance (elective patients), and that it does not capture the dynamics of uncertainty on patient arrivals and departures. An extension to the problem definition is proposed where patient admission and discharge dates are revealed a few days before they occur (denoted as the *forecast level*). To this end, Ceschia and Schaerf developed a dynamic version of their algorithm that can be used for day-to-day scheduling. The performance of this algorithm is analysed under an increasingly larger forecast level. Ceschia and Schaerf (2012) continue their effort in developing a dynamic version of the PA problem formulation, also considering registration dates, the possibility of delaying patients, and minimizing the risk of room overcrowding.

The PA problem where patient transfers are not allowed, is related to the interval scheduling problem: patients can be represented by fixed length intervals (i.e. jobs with fixed start and end time) that need to be assigned to a machine (a room) for 'processing'. The PA problem comprises required jobs and non-identical machines with different capacities, the goal being to find a minimum-cost schedule subject to side-constraints. In a dynamic context, it constitutes an *online* interval scheduling problem with uncertainty on the interval lengths. In the case where patients are allowed to be transferred from one room to another, the problem can be seen as an interval scheduling problem which permits pre-emption of jobs. Kolen et al. (2007) provide a review on the subject of (online) interval scheduling problems. However, a critical difference is that the PA problem also includes costs which are directly related to sets of patients being assigned to the same room (gender conflicts, see Sect. 2), in contrast to the costs related to a single patient-room assignment. This already makes the problem hard (Vancroonenburg et al. 2011), for a single time-unit instance (which effectively drops the notion of intervals, as only one specific instant is being considered).

1.2 Contribution

The present study was motivated by the work in Ceschia and Schaerf (2011) and was developed in parallel to Ceschia and Schaerf (2012), who discuss the PA problem in a dynamic context. This study complements the work by Ceschia and Schaerf. We similarly define a new extension to the PA problem in a dynamic context. To this end, registration dates for each patient are added to the problem definition signaling a patient's possible future arrival time. This contrasts with the approach of defining an absolute forecast level (Ceschia and Schaerf 2011), which assumes that all patient arrivals within the forecast level are known. However, such an approach does not allow accurate modelling of emergency patients.

Moreover, the present study makes a more general assumption on the patients' length of stay (LOS). Only the availability of an estimate on each patients' *length of stay* is assumed, which in practice is often the case (either by historical data, or the physician's estimate). However, this requires to make adjustments to previous decisions when patients outstay their estimated length of stay (and thus room-assignment collisions occur). This contrasts with the work by Ceschia and Schaerf (2012), that models this issue as a static overcrowd risk that should be avoided. The effect of replanning patients on the solution quality is not considered. The question thus remains how to address this, should it occur. In our study, special care is taken to accommodate this specific decision process.

This dynamic version of the problem is modelled and solved using Integer Linear Programming (ILP). The performance of this approach is discussed and the effect of the percentage of emergency cases and the accuracy of the LOS estimate is studied. It is shown that taking into account information on future, but registered, arrivals allows for improved decision making, even in the presence of emergency arrivals and inaccurate LOS estimates.

2 Problem formulation

2.1 Patient-to-room assignment in a static context

The PA problem as described by Demeester et al. (2010) considers a set of patients P that each need to be assigned to one of a set of hospital rooms R over a certain time horizon $H = \{1, \ldots, T\}$. Each room $r \in R$ has a given capacity, denoted by $c(r)$. Each patient $p \in P$ is attributed an arrival time $a(p)$ and a departure time $dd(p)$, with the time interval $H_p = \{t \in H : a(p) \le t < dd(p)\}$ representing the patient's stay in the hospital. The length of the patient's stay, $dd(p) - a(p) = |H_p|$, is denoted as $los(p)$.

The problem is to find an assignment $\sigma : P \times H \mapsto R$ of patients to rooms, for each time unit of their stay, that minimizes a certain cost $w(\sigma)$ related to these assignments. This cost $w(\sigma)$ consists of three parts:

– each patient/room combination is attributed a cost $c(p, r)$, that relates to the *appropriateness* of assigning patient p to room r for one time unit (the lower $c(p, r)$, the better). The goal is to minimize the sum of these assignment costs.

$$\text{Min } w_1(\sigma) = \sum_{p \in P} \sum_{t \in H_p} c\big(p, \sigma(p, t)\big) \tag{1}$$

– the number of *gender conflicts* in all rooms r over the entire planning horizon H. The goal is to avoid that male and female patients are assigned to the same room at the same time.

$$\text{Min } w_2(\sigma) = \sum_{r \in R} \sum_{t=1}^{T} \mathit{Conflict}_{\sigma r t} \tag{2}$$

These conflicts are calculated as follows:

$$\mathit{Conflict}_{\sigma r t} = \min\big(|p \in P_{\sigma r t} : p \text{ is } \mathit{male}|, |p \in P_{\sigma r t} : p \text{ is } \mathit{female}|\big) \tag{3}$$

with

$$P_{\sigma r t} = \big\{p \in P : a(p) \le t < dd(p), \sigma(p, t) = r\big\} \tag{4}$$

denoting the set of patients assigned to room r at time t.

– the number of patient transfers from one room r to another r' should also be minimized:

$$\text{Min } w_3(\sigma) = \sum_{p \in P} \sum_{t \in H_p \setminus \{a(p)\}} \mathit{Transfer}_{\sigma p t} \tag{5}$$

with

$$Transfer_{\sigma pt} = \begin{cases} 1 & \text{if } \sigma(p, t-1) \neq \sigma(p, t), \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

The complete objective can then be expressed as follows:

$$\text{Min } w(\sigma) = w_1(\sigma) + w_G \cdot w_2(\sigma) + w_{Tr} \cdot w_3(\sigma) \tag{7}$$

with w_G, w_{Tr} weights denoting the relative importance of gender conflicts and transfers. Finally, the assignment σ should respect the room capacities at all times, i.e.:

$$\forall t = 1, \ldots, T,\ r \in R\colon \quad |P_{\sigma rt}| \leq c(r) \tag{8}$$

2.2 Extension to an online, dynamic context

In practice, the arrivals and departures of patients are gradually revealed over the planning horizon. The present contribution therefore extends the problem definition to account for these dynamics. Each patient p is attributed a *registration* date $r(p)$, at which point the patient becomes known to the system, and an *expected* departure date, $ed(p)$, which is an estimate of the patient's departure date. The actual departure date of the patient, $dd(p)$, however remains hidden until it has passed.

The dynamic problem definition requires a new problem to be solved at each $t' \in H$ where the following information is available:

- $P_{t'}$: the set of patients with $r(p) = t'$, i.e. the patients that are registered at time t'. At this point, only $a(p)$ and $ed(p)$ are known for each patient $p \in P_{t'}$, $dd(p)$ remains hidden.
- $DP_{t'}$: the set of patients with $dd(p) = t'$, i.e. the patients that leave the hospital at time t'.

as well as the information in $P_1, P_2, \ldots, P_{t'-1}$ and $DP_1, DP_2, \ldots, DP_{t'-1}$. Let $A_{t'}$ denote the set of patients that arrived at t', i.e. :

$$A_{t'} = \left\{ p \in P : a(p) = t' \right\} \tag{9}$$

The goal of the problem is to find at each time t' an assignment

$$\sigma_{t'} : \left(\bigcup_{i=1}^{t'} A_i \right) \times \left\{ 1, \ldots, t' \right\} \mapsto R \tag{10}$$

that maps each *arrived* patient p (i.e. all p for which $t' \geq a(p)$) to a hospital room r, for each time unit of their individual stay up till t'. Obviously, the assignment $\sigma_{t'}$ should still respect room capacity at all times. The following condition should also hold:

$$\forall p \in \left(\bigcup_{i=1}^{t'-1} A_i \right), t \in \left[a(p), t'-1 \right] : \quad \sigma_{t'}(p, t) = \sigma_{t'-1}(p, t) \tag{11}$$

i.e. the new assignment σ_t' should respect decisions made at times $1, \ldots, t'-1$.

The assignment σ_T denotes the solution at the end of the planning horizon. It contains all the patients' assignments within that period. The solution quality can be assessed by computing $w(\sigma_T)$, which is again the sum of patient-to-room assignment costs, gender conflicts over the entire planning horizon, and finally patient transfer penalties. It is interesting to compare this value with the quality obtained for the static variant of the problem, which assumes that each patient's departure date is fixed in advance. Any lower bound for the static version is a lower bound for the dynamic problem, and thus is an indication of what can be achieved when all information is known *a priori*.

3 Optimization models

We developed two models for the dynamic PA planning problem that correspond with the decision that must be made at time t', that is: give a new assignment of patients to rooms considering the current situation. The first model is modelled after current practice, namely the assignment decision is made shortly before patient arrival and only current room availability is considered. The model tries to find the optimal assignment for the patients who arrived at t'. Moreover, it uses the estimate of the newly arrived patients' LOS. If any patient stays longer than expected (i.e. $t' \geq ed(p)$ and $p \notin DP_1, DP_2, \ldots DP_{t'}$), it is assumed that the patient stays at least one time unit longer.

The second model builds on the previous model by also considering all registered patients at each t', therefore anticipating future occupancy and room demand. Both models are implemented as Mixed Integer Linear Programming models. They are described in Sects. 3.1 and 3.2. In order to simplify the description, the following notation will be used:

- $\mathbf{P}_{t'} = \bigcup_{i=1}^{t'} P_i \setminus \bigcup_{i=1}^{t'} DP_i$, the set of all registered patients, that have not yet left the hospital, up to (and including) t',
- $\mathbf{A}_{t'} = \bigcup_{i=1}^{t'} A_i \setminus \bigcup_{i=1}^{t'} DP_i$, the set of all arrived patients, that have not yet left the hospital, up to (and including) t',
- superscript M, F, restrict a set of patients P to either *males* or *females* respectively,
- $elos(p) = \max(ed(p), t'+1) - \max(a(p), t')$, the remaining, expected length of stay of patient p as it is known at decision time t'. If the patient's stay has exceeded his or her expected departure date $ed(p)$, he or she is expected to stay at least one time unit longer.
- $\mathbf{AP}_{tt'} = \{p \in \mathbf{A}_{t'} : t' \leq t < \max(ed(p), t'+1)\}$, the set of *arrived* patients that are *expected* to be present at time t $(t \geq t')$,
- $\mathbf{PP}_{tt'} = \{p \in \mathbf{P}_{t'} : t' \leq t < \max(ed(p), t'+1)\}$, the set of *registered* patients that are *expected* to be present at time t $(t \geq t')$.
- $Transfer_{t'pr} = \begin{cases} 1 \text{ if } \sigma_{t'-1}(p) \neq r, \\ 0 \text{ otherwise.} \end{cases}$
- $MaxClique_{t'}(P)$, the set of subsets of P, whose intervals, starting from t', form maximal cliques in the corresponding interval graph. We refer to Sect. 3.1.1 for more information.

3.1 Model 1: reactive assignments

The decision variables are defined as follows:

$$x_{p,r} = \begin{cases} 1 & \text{if patient } p \text{ is assigned to room } r, \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

$$v_{r,t} = \text{the number of } \textit{gender} \text{ conflicts in room } r \text{ at time } t \tag{13}$$

$$y_{r,t} = \begin{cases} 1 & \text{if the number of } \textit{male} \text{ patients assigned to room } r \\ & \text{at time } t \text{ is larger than or equal to the number of } \textit{female} \text{ patients,} \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

The optimization problem is then modelled as follows:

$$\text{Min} \sum_{p \in \mathbf{A}_{t'}} \sum_{r \in R} \left(elos(p) \cdot c(p,r) + w_{Tr} \cdot Transfer_{t'pr} \right) \cdot x_{p,r} + \sum_{r \in R} \sum_{t=t'}^{T} w_G \cdot v_{r,t} \tag{15}$$

$$\text{s.t.} \sum_{r \in R} x_{p,r} = 1 \quad \forall p \in \mathbf{A}_{t'} \tag{16}$$

$$\sum_{p \in P_c} x_{p,r} \leq c(r) \quad \forall r \in R,\, P_c \in MaxCliques_{t'}(\mathbf{A}_{t'}) \tag{17}$$

$$\sum_{p \in \mathbf{AP}^M_{tt'}} x_{p,r} \geq v_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{18}$$

$$\sum_{p \in \mathbf{AP}^F_{tt'}} x_{p,r} \geq v_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{19}$$

$$\sum_{p \in \mathbf{AP}^M_{tt'}} x_{p,r} \leq v_{r,t} + c(r) \cdot y_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{20}$$

$$\sum_{p \in \mathbf{AP}^F_{tt'}} x_{p,r} \leq v_{r,t} + c(r) \cdot (1 - y_{r,t}) \quad \forall r \in R, t = t', \ldots, T \tag{21}$$

$$x_{p,r} \in \{0, 1\} \quad \forall p \in \mathbf{A}_{t'}, r \in R$$

$$v_{r,t} \geq 0 \quad \forall r \in R, t = t', \ldots, T$$

$$y_{r,t} \in \{0, 1\} \quad \forall r \in R, t = t', \ldots, T$$

The model describes an assignment problem minimizing the expected cost of the newly arrived patients (Expression (15)). Constraint (16) specifies that each arrived patient has to be assigned to a room, while constraint (17) expresses that room capacity should be respected for all maximal cliques in the interval graph corresponding to $\mathbf{A}_{t'}$ (see Sect. 3.1.1 for more information). Constraints (18), (19), (20), and (21) relate the variables $v_{r,t}$ and $y_{r,t}$, forcing $v_{r,t}$ to take on the expected value of the minimum number of either males or females in room r at time t.

3.1.1 Using maximal cliques for room capacity

At any given time t, the room capacity constraint needs to be respected. A straightforward way to implement this constraint is to add the following expression to the model:

$$\sum_{p \in \mathbf{AP}_{tt'}} x_{p,r} \leq c(r) \quad \forall r \in R, t = t', \ldots, T \tag{22}$$

However, this is a fairly inefficient way of implementing this constraint as the following example shows. Consider the patient intervals (patient stays) shown in Fig. 1. For any given room r with capacity $c(r)$, the following constraints would be imposed using formulation (22):

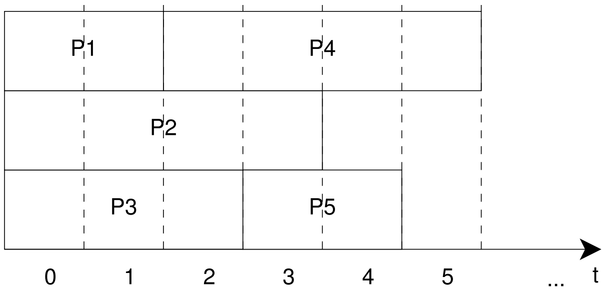

Fig. 1 An example showing five patient intervals (patient stays) for which the capacity should be enforced. Implementing capacity constraints for this situation can be done more efficiently by generating a constraint for the max-clique of the corresponding interval graph, rather than imposing a capacity constraint for $t = 1, 2, \ldots, 6$

$$x_{1,r} + x_{2,r} + x_{3,r} \leq c(r) \quad t = 0 \tag{23}$$

$$x_{1,r} + x_{2,r} + x_{3,r} \leq c(r) \quad t = 1 \tag{24}$$

$$x_{2,r} + x_{3,r} + x_{4,r} \leq c(r) \quad t = 2 \tag{25}$$

$$x_{2,r} + x_{4,r} + x_{5,r} \leq c(r) \quad t = 3 \tag{26}$$

$$x_{4,r} + x_{5,r} \leq c(r) \quad t = 4 \tag{27}$$

$$x_{4,r} \leq c(r) \quad t = 5 \tag{28}$$

It is clear that (23)–(24) are identical, and (27) and (28) are already implied by (26). A more efficient way of implementing this constraint would be as follows:

1. Construct an *interval graph* based on the intervals from a given subset of patients P'. An interval graph is a graph $G(V, E)$ where each vertex $v \in V$ corresponds to an interval (in this case the interval of a patient $p \in P'$). The edge set is given by $E = \{\{v_1, v_2\} | v_1, v_2 \in V \wedge v_1 \text{ overlaps } v_2\}$, i.e. there is an edge between two vertices v_1, v_2 if their corresponding intervals overlap. Figure 2 shows the interval graph corresponding to the example in Fig. 1.
2. Enumerate all *maximal cliques* from this interval graph. A clique C is a subset of vertices ($C \subseteq V$), such that $\forall v_i, v_j \in C \Rightarrow \{v_i, v_j\} \in E$, i.e. a clique is a subset of nodes which are pairwise directly connected. The maximal cliques of G are all cliques C for which $C \cup \{v'\}$, with $v' \in V \backslash C$, does not form a clique. That is, maximal cliques[2] are cliques of maximal cardinality and can not be expanded by adding any node from V not in C. The maximal cliques of the interval graph in Fig. 2 are $\{p_1, p_2, p_3\}$, $\{p_2, p_3, p_4\}$ and $\{p_2, p_4, p_5\}$.
3. For each room $r \in R$, and each maximal clique P_c (corresponding to a clique C in the interval graph), construct a capacity constraint:

$$\sum_{p \in P_c} x_{p,r} \leq c(r) \tag{29}$$

In the example, the following constraints are constructed:

$$x_{1,r} + x_{2,r} + x_{3,r} \leq c(r) \quad \text{for clique } \{p_1, p_2, p_3\} \tag{30}$$

$$x_{2,r} + x_{3,r} + x_{4,r} \leq c(r) \quad \text{for clique } \{p_2, p_3, p_4\} \tag{31}$$

[2]Note that the notion of a maximal clique differs from a *maximum* clique. A *maximum* clique is the largest cardinality clique that can be found in a graph.

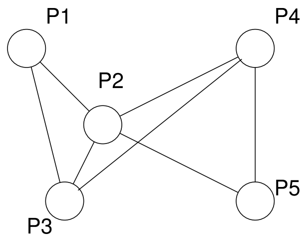

Fig. 2 The interval graph corresponding to the intervals in Fig. 1

$$x_{2,r} + x_{4,r} + x_{5,r} \leq c(r) \quad \text{for clique } \{p_2, p_4, p_5\} \tag{32}$$

which are identical to resp. (23), (25) and (26)

Consider these maximal cliques to correspond with the maximal subsets of patients from P' present in the hospital at any given time. Thus, to enforce that room capacity is respected, it is sufficient to enforce that for each of these subsets no more than $c(r)$ patients may be assigned to a room r (and this for all $r \in R$). As these cliques are maximal, any smaller clique of these patients that may be present at a later time, is already implied by the constraint for the maximal clique.

Enumerating the maximal cliques of an interval graph can be done in polynomial time. Such an algorithm is described by Krishnamoorthy et al. (2012), which is the algorithm that was implemented for this work. We truncate the start of the intervals, corresponding to patients in P', such that the decision time t' is the minimal time considered. That is:

$$\forall p \in P' : H'_p = \left\{t \in H : \max\left(a(p), t'\right) \leq t < dd(p)\right\} \tag{33}$$

In this way, the capacity constraint is checked only from t' onwards, and thus ensures that the resulting constraints properly allow transfers of patients.

The maximal clique model requires fewer constraints than the former one. We compared the two formulations (with and without the maximal clique version of the capacity constraint) of this model on a set of instances, based on instance 5 of the test set (please refer to Sect. 4 for more details on the experimental setup). In this setting, the total number of constraints generated by the model (for solving the first decision problem, at $t' = 0$) was reduced from 5939 to 4885 (averaged over 300 models generated), a reduction by 17.7 %. However, it must be noted that after the *presolve* phase of the ILP solver (CPLEX 12.4 was used for the experiments), which tries to identify redundant constraints, the final constraint set for both formulations was reduced to the same number of constraints. Instead of relying on ILP solver capabilities, we prefer to present the most efficient formulation: the maximal clique formulation.

3.2 Model 2: anticipatory assignments

The second model defines the same decision variables as Model 1, but it differs in the set of patients for which they are defined. The $x_{p,r}$ variables are defined for all arrived patients $\mathbf{A}_{t'}$ in the first model, whereas they are defined for all *registered* patients $\mathbf{P}_{t'}$ in the new model. Another difference is that patients can be assigned to a *dummy* room, denoted as $\perp$. Only registered patients who have not arrived ($p \in \mathbf{P}_{t'} \setminus \mathbf{A}_{t'}$) are allowed in this dummy room, so as to ensure feasibility of the model under an expected, future, undercapacity. These assignments are attributed a high cost $c(p, \perp)$ in such a way that the model gives priority to a real assignment for each future arrival. Finally, the model also does not require that the assignment for registered, not-arrived patients takes on an integer value. Therefore, the model takes into account a *lower bound* on the assignment cost for these patients, which speeds up calculations while still allowing for an informed decision on the current assignments.

The model is defined as follows:

$$\text{Min} \sum_{p \in \mathbf{P}_{t'}} \sum_{r \in R \cup \perp} \left(elos(p) \cdot c(p,r) + w_{Tr} \cdot Transfer_{t'pr} \right) \cdot x_{p,r}$$

$$+ \sum_{r \in R} \sum_{t=t'}^{T} w_G \cdot v_{r,t} \tag{34}$$

$$\text{s.t.} \sum_{r \in R} x_{p,r} = 1 \quad \forall p \in \mathbf{A}_{t'} \tag{35}$$

$$\sum_{r \in R \cup \perp} x_{p,r} = 1 \quad \forall p \in \mathbf{P}_{t'} \backslash \mathbf{A}_{t'} \tag{36}$$

$$\sum_{p \in P_c} x_{p,r} \le c(r) \quad \forall r \in R, P_c \in MaxCliques_{t'}(\mathbf{P}_{t'}) \tag{37}$$

$$\sum_{p \in \mathbf{PP}^M_{tt'}} x_{p,r} \ge v_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{38}$$

$$\sum_{p \in \mathbf{PP}^F_{tt'}} x_{p,r} \ge v_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{39}$$

$$\sum_{p \in \mathbf{PP}^M_{tt'}} x_{p,r} \le v_{r,t} + c(r) \cdot y_{r,t} \quad \forall r \in R, t = t', \ldots, T \tag{40}$$

$$\sum_{p \in \mathbf{PP}^F_{tt'}} x_{p,r} \le v_{r,t} + c(r) \cdot (1 - y_{r,t}) \quad \forall r \in R, t = t', \ldots, T \tag{41}$$

$$x_{p,\perp} = 0 \quad \forall p \in \mathbf{A}_{t'} \tag{42}$$

$$x_{p,r} \in \{0,1\} \quad \forall p \in \mathbf{A}_{t'}, r \in R \cup \perp$$

$$0 \le x_{p,r} \le 1 \quad \forall p \in \mathbf{P}_{t'} \backslash \mathbf{A}_{t'}, r \in R \cup \perp$$

$$v_{r,t} \ge 0 \quad \forall r \in R, t = t', \ldots, T$$

$$y_{r,t} \in \{0,1\} \quad \forall r \in R, t = t', \ldots, T$$

The objective of the model, expression (34), is again to minimize the total assignment cost, including minimizing any possible dummy assignments. Constraints (35) and (36) specify that each arrived and registered patient should be assigned to one room, allowing for dummy assignments for future arrivals. Constraints (37)–(41) are similar to their counterparts in Model 1, this time also considering future arrivals. Constraint (42) ensures that arrived patients are not assigned to dummy rooms.

Table 1 Problem characteristics of the instances

instance	$\|P\|$	$\|R\|$	$\sum_{r\in R} c(r)$	avg. occupancy (%)	planning horizon
1	652	98	286	59.69	14
5	587	102	325	49.32	14
8	895	148	441	43.90	21
10	1575	104	308	47.76	56

4 Experimental setup

We tested the sensitivity of the two models to the following problem characteristics:

- Occupancy
- Accuracy of the length of stay estimate
- Emergency versus planned cases

Also, we investigated whether or not allowing patient transfers in the model has a large impact on the final solution quality of the algorithms. Therefore, we tested two versions of both the reactive model (Model 1) and the anticipatory model (Model 2): one version corresponds to the previously described models (see Sect. 3) that allow transfers, whereas the second version does not allow transfers. The latter implies that the second model fixes the assignment of a patient after arrival, or mathematically:

$$x_{p,r} = 1 \quad \forall p \in \mathbf{A}_{t'-1} \cap \mathbf{A}_{t'}, r = \sigma_{t'-1}(p) \tag{43}$$

For the purpose of experimentation, we used a subset of the benchmark instances for the static PA problem available from the patient admission scheduling website (Demeester 2012). The instances served as the basic problem setting to test these models, which were then extended to test for the above mentioned factors.

The instances were extended to the dynamic problem by adding a random registration date $r(p)$ and an expected departure date $ed(p)$ for each patient p over the planning horizon. We refer to Table 1 for the characteristics of these instances. The weights $c(p,r)$ are based on the decision rules (is a patient assigned to a department with the correct specialism, are their room preferences met, etc.) discussed by Demeester et al. (2010) and detailed by Demeester (2012). The weights w_G and w_{Tr} are set to 5.0 and 11.0, corresponding to the weights used by Demeester et al. However, note that in the implementation/testing of the model, these weights have been multiplied by 10 in order to obtain an integer representation, rather than a decimal representation (Demeester et al. use weights with accuracy up to 1 decimal place).

The procedure for enriching the instances is as follows:

- $ed(p)$ is selected uniformly from the interval $[dd(p) - acc, dd(p) + acc]$ for each patient individually. If $ed(p) <= a(p)$, then it is set to $ed(p) = a(p) + 1$. We investigated the effect of *acc*, i.e. the effect of the accuracy of the expected departure date estimate.
- $r(p)$ is either selected uniformly from the interval $[a(p) - T, a(p) - 1]$ for planned patients, or is set to $a(p)$ for emergency patients. We investigated the effect of the percentage (denoted *em*) emergency versus planned cases.

Both models (and both the transfer and non-transfer versions) were tested on all combinations of the factors *acc* (LOS estimate) and *em* (percentage emergency cases), with *acc* ranging from 0 time units (perfect estimate) to 5 time units (a poor estimate) and $em \in \{0, 0.25, 0.50, 0.75, 1.0\}$. All tests were performed on 10 randomized instances for each specified combination of the mentioned factors.

The effect of increasing occupancy was tested by randomly removing beds (uniformly selected) from the instances in order to increase the projected average occupancy. This procedure is similar to what is done in Ceschia and Schaerf (2011). Feasibility is maintained by limiting the peak occupancy to 100 %. For studying the effect of increasing occupancy, we have calculated the lower bound for every occupancy setting since the lower bound increases as beds are removed from the instance. In the figures discussed in the following section, this lower bound is denoted as LB.

The ILP models have been implemented using CPLEX 12.4 (IBM ILOG 2012) with a free academic license.The computations were conducted on the infrastructure of the VSC—Flemish Supercomputer Center, funded by the Hercules foundation and the Flemish Government—department EWI. All experiments were performed on computers equipped with an 8 core, 2.8 GHz Xeon X5560 (Nehalem) processor, and 24 GB of ram memory, running a GNU/Linux operating system. The supporting code was implemented in Java 1.7. The CPLEX 12.4 solver was configured to use only one processing thread, enabling us to test up to 8 instances in parallel on one machine and as such to reduce computation time. In total, 9 machines from the Flemish Supercomputer Center were used for these tests, reducing the total computation time 72-fold. The overall average computation time (over all instances/experiments) for a complete run of one instance was 4.80 minutes, with a maximum computation time of 227.15 minutes. Although powerful computer resources were used for these computer tests, the single core performance of such a machine (used for one complete run of an instance) is comparable to a high-end desktop computer available in 2009. Similar computation times can thus be expected on recent consumer hardware. All results and graphics were processed with the R software environment for statistical computing (R Core Team 2012).

5 Discussion

5.1 Emergency versus planned cases, and the effect of the LOS estimate

Figure 3 shows the effect of an increasing percentage of emergencies on the value $w(\sigma_T)$, the value of the solution obtained at the end of the planning horizon. This has been averaged over all runs and parameter settings of the LOS estimate. The results show that the anticipatory models (Model 2) consistently outperform the reactive models (Model 1), for all problem instances: for each setting of the parameter *em*, the cost value $w(\sigma_T)$ is lower for the anticipatory models than for the reactive models. In the limit for increasing percentage of emergencies, the result of the anticipatory models converge to those of the reactive models. Obviously, in the case for 100 % emergency cases, no future arrivals can be planned and the anticipatory models reduce to the reactive models. Due to an excessive computation time, the setting $em = 0$ % is not shown for instances 5, 8 and 10. To avoid complicating interpretation of the results, no time limit was set to ensure the models are solved to optimality at each t'.

Instance 1 (Fig. 3, topleft), shows a clear advantage to allow transfers in the reactive model, while there is no clear advantage to allow them in the anticipatory models. For instance 5, 8 and 10, the results lie differently. Although the anticipatory models still outperform the reactive models, the reactive model that does allow transfers now clearly performs

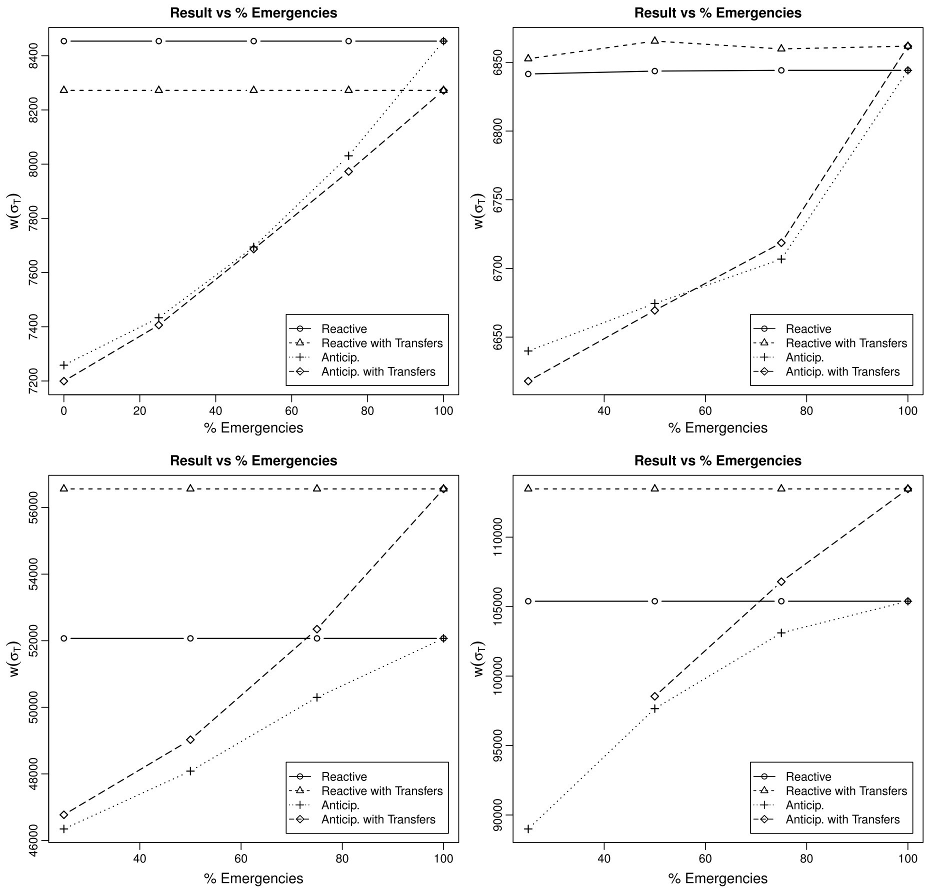

Fig. 3 Model performance for increasing percentage of emergencies. Results shown for instances 1 (*topleft*), 5 (*topright*), 8 (*bottomleft*) and 10 (*bottomright*)

worse than the version that does not. Although seemingly counter intuitive, namely allowing more flexibility causing a worse performance, this can be explained by the dynamics of the problem. For example, consider that at time t' it might appear beneficial to transfer one or several patients in order to obtain a better overall bed assignment. Then, at time $t' + \Delta t$ some patients not known at time t' might arrive who now obtain a much poorer assignment due to the assignments made at time t. Thus, both the transfer cost at time t' is incurred, as well as the assignment cost at time $t' + \Delta t$.

Figure 4 shows the average number of occurrences of transfers in σ_T for an increasing percentage of emergencies, for instances 1 and 5. It is clear that transfers do occur when enabling it in the models. The anticipatory model clearly avoids transfers more than the reactive model, as it has more information on the planned arrivals.

Figure 5 shows the effect of the models' performance under increasingly poorer LOS estimates, for instances 1, 5, 8 and 10. The results show that the performance of both the reactive models (Model 1) and the anticipatory models (Model 2) deteriorates for an increasingly poorer LOS estimate, while the anticipatory models always outperform the reactive models.

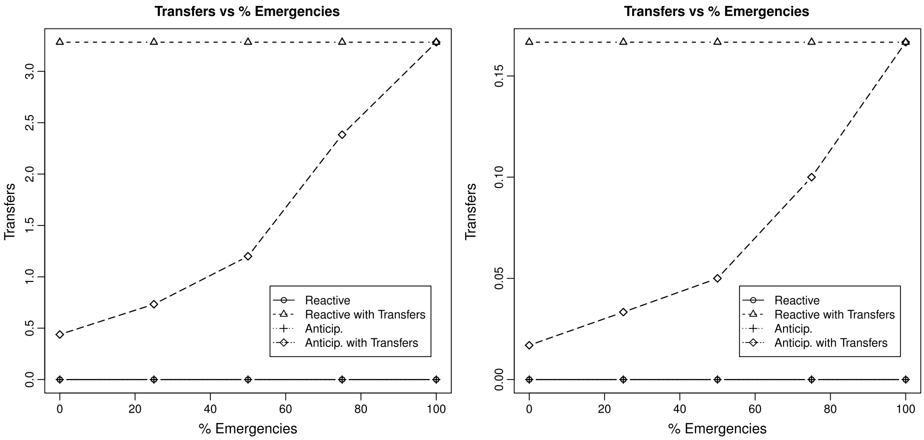

Fig. 4 Average # transfers for increasing percentage of emergencies. Results shown for instances 1 (*left*) and 5 (*right*)

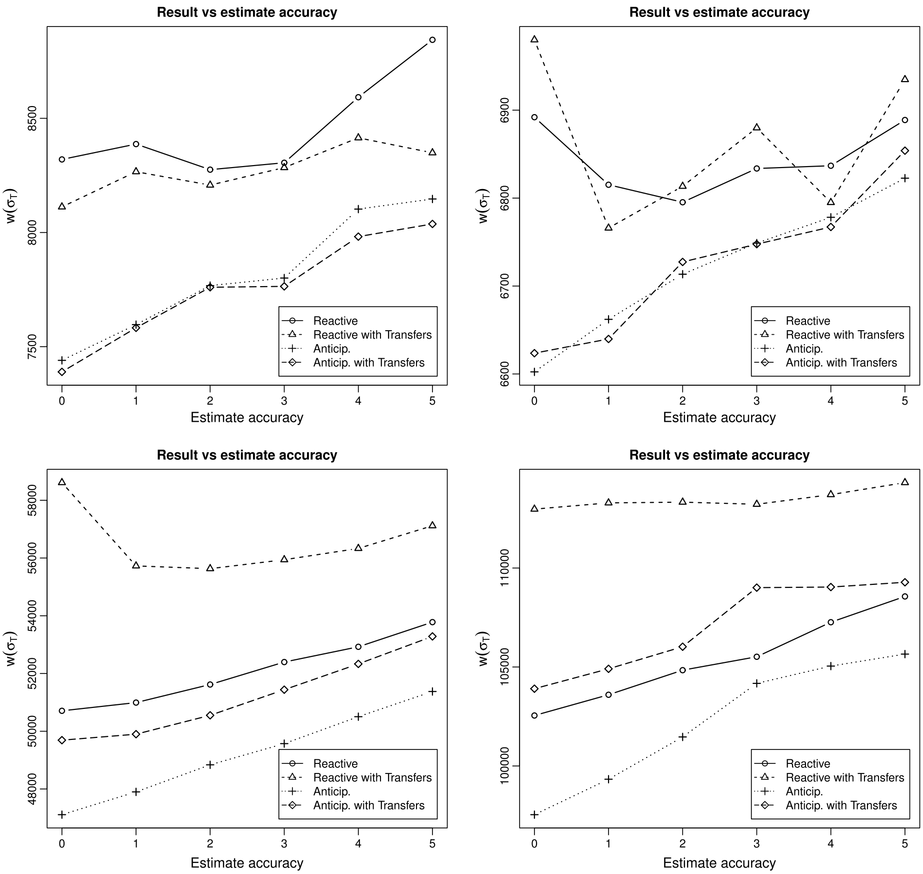

Fig. 5 Model performance for an increasingly poorer LOS estimate. Results shown for instances 1 (*topleft*), 5 (*topright*), 8 (*bottomleft*) and 10 (*bottomright*)

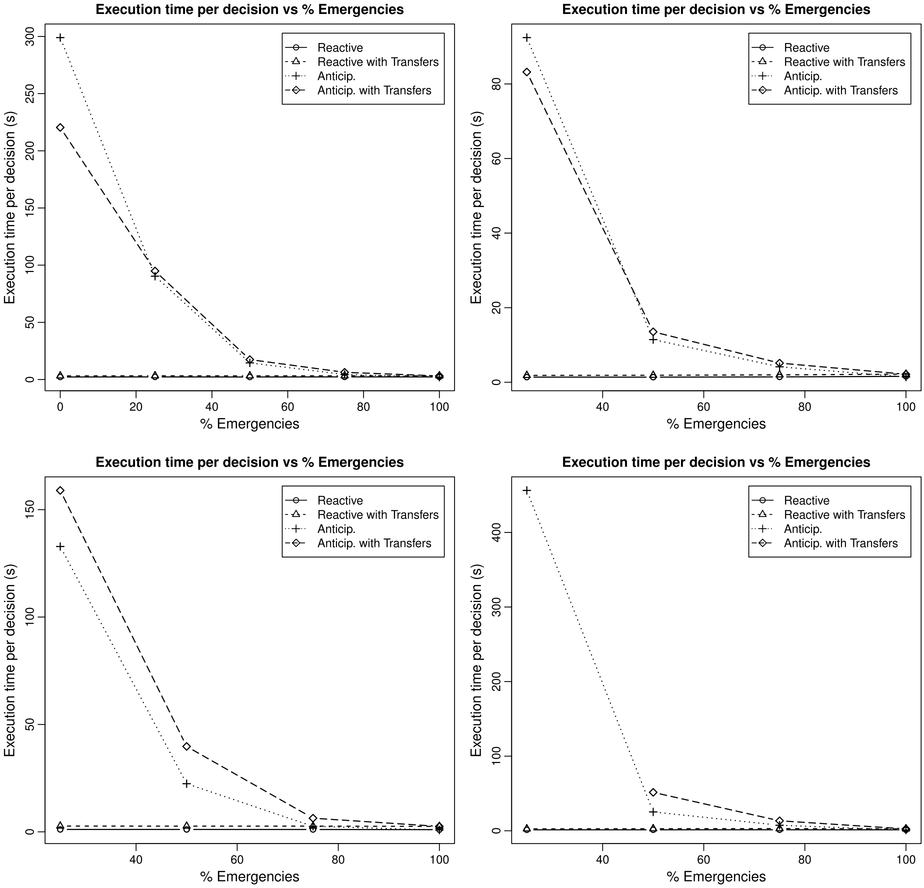

Fig. 6 Average execution time for increasing percentage of emergencies. Results shown for instances 1 (*topleft*), 5 (*topright*), 8 (*bottomleft*) and 10 (*bottomright*)

This result is expected, as an increasing inaccuracy of the LOS estimate causes inaccurate weighing of the patient assignments and thus generates suboptimal solutions.

Furthermore, the graphs show a more erratic behaviour, which appears unrelated to the percentage of emergencies. The reason for this behaviour is that a decision (both for the anticipatory and reactive models) may turn out good or bad when patients depart earlier or later than estimated. However, the overall trend is a decreasing performance for all models.

Another important factor that also needs to be considered is the execution time required to solve the assignment problem at each t'. Figure 6 shows the average execution time for the different models, with respect to the percentage of emergency arrivals. It is clear that the anticipatory models require more time to solve the problem at each t', and this difference becomes larger as the percentage of emergencies decreases. This is of course expected, because the anticipatory models require more variables as the number of elective patients increases. Furthermore, it is clear that the models which do allow patient transfers require more computation time. Not allowing patient transfers constrains the model a lot more (resulting in many variables being removed in the MIP solver presolve phase). We can also report several outliers, in terms of execution time, for the anticipatory models, where the

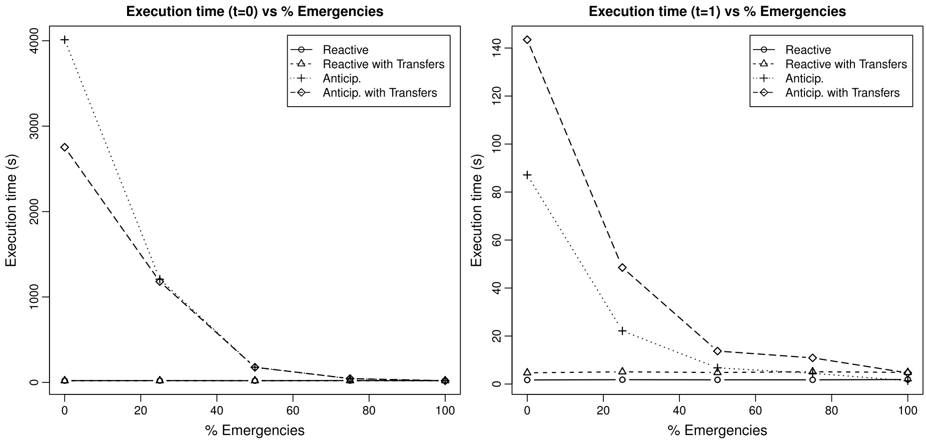

Fig. 7 Average execution time for increasing percentage of emergencies at $t' = 0$, (*left*) and at $t' = 1$ (*right*)

MIP solver would take a very long time (exceeding 1 hour of computation time for solving the MIP at a specific t'). However, this always occurs at time $t' = 0$ and for $em = 0$ %, which is the initial decision problem and the case of no emergencies. In this case, a great number of patients are taken into account (all patients registered before $t' = 0$), often more than half of the patients considered over the complete time horizon. For subsequent t', the execution time is much lower, as the MIP solver can make use of the previous solution for a *warm start* (i.e. in this case MIP solver heuristics can quickly produce a very good initial feasible incumbent, which often speeds up the branch-and-bound phase of the search). This is clearly shown in Fig. 7, which compares the execution time with respect to the percentage of emergency arrivals, for $t' = 0$ (left) and $t' = 1$ (right).

5.2 Effect of increasing occupancy

The effect of an increasing occupancy was tested by artificially forcing a higher, average, occupancy in instances 1 and 5, ranging from 59 % to 77 % for instance 1 and from 49 % to 67 % for instance 5. Both instances reach a peak occupancy of 100 %. Again, all combinations of factors were tested 10 times to reduce random effects. The following results report on the averages of those 10 runs.

Figure 8 shows the effect of an increasing occupancy on the performance of both models, under an increasing percentage of emergencies (from left-to-right, top-to-bottom) for instance 1. It is clear that both models perform worse under an increasing occupancy. However, the lower bounds of the instances also increase as beds are removed from the instances. Thus, the relative performance of the models compared to the lower bound does not change, indicating that occupancy does not have an effect on the relative behaviour of both models.

6 Conclusion and future perspectives

The contribution of the present paper is a dynamic version of the patient assignment problem, as defined by Demeester et al. (2010). The work extends the existing patient assignment (PA) problem definition from an offline setting to an online one. The problem formulation

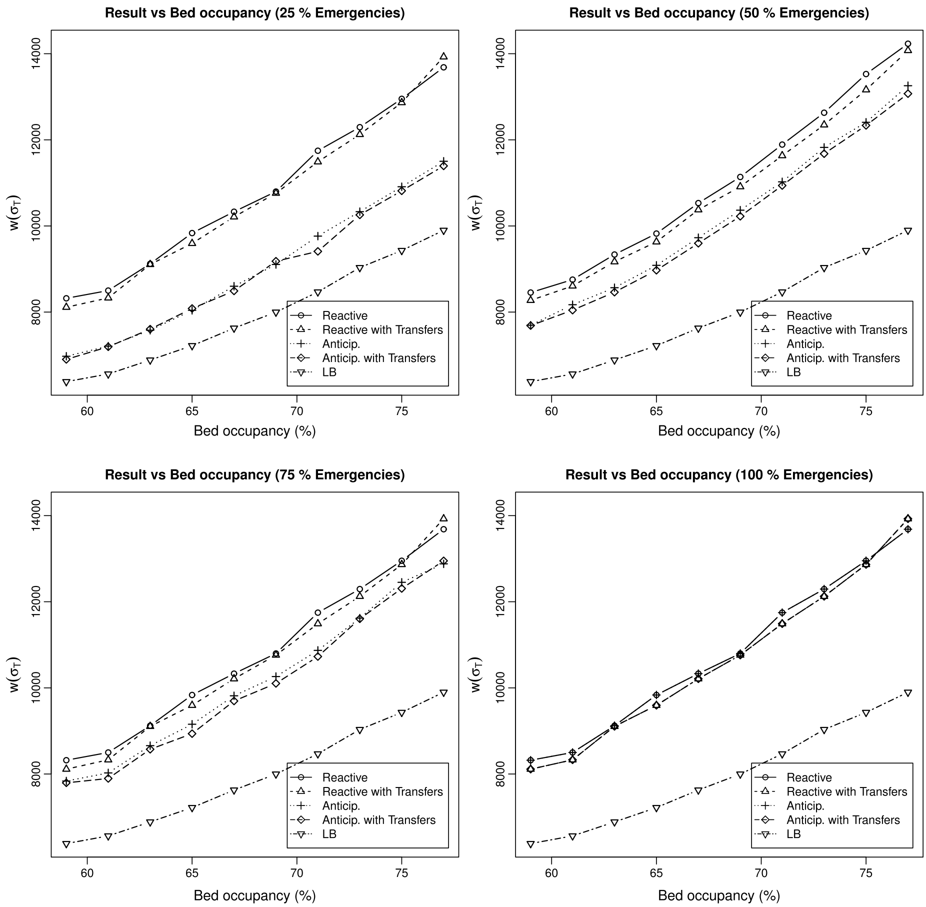

Fig. 8 Model performance for an increasingly higher occupancy rate, under different levels of emergency vs planned patients. Results shown for instance 1 with a perfect estimate

accounts for the dynamics of *online* patient arrivals, including emergency patients, and explicitly models the length of stay (LOS) of a patient as an *estimate*. This definition clearly maps more closely to the current practice of hospital admissions, where often only 50 percent of the patients are electively planned and patients' LOS are not known *a priori*.

Two ILP models were developed that aid in decision making when new patients arrive: one that is modelled after current practice, namely assigning patients to rooms as they arrive, and one that also accounts for planned arrivals. The first model improves on current practice by also considering the expected LOS of patients, therefore enabling proper weighing of patient assignments. Furthermore, this model is solved to optimality (with respect to the data provided), whereas in practice hospital admission officers employ rule-of-thumb heuristics, and often do not account for the LOS of patients, leading to suboptimal solutions. The second model also accounts for future, but planned, arrivals in order to weigh patient assignments even better. It does so by including a lower bound on the future arrivals, based on a *relaxed* assignment of the corresponding patient intervals.

Experimental results showed that the second model yields a better global result than the first model, due to it considering more available information on future arrivals. In addition, this model can still be solved in a reasonable amount of time using a commercial MIP solver in most practical cases (percentage of emergencies larger than 25 percent). Experimentation with the percentage of emergency patients, poorer LOS estimates and an increasing hospital occupancy indicate that this behaviour does not change under these conditions, advocating the use of the anticipatory model over the reactive one independently of the various factors considered. Lastly, allowing patients to be transferred from one room to another is not necessarily beneficial both in terms of computation time and computational result.

The research results are being prepared for practical application. The authors envision the presented models to be applied in a central, hospital wide admission system, supporting bed managers and admission officers in their daily task of assigning patients to hospital rooms. The models can be used each day to plan the room assignments for the patient arrivals of the upcoming day, and to replan room assignments on the next day, considering unplanned, urgent/emergency arrivals. In this setting, the necessary computational resources can be expected to be available, as well as sufficient computational time when planning for the next day. In the worst case of an excessive long computation time, however, a time limit can be used to obtain an approximative solution or a heuristic procedure can be used. In either case, the main conclusion of this paper should be to consider the information available on planned arrivals as the anticipative model does.

For future research, it is interesting to look into further integration of the scheduling process of patients and the room assignment process. Currently, the patient assignment problem takes the scheduled arrival times of elective patients as input, as found by a scheduling process. It is therefore limited in what can be achieved. If the two processes could be integrated, or if the scheduling process could at least take into account some information from the room assignment process, better results could be obtained. This is, however, a difficult research question, as the two processes have conflicting objectives: scheduling processes typically aim for high utilization of resources such as beds and the operating theatre, whereas the difficulties that the room assignment process tries to solve follow from a high occupancy of the wards.

Acknowledgements We would like to thank Sara Ceschia and Andrea Schaerf from the University of Udine for the many fruitful discussions on the specific subject of this study. This research is funded by a Ph.D. grant of the Agency for Innovation by Science and Technology (IWT).

References

Beliën, J., & Demeulemeester, E. (2007). Building cyclic master surgery schedules with leveled resulting bed occupancy. *European Journal of Operational Research*, *176*(2), 1185–1204. doi:10.1016/j.ejor.2005.06.063.

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. doi:10.1007/s10732-011-9192-0.

Burke, E. K., De Causmaecker, P., Vanden Berghe, G., & Van Landeghem, H. (2004). The state of the art of nurse rostering. *Journal of Scheduling*, *7*, 441–499. doi:10.1023/B:JOSH.0000046076.75950.0b.

Cardoen, B., Demeulemeester, E., & Beliën, J. (2010). Operating room planning and scheduling: a literature review. *European Journal of Operational Research*, *201*(3), 921–932. doi:10.1016/j.ejor.2009.04.011.

Ceschia, S., & Schaerf, A. (2011). Local search and lower bounds for the patient admission scheduling problem. *Computers & Operations Research*, *38*(10), 1452–1463. doi:10.1016/j.cor.2011.01.007.

Ceschia, S., & Schaerf, A. (2012). Modeling and solving the dynamic patient admission scheduling problem under uncertainty. *Artificial Intelligence in Medicine*, *56*(3), 199–205. doi:10.1016/j.artmed.2012.09.001.

Demeester, P. (2012). Patient admission scheduling problem website. Online, http://allserv.kahosl.be/~peter/pas/.

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. doi:10.1016/j.artmed.2009.09.001.

Fei, H., Chu, C., & Meskens, N. (2008). Solving a tactical operating room planning problem by a column-generation-based heuristic procedure with four criteria. *Annals of Operations Research*, *166*(1), 91–108. doi:10.1007/s10479-008-0413-3.

Guinet, A., & Chaabane, S. (2003). Operating theatre planning. *International Journal of Production Economics*, *85*(1), 69–81. doi:10.1016/S0925-5273(03)00087-2.

IBM ILOG (2012). Ibm ilog cplex optimization studio information center. Online, http://pic.dhe.ibm.com/infocenter/cosinfoc/v12r4/index.jsp.

Kolen, A. W. J., Lenstra, J. K., Papadimitriou, C. H., & Spieksma, F. C. R. (2007). Interval scheduling: a survey. *Naval Research Logistics*, *54*(5), 530–543. doi:10.1002/nav.20231.

Krishnamoorthy, M., Ernst, A., & Baatar, D. (2012). Algorithms for large scale shift minimisation personnel task scheduling problems. *European Journal of Operational Research*, *219*(1), 34–48. http://www.sciencedirect.com/science/article/pii/S0377221711010435. doi:10.1016/j.ejor.2011.11.034.

R Core Team (2012). *A language and environment for statistical computing*. Vienna: R Foundation for Statistical Computing. http://www.R-project.org/, ISBN 3-900051-07-0.

Rais, A., & Viana, A. (2011). Operations research in healthcare: a survey. *International Transactions in Operational Research*, *18*(1), 1–31. doi:10.1111/j.1475-3995.2010.00767.x.

Riise, A., & Burke, E. K. (2010). Local search for the surgery admission planning problem. *Journal of Heuristics*, *17*(4), 389–414. doi:10.1007/s10732-010-9139-x.

Vancroonenburg, W., Goossens, D., & Spieksma, F. C. R. (2011). *On the complexity of the patient assignment problem* (Tech. rep.). KAHO Sint-Lieven, Gebroeders De Smetstraat 1, Gent, Belgium. URL http://allserv.kahosl.be/~wimvc/pas-complexity-techreport.pdf