Adaptive operating rooms planning and scheduling

Citation for published version (APA):

Kamran, M. A., Karimi, B., Dellaert, N., & Demeulemeester, E. (2019). Adaptive operating rooms planning and scheduling: a rolling horizon approach. *Operations Research for Health Care*, 22, [100200].
<https://doi.org/10.1016/j.orhc.2019.100200>

DOI:

[10.1016/j.orhc.2019.100200](https://doi.org/10.1016/j.orhc.2019.100200)

Document status and date:

Published: 01/09/2019

Document Version:

Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Adaptive operating rooms planning and scheduling: a rolling horizon approach

Mehdi A. Kamran^{1,2}, Behrooz Karimi^{3,*}, Nico Dellaert⁴, Erik Demeulemeester⁵

¹*Faculty of Business and Economics, Department of Logistics, Tourism and Service Management, German University of Technology, Muscat, Oman*

²*Department of Industrial Engineering, Urmia University of Technology, Urmia, Iran*

³*Department of Industrial Engineering and Management Systems, Amirkabir University of Technology, Tehran, Iran*

⁴*Department of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, Eindhoven, Netherlands*

⁵*KU Leuven, Faculty of Economics and Business, Department of Decision Sciences and Information Management, Leuven, Belgium*

Abstract

Accounting for a large portion of the hospital's total revenue and cost, better management of the operating rooms is extremely important in improving healthcare resource utilization. This paper investigates the *Operating Rooms (ORs) Planning and Scheduling Problem* in a hospital with a *modified block scheduling policy*. Thus, the candidate patients have to be assigned a date and an operating room/block as well as being sequenced in the assigned operating rooms/blocks. A *reserved slack policy* is considered to take care of the arrival of emergency patients. Surgery durations are considered to be randomly distributed. In this regard, a stochastic mixed integer linear programming model is proposed that includes different patient, staff and surgeon preferences: minimization of the total patient waiting time, the tardiness, the number of cancellations, the patient surgery start times, the block overtime, the number of surgeon's surgery days within the planning horizon and the sum of the idle times of the surgeons. Two different 2-phase heuristic solution approaches are developed in a rolling horizon framework in order to solve the *Adaptive ORs Planning and Scheduling Problem*. The efficiency of the solution framework is surveyed by applying real data obtained from hospital records through numerical experiments. The results show that the developed solution framework significantly outperforms the commercial solver CPLEX in terms of solution quality and CPU time, in medium- as well as in large-sized problems. Furthermore, the results show that the assumptions and features made to the formulation (i.e. the modified block scheduling policy, the reserved slack policy, and the stochastic surgery durations) will result in more efficient solutions.

Keywords: Adaptive Operating Rooms Planning and Scheduling Problem; Modified Block Scheduling Policy; Reserved Slack Policy; Elective and Emergency patients; 2-phase Heuristic.

1. Introduction

It is well established that the operating theatre is one of the most important elements in the chain of care creation that generates more than 40% of a hospital's total revenues and expenses (Roland et al., 2010; Gordon et al., 1988; Denton et al., 2007; Healthcare Financial Management Association 2003). Furthermore, a continually growing demand for medical care, an aging population and increasing requirements from better-informed patients increase the complexity of the problem of delivering high-quality services under limited available resources. Hence, it is essential to design an efficient operating room management scheme that comes with improvements in the patient flow and in the physical/human resource utilization (Etzioni et al., 2003).

The operating rooms planning and scheduling problem deals with allocating a proper date and OR/block to the on-hand surgeries, called the *ORs Planning Problem* or the *Advance Scheduling Problem*, as well as sequencing the allocated surgeries in ORs/blocks, called the *ORs Scheduling Problem* or the *Allocation Scheduling Problem*. Discussing and considering several antagonistic objectives (e.g., minimizing costs, waiting and idle time, tardiness, overtime, number of cancellations or maximizing resource utilization, patients/surgeons/staff satisfaction, etc.), several constraints (e.g., the human and physical resources availability) should be met in order to ensure a fluent surgery flow in different steps of pre-operative, peri-operative, and post-operative surgical activities (Guerriero and Guido 2011; Batun, et al., 2011). An inefficient plan and/or schedule will keep the decision makers away from the hospital goals, while in the meantime unexpected events or disruptions (i.e., the arrival of an emergency, surgery duration variability, an equipment failure, resource unavailability, etc.) makes planning and scheduling a challenging task.

Hospitals usually have a specific policy in assigning capacity to different specialty groups of surgeons. This could be an *Open Scheduling Policy*, in which there is no specific classification on different specialty groups, and the planning and scheduling of the patients are determined based on their priority, usually based on the first-come-first-serve rule. An alternative is the *Block Scheduling Policy*, in which the available capacity of operating rooms is scheduled based on a prior allocation of the time intervals (blocks) to each surgery group or to particular surgeons. Finally, *Modified Block Scheduling Policy* is an option, which is the combination of the two previous policies: a block scheduling policy could be applied for some blocks, while others remain open or with a reallocation of the unused time to other areas of surgery or different other feasible combinations (Fei et al., 2009; Guerriero & Guido, 2011).

Regarding emergency patients, a *dedicated policy*, an *insertion policy*, and a *reserved slack policy* are the three most common policies that are adopted by hospitals where an insertion policy is classified in the category of post-disruption management policies, and dedicated and reserved slack approaches are classified in the category of predictive disruption management policies. In an insertion policy all capacities of the operating rooms are planned for elective patients in advance, and emergency patients are directly inserted into the schedule; in a dedicated policy, one or more operating rooms are reserved for the emergency patients; and in a reserved slack policy, an amount of time, usually called slack, from

the total capacity of each or some of the operating rooms is reserved for the emergency patients. It is clear that by applying a dedicated policy, although separation of the flow of the different patients helps to increase arrangements, the hospital cannot efficiently utilize its resource capacities. On the other hand, by employing an insertion policy, although the hospital makes more efficient use of its resources, delays, waiting times and disorderliness will cause dissatisfaction of the personnel, the surgeons, and especially the scheduled elective patients. A reserved slack policy could have most of the advantages of both other mentioned policies, with less chaos and conversions (for more details refer to [Ferrand et al., 2014](#); [Van Riet and Demeulemeester, 2015](#)).

In this paper, the arrival of emergency patients is managed in a predictive disruption management approach by the reserved slack policy. The required capacity is predicted beforehand while planning for the operating rooms and embedded in the schedule as slacks. Generally, this paper addresses a framework to solve an ORs planning and scheduling problem with a modified block scheduling strategy, considering uncertain surgery durations and emergency patients arrivals in a rolling horizon approach to capture updates and disruptions.

2. Literature Review

Since the literature on the planning and scheduling of surgeries in ORs is extensive, we focus on the literature that is most relevant to our problem. For a more detailed review and taxonomy, the reader can read relevant review papers such as [Magerlein and Martin \(1978\)](#), [Cardoen et al. \(2010\)](#), [Guerriero and Guido \(2011\)](#), [Van Riet and Demeulemeester \(2015\)](#), [Samudra et al. \(2016\)](#), and [Zhu et al. \(2019\)](#).

[Guinet and Chaabane \(2003\)](#) considered the ORs planning and scheduling problem in a 2-phase approach. In the first planning phase, an extended version of the Hungarian method was developed to assign the patients to operating rooms over the time horizon. In the second phase, they treated the sequencing problem as a two-stage no-wait hybrid flow-shop problem. [Roland et al. \(2010\)](#) investigated the ORs planning and scheduling problem. Due to the computational complexity of the problem, a heuristic solution procedure based on genetic algorithms was suggested to effectively solve the mathematical model. [Fei et al. \(2010\)](#) studied the ORs planning and scheduling problem considering the availability of the human and physical resources of surgeons, ORs and recovery beds with an open scheduling policy. They developed a 2-phase solution approach. In the first phase, the planning problem is solved by using a column-generation-based heuristic procedure, and in the second phase, the daily scheduling problem is described as a two-stage hybrid flow-shop problem and solved by a hybrid genetic algorithm. Numerical experiments showed superior results in terms of the idle time between surgical cases, the utilization of the operating rooms and the overtime in comparison with the actual surgery schedules in a Belgian university hospital. [Marques et al. \(2012\)](#) jointly considered advance and allocation scheduling problems on elective patients with the aim of maximizing the use of the

surgical suite. They developed an integer linear programming model and a simple 2-phase improvement heuristic to solve the problem. The efficiency of the proposed approach is tested on real data obtained in a Lisbon hospital. The results showed, in addition to compliance with the hospital conditions, that their approach improves the objective function of maximizing the use of the surgical suite. They carried on their research in [Marques et al. \(2014\)](#) by means of considering two conflicting optimization criteria, namely of maximizing the use of the surgical suite and the number of scheduled surgeries. They contributed to the problem by developing two versions of a single objective genetic algorithm as a solution approach and examined their approach on the data. The computational results showed improvements in terms of solution quality and CPU time. [Molina-Pariente et al. \(2015\)](#) addressed ORs planning and scheduling problem with surgical teams composed of one or two surgeons where surgery durations depend on their experience and skills. In doing so, a mixed integer linear programming model and an iterative constructive solution method are proposed. The conducted computational experiments showed that the proposed algorithm finds feasible solutions in shorter CPU times and provides better average relative percentage deviations than the MILP model. [Noorizadegan and Seifi \(2018\)](#) proposed a surgery planning and scheduling problem with chance constraints. A solution method based on a decomposition of the set-partitioning formulation of the problem is developed in such a way that the column generation subproblem is decomposed over ORs and time periods. They benefited from the shortest path problem structure in their subproblem and designed a search algorithm to effectively solve the resulting subproblem. Their proposed approach outperformed the standard chance-constrained model in terms of CPU time.

There are several other papers that investigate the ORs planning and scheduling problem in a daily planning horizon, rather than in a weekly planning horizon, or at least consider a small time horizon. In those papers, assignment decisions are being made along with sequencing ones. Some of the examples of such research are: [Jebali et al. \(2006\)](#), [Denton et al. \(2007\)](#), [Pham and Klinkert \(2008\)](#), [Batun et al. \(2011\)](#), [Meskens et al. \(2013\)](#), [Xiang et al. \(2015\)](#), [Latorre-Núñez et al. \(2016\)](#), and [Bam et al. \(2017\)](#).

From the literature, it can be inferred that due to the complexity of the ORs planning and scheduling problem, the main focus of the literature has been on deterministic approaches rather than on uncertain/stochastic approaches ([Pulido et al., 2014](#)). Moreover, less attention has been paid to the arrival of emergency patients and other possible disruptions (e.g., surgery cancellations or rejections, etc.). On the other hand, to the best of the authors' knowledge, this is the first time that the adaptive ORs planning and scheduling problem is considered in a modified block scheduling policy with consideration of stochastic/uncertain surgery durations and emergency patients' arrivals. Also, from reviewing the literature, one can find out that there are scarce studies like [Rachuba and Werners \(2017\)](#), which considered the preferences of different stakeholders in the objective function (e.g., patient, staff, management preferences). The reason of such an approach is because of the fact that most solution methods that are developed for the scheduling problems, generally speaking for *Job Shop* or *Flow Shop*

Scheduling Problems, are specific for certain types of objective functions. Thus, most researchers just focus on those specific objective functions (e.g., minimizing the cycle time or minimizing the overtime or the tardiness, etc.) without considering the multiple interests of different groups affected by the schedule. Overall, this paper tries to cover the above-mentioned gaps. In doing so, assigning a proper date and operating room block to the patients that are waiting in the list and daily sequencing them regarding different perspectives of the hospital, surgeon, patient and staff preferences in a modified block booking policy is considered. Uncertainty in surgery durations are tackled with stochastic variables and the arrival of emergency patients is considered in the problem. A framework is developed for constructing a mid-term solution in a day-by-day rolling horizon approach to be able to adapt the proposed schedule to daily disruptions. More details about the problem statement, the assumptions, and the solution approach are discussed in Sections 3 and 4, respectively. Section 5 verifies the effectiveness of the solution methodology by means of solving some numerical examples that are based on real data obtained from hospital records. Finally, the conclusions are discussed in Section 6.

3. Problem Statement

Inspired by the rolling horizon approach for solving the advance scheduling problem proposed by [Addis et al. \(2016\)](#), and extending [Kamran et al. \(2018a,b\)](#), wherein they discussed the advance scheduling problem and the allocation scheduling problem separately, in this study, adaptive ORs planning and scheduling problems (*AORPS*) are considered in a modified block scheduling policy with consideration of stochastic/uncertain surgery durations and emergency patients' arrivals. The initial patient waiting list is updated in a day-by-day rolling horizon approach, regarding new patient arrivals and/or surgery cancellations/rejections. Strictly speaking, by running the solution framework, the planning and scheduling of patients to the ORs (*ORPS*) is solved for a certain planning horizon, say a week. According to an adapting and a look-ahead strategy, at the end of the first day of the schedule, taking into account the realization of the real-life situations and possible disruptions (e.g., new patient arrivals and/or surgery cancellations/rejections, etc.), the solution framework is run again to update and modify the previous schedule. Therefore, the proposed framework provides both a schedule and a reschedule to the *ORPS* in a mid-term planning horizon. In order to have minimum conversions in the daily scheduling, the possible arrival of emergency patients is forecasted beforehand and added to the waiting lists, i.e. reserved slack policy. It is worth noting that in our models the arrival of emergency patients is predicted formerly and embedded in the patients' waiting list using a predictive disruption management approach. However, during the implementation of the scheduled plan within a day, at the time when the unexpected events occur, post-disruption management strategies (e.g., insertion policies) are more likely to be used (for more details about how to deal with unexpected disruptions, such as arrivals of emergency patients, we refer to [Kamran et al., 2018a](#)).

Figures 1 and 2 depict the architecture of the proposed framework.

Figure 1. The Adaptive Operating Rooms Planning and Scheduling Framework (inspired by Addis et al. (2016))

Figure 2. Rolling horizon approach in the AORPS (inspired by Addis et al. (2016))

F1: In this component, a one-week assigning and sequencing arrangement of patients is established regarding the *ORPS* model. A detailed model description is deferred to the subsequent subsection.

F2: This module generates the realization of the random surgery duration of each patient on the first day of the schedule.

F3: The arrival and surgery cancellation/rejection of new patients (elective and emergency) lead to an update of the patient list. The updated list is fed to the *ORPS* module (F1) together with the current schedule for the remaining weeks/days.

Accordingly, a set of patients, consisting of elective and emergency patients, $Q = P \cup I$ should be planned and sequenced in a set of blocks B of $|R|$ operating rooms in a set of days D . All patient are

assigned a priority coefficient U_q ($q = 1, \dots, |Q|$) to be able to weigh them based on the clinical preferences. A release and due date/time, A_q and D_q , is assigned to each patient. Surgery (elective) cancellation is allowed and a patient is considered tardy if he/she does not receive the required surgical services before his/her due time. The *Master Surgical Schedule* is assumed to be arranged beforehand. Thus, the distribution of specialty groups to blocks of each OR is assumed to be known. Consequently, (elective) patients should be operated in their related blocks, but according to a modified block scheduling policy, considering medical directions and discretions, some elective patients could be operated in some blocks other than the related one. It is worth noting that such a policy could also cover *open or block scheduling policies*, either by authorizing the patients to be operated in all rooms/blocks or by preventing them from being operated in rooms/blocks other than their related ones. Emergency patients are entitled to be operated in all blocks. The allotment of surgeons to the patients is predetermined. Each surgery is carried out through three stage processes of pre-operative, peri-operative, and post-operative surgical activities. It is mandatory that the peri-operative stage is accomplished by the surgeon, but the pre- and post-operative stages are allowed to be left to nurses or to co-surgeons, who are not supposed to be bottleneck resources in this study. In contrast to some research, e.g., [Wang et al. \(2016\)](#) and [Jebali and Diabat \(2017\)](#), the downstream resources such as the post-anesthetic care unit or the intensive care unit resources are not considered as bottleneck resources either. Surgery duration uncertainty is handled by a finite set of (discrete) random durations (i.e., scenarios, $\omega \in \Omega$). In order to schematically illustrate the proposed *ORPS* problem in Figure 3, we figured a hospital consisting of three ORs, which books the patients based on a modified block scheduling policy. Each OR is assumed to be assigned to one surgery group on each day. Different sets of elective patients (with four surgery groups of Neurology, Orthopedic, Female, and Cardiology), emergency patients, and surgeons' of each group are defined. Emergency patients are managed based on the reserved slack policy. Patients are planned and scheduled in a one-week time horizon.

Figure 3. An illustration of the proposed *ORPS* problem at each iteration of the solution framework.

In the following subsection, the notation applied to represent the mathematical model of the *ORPS* problem is addressed.

3.1. Notations

The notations used in the mathematical model are summarized in Table 1.

Table 1. Notations used in the paper

Indices, sets, and parameters:

P	set of elective patients ($p, \hat{p} \in P$, where $ P $ is the number of elective patients)
I	set of non-elective patients ($i \in I$, where $ I $ is the number of non-elective patients)
Q	set of elective and non-elective patients ($q, \hat{q} \in Q = P \cup I$, where $ Q = P + I $ is the number of all patients)
R	set of operating rooms ($r \in R$, where $ R $ is the number of operating rooms)
B	set of blocks ($b \in B$, where $ B $ is the total number of blocks in all ORs)
S	set of surgeons ($s \in S$, where $ S $ is the number of surgeons)

E	set of expertise (specialty groups) ($e \in E$, where $ E $ is the number of specialty groups)
D	set of days, the time horizon ($d \in D$, where $ D $ is the total number of days in the planning horizon)
Ω	set of scenarios ($\omega \in \Omega$, where $ \Omega $ is the total number of scenarios)
Q_b	set of potential patients (elective and non-elective) who are permitted to be assigned to block b
S_q^P	surgeon that is assigned to elective patient q , $q \in P$, $S_q^P \in S$
U_q	priority of patient $q \in P \cup I$
A_q	release time/date for the surgery of patient q
D_q	due time/date for the surgery of patient q
$N_s^{\max}$	maximum number of surgeries allowed to be accomplished on a day by a surgeon
D_d^D	$= d$, total number of waiting days by a patient during the planning horizon
$T_{q[\omega]}^{Pre}$	regular pre-operation duration of patient q , $q \in P \cup I$ in scenario ω
$T_{q[\omega]}^O$	regular surgical duration of patient q , $q \in P \cup I$ in scenario ω
$T_{q[\omega]}^{Post}$	regular post-operation duration of patient $q \in P \cup I$ in scenario ω
$T_{q[\omega]}^{Cl}$	time needed for cleaning the ORs after the surgery of patient q and preparing it for the next surgery in
$T_{q[\omega]}^S$	time needed for the surgeon of patient $q \in P$ to rest after his surgery in scenario ω
S_b^B	regular start time of block $b \in B$
E_b^B	regular capacity (finish time) of block $b \in B$
$O_b^{\max}$	upper bound on the overtime in each block $b \in B$
m_j	weight of term j ($j = 1, \dots, 9$) in the objective function
$P_{[\omega]}$	probability of scenario ω
M	big positive number

Decision variables:

x_{qbd}	=1 if patient q is assigned to block b on day d ; 0 otherwise
n_{sd}	=1 if surgeon s is scheduled for a surgery on day d ; 0 otherwise
$y_{q\dot{q}bd}$	=1 if patient q is operated before patient $\dot{q}$, on the same block on the same day; 0 otherwise
$z_{q\dot{q}d}$	=1 if patient q is operated before patient $\dot{q}$, on the same day, with the same surgeon; 0 otherwise
$s_{qd[\omega]}$	start of surgical care of patient q on day d in scenario ω
$s_{sd[\omega]}^S$	start time of the surgeries of surgeon s on day d in scenario ω
$e_{sd[\omega]}^S$	end time of the surgeries of surgeon s on day d in scenario ω
$i_{sd[\omega]}$	idle time of surgeon s between his/her surgeries on day d in scenario ω
$c_{bd[\omega]}^B$	completion time of the surgeries in block b on day d in scenario ω
$o_{bd[\omega]}^B$	overtime in block b on day d in scenario ω

3.2. Mathematical Model

The stochastic mixed integer programming model of the proposed *ORPS* problem is given below:

$$Obj_{ORPS} = \min \left\{ \begin{aligned} & \left(\sum_q \sum_b \sum_d [m_1(D_d^D - A_q) + m_2(D_d^D - D_q)^+] x_{qbd} U_q \right) + \\ & \left(\sum_q [m_3(D_q - A_q) + m_4((D+1) - D_q)] \left(1 - \sum_b \sum_d x_{qbd} \right) U_q \right) + \\ & m_5 \left(\sum_q \left(U_q - \sum_b \sum_d U_q x_{qbd} \right) \right) + m_6 \left(\sum_s \sum_d n_{sd} \right) + \\ & \sum_{\omega} P_{[\omega]} \left(m_7 \left(\sum_b \sum_d o_{bd[\omega]} \right) + m_8 \sum_s \sum_d i_{sd[\omega]} + m_9 \sum_q \sum_d U_q s_{qd[\omega]} \right) \end{aligned} \right\} \quad (1)$$

Subject to:

$$\sum_d x_{qbd} \leq 1, \quad \forall q \in Q_b, b \in B, \quad (2)$$

$$\sum_d \sum_b x_{qbd} = 1, \quad \forall q \in I, \quad (3)$$

$$A_q \sum_b x_{qbd} \leq D_d^D, \quad \forall q \in Q, d \in D, \quad (4)$$

$$\sum_b x_{qbd} \leq n_{sd}, \quad \forall q \in P, d \in D, s \in S, \quad (5)$$

$$\sum_{q \in \{Q_b | S_q^P = s\}} \sum_b x_{qbd} \leq N_s^{\max}, \quad \forall d \in D, s \in S, \quad (6)$$

$$y_{q\dot{q}bd} + y_{\dot{q}qbd} \leq \frac{x_{qbd} + x_{\dot{q}bd}}{2}, \quad \forall q, \dot{q} \in Q_b, b \in B, d \in D, \quad (7)$$

$$y_{q\dot{q}bd} + y_{\dot{q}qbd} \geq x_{qbd} + x_{\dot{q}bd} - 1, \quad \forall q, \dot{q} \in Q_b, b \in B, d \in D, \quad (8)$$

$$y_{q\dot{q}bd} - y_{\dot{q}qbd} \geq 0, \quad \forall q \in P, \dot{q} \in I, b \in B, \quad (9)$$

$$s_{qd[\omega]} \geq S_b^B x_{qbd}, \quad \forall q \in Q_b, b \in B, d \in D, \omega \in \Omega, \quad (10)$$

$$s_{qd[\omega]} \leq \sum_b (E_b^B + O_b^{\max}) x_{qbd} - (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl}) \left(\sum_b x_{qbd} \right), \quad \forall q \in Q_b, d \in D, \forall \omega \in \Omega, \quad (11)$$

$$s_{\dot{q}d[\omega]} \geq s_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl} - M(1 - y_{q\dot{q}bd}), \quad \forall q, \dot{q} \in Q_b, \dot{q} \neq q, b \in B, d \in D, \omega \in \Omega, \quad (12)$$

$$s_{\dot{q}d[\omega]} + T_{q[\omega]}^{Pre} + M(2 - x_{qbd} - x_{\dot{q}bd}) \geq s_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S - M(1 - z_{q\dot{q}d}), \quad \forall q, \dot{q} \in P, \dot{q} \neq q, S_q^P = S_{\dot{q}}^P, \forall b, \dot{b} \in B, \dot{b} \neq b, \forall b \in B_r^R, \forall \dot{b} \in B_{\dot{r}}^R, \forall \omega \in \Omega, \quad (13)$$

$$s_{qd[\omega]} + T_{q[\omega]}^{Pre} + M(2 - x_{qbd} - x_{\dot{q}bd}) \geq s_{\dot{q}d[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S - M(z_{q\dot{q}d}), \quad \forall q, \dot{q} \in P, \dot{q} \neq q, S_q^P = S_{\dot{q}}^P, \forall b, \dot{b} \in B, \dot{b} \neq b, \forall b \in B_r^R, \forall \dot{b} \in B_{\dot{r}}^R, \forall \omega \in \Omega, \quad (14)$$

$$s_{sd[\omega]}^S \leq s_{qd[\omega]} + T_{q[\omega]}^{Pre} + M \left(1 - \sum_b x_{qbd} \right), \quad \forall s \in S, q \in \{P | S_q^P = s\}, d \in D, \omega \in \Omega, \quad (15)$$

$$e_{sd[\omega]}^S \geq s_{qd[\omega]} + (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S) \left(\sum_b x_{qbd} \right), \quad \forall s \in S, q \in \{P | S_q^P = s\}, d \in D, \omega \in \Omega, \quad (16)$$

$$i_{sd[\omega]} \geq (e_{sd[\omega]}^S - s_{sd[\omega]}^S) - \sum_b \sum_{q \in \{Q_b | S_q^P = s\}} (T_{q[\omega]}^S + T_{q[\omega]}^O) x_{qbd}, \quad \forall s \in S, d \in D, \omega \in \Omega, \quad (17)$$

$$c_{bd[\omega]}^B \geq s_{qd[\omega]} + (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_q^{Cl}) x_{qbd}, \quad \forall q \in Q, b \in B, d \in D, \omega \in \Omega, \quad (18)$$

$$o_{bd[\omega]}^B \geq c_{bd[\omega]}^B - E_b^B, \quad \forall b \in B, d \in D, \omega \in \Omega, \quad (19)$$

$$o_{bd[\omega]}^B \leq O_b^{\max}, \quad \forall b \in B, d \in D, \omega \in \Omega, \quad (20)$$

$$x_{qbd}, y_{q\dot{q}b}, z_{q\dot{q}}, n_{sd} \in \{0,1\}, \quad \forall q, \dot{q} \in Q, b \in B, d \in D, s \in S, \quad (21)$$

$$s_{qd[\omega]}, s_{sd[\omega]}^S, e_{sd[\omega]}^S, i_{sd[\omega]}, c_{bd[\omega]}^B, o_{bd[\omega]}^B \geq 0, \quad \forall q \in Q, b \in B, d \in D, s \in S, \omega \in \Omega. \quad (22)$$

In the proposed model, objective function (1) is made up of nine terms which are normalized by penalty coefficients $m_j, j = 1, \dots, 9$. The first and second term respectively minimize the weighted sum of the waiting time and the tardiness for the patients who are planned to receive clinical services in the planning horizon. The third and fourth term respectively penalize the weighted sum of the waiting times and the tardiness for the unscheduled patients whose due times are within the planning horizon. Due to the fact that the admission time of unscheduled patients are not known, and thus, the real tardiness of such patients cannot be figured out, the day after the end of the planning horizon ($D + 1$) is supposed to be the admission date. The fifth term associates the number of unscheduled patients regarding preferences on patient's urgency coefficient U_q . Thus, the waiting time, the tardiness, and the number of rejected/unscheduled patients are penalized through the third to the fifth terms in the objective function. The sixth term minimizes the number of surgeon's surgery days within the planning horizon. The seventh term represents the total blocks overtime throughout the planning horizon. The eighth term calculates the idleness of the surgeons during the planning horizon, and finally, the ninth term concerns the start time of the surgeries during the planning horizon. Constraints (2) guarantee that each patient receives the surgical services at most once. Constraints (3) enforce inevitably to embed emergency patients to the schedule. Constraints (4) impose the surgery date to be scheduled after the release time of the patient surgery (A_q). Constraints (5) force n_{sd} to become 1 when a surgeon has at least one surgery in the blocks of a day. Constraints (6) guarantee that each surgeon obeys the maximum permitted number of surgeries on each day. Constraints (7) and (8) jointly determine the sequence of the surgeries in a block. According to the reserved slack policy, constraints (9) set the emergency surgeries at the end of each block time. Constraints (10) and (11), respectively, restrict the start times of surgeries to be greater and smaller than respectively the regular start and finish times of the related

block. Constraints (12) establish the start time of the surgeries at each block. Besides, it provides the precedence relation between the surgeries and prevents cyclic surgery sequences. Without this constraint, and just considering constraints (7) and (8), we may have cyclic surgery sequences. Constraints (13) and (14) jointly prevent overlaps between surgeries of a surgeon in different OR blocks. Constraints (15), (16) and (17) represent the start time, end time and idle time of each surgeon at each day, respectively. Constraints (18), (19) and (20) calculate the completion time and the overtime of surgeries at each block, and bound the overtime from exceeding an upper bound. Finally, constraints (21) and (22) define binary and nonnegativity domain restrictions of the different variables.

4. Solution Methodology

Along with strategic and tactical level decisions, operational level ORs scheduling decisions, consisting of weekly planning and daily scheduling decisions, separately or integrated, have received growing attention in the last two decades (Cardoen et al., 2010) and various solution approaches have been applied to solve these operational level problems. From Cardoen et al. (2010), we know that the mathematical programming approaches (including integer programming, mixed integer programming, Lagrangian relaxation, column generation, branch and price, L-shaped decomposition methods, etc.) and heuristic and meta-heuristic approaches (SA, TS, GA, etc.) are the most commonly used ones. On the other hand, considering multiple operating rooms/blocks in our problem, the block-based structure of the problems motivates the application of decomposition methods: from column generation (Lamiri et al., 2008; Fei et al., 2009, 2010; Hashemi Doulabi et al., 2014; Wang et al., 2014; Kamran et al., 2018a, etc.) and branch and price methods (Cardoen et al., 2009; Fei et al., 2008; Beliën & Demeulemeester, 2008; Noorizadegan & Seifi, 2018, etc.) to Benders' and L-shaped decomposition methods (Denton and Gupta, 2003; Denton et al., 2007, 2010; Batun et al., 2011; Xiao et al., 2016; Roshanaei et al., 2017a,b; Kamran et al., 2018a,b, etc.). But in this paper, none of the above-mentioned decomposition methods is applied, and instead, a simple but fast heuristic is developed as a solution method. The reason for such a decision is described in the following:

- In the relevant literature of *ORPS* problems, some valid inequalities have been introduced to eliminate the symmetry in the solutions. Those so-called symmetry breaking constraints make the search in the solution space more efficient by eliminating the symmetry of the solutions (Denton et al., 2010; Batun et al., 2011). As an example, Batun et al. (2011) proposed such symmetry breaking constraints in their mathematical model and applied them in a Benders' decomposition algorithm. Practically, those symmetry breaking constraints improve the relation and knowledge between the Benders' master and subproblem, and consequently improve the performance of the method in searching for optimal solutions (Batun et al., 2011). In our problem, adhering to the realistic assumption that the patients are not identical, with respect to their clinical situations and release and due dates, prevents us from

employing patients' symmetry breaking constraints. Thus, we cannot accelerate the performance of a classic Benders decomposition approach by applying such inequalities. Although [Kamran et al. \(2018b\)](#) shows that it could have slight advantages, a classic Benders decomposition approach cannot provide beneficial results in such problems with real-sized scales.

- According to [Kamran et al. \(2018a\)](#), the concept of the modified block scheduling strategy will make it complicated to take advantage of the column generation-based decomposition methods. Having the possibility to be operated in some other blocks rather than just the related one, correlates the blocks, i.e., columns, to each other, therefore, one cannot benefit from solving smaller block-based subproblems. Besides, considering surgeons as resources in the problem, along with taking into account the surgeons' idle time in the objective function exacerbates the concern, as they could be planned in rooms/blocks other than their related ones. These features complicate the use of combinatorial algorithms/problems such as the shortest path problem, the labeling methods, etc. which were applied in the literature for solving the subproblems in a more convenient way ([Dellaert et al., 2018](#); [Noorizadegan and Seifi, 2018](#)).

Considering the above-mentioned dilemmas and taking into account that the proposed *AORPS* needs to be run every day, it is necessary to be able to run the framework in a relatively small time. Especially considering different alternatives and when getting the patients' and the surgeons' approval is needed, it is necessary to do the calculations in a short time. Therefore, a fast 2-phase heuristic is suggested to solve the ORs planning and scheduling problem (F1) which will be described in the following.

Knowing that the ORs planning and scheduling problem is strictly NP-hard, and based on the nature and structure of the problem, it is common to break down the solution approach of the problem into a 2-phase one. Commonly, the first phase consists of assignment decisions (the planning phase decisions) and the second phase relates to the sequencing decisions (the scheduling phase decision) ([Guinet and Chaabane, 2003](#); [Jebali et al., 2006](#); [Fei et al., 2006, 2010](#); [Marques et al., 2012](#); [Bam et al., 2017](#), etc.). The difference between the approach developed in this paper with previous research in the literature is two-fold:

(I) We solve and determine the assignments in an iterative multi-stage manner. Plainly, those patients who have later release dates could not be operated earlier, but taking into account the patient's clinical situation and also, considering the associated penalty in the objective function, patients could be planned to be operated even after their due date D_q . Let us make this clear by giving an example. Suppose that we are planning the patients on-hand for the week ahead (Day1–Day7). Day3 and Day5 are considered as the release and due dates, A_q and D_q , of patient i , respectively. It is clear that it is desired to plan this operation within its release and due dates. We cannot plan his/her surgery before the release date, but in conditions of compulsion and taking into consideration the patient's clinical situation, maybe we could plan his/her surgery after his/her due date. Accordingly, the patient is considered tardy, and of

course, this postponement will lead to some discontentment which should penalize the objective function. Therefore, with regard to this concept, in our heuristics, the patients are sorted according to their release date A_q , and in a day-by-day approach, the assignment problem (*ORP*) is solved only for the corresponding day. The patients who are kept outside of the plan of that specific day could be planned in the coming days, taking into account its related penalties.

(II) As the major difficulty of the problem lies in the sequencing phase, solving the sequencing phase problem for the one-week planning horizon, even for given assignment decisions, is quite time-consuming. Therefore, two approaches have been tried as the second phase in this paper.

(II-A) The same iterative day-by-day concept is applied in solving the sequencing phase problem (*ORS*). If the solution of the *ORP* is infeasible through *ORS* in terms of the overlaps between surgeries of a surgeon in different OR blocks on a day, a *feasibility combinatorial cut* in the form of (FC1) is incorporated into the *ORP*. These *feasibility combinatorial cuts* guide the model search toward feasibility by providing feedback over the previous *ORP* solution.

$$\sum_{b \in \{B | \bar{x}_{qbd}^l = 1\}} x_{qbd} + \sum_{\bar{b} \in \{B | \bar{x}_{\bar{q}\bar{b}d}^l = 1\}} x_{\bar{q}\bar{b}d} \leq 1, \quad \forall q, \bar{q} \in Q^{list}, l \in L, \quad (FC1)$$

where Q^{list} is the set of patients, updated per each day, and $\bar{x}_{qbd}^l$ is the given solution which is infeasible through *ORS* in the previous iteration ($l \in L$).

(II-B) Rather than solving the *ORS*, an approximation of the *ORS* is solved based on just determining the position of surgeries in each OR/block hereafter called *ORS-A*. In this regard and according to the context of the mathematical modeling of scheduling and sequencing problems, a *position-based* formulation of the sequencing phase problem is developed in contrast to the main formulation which is a *relation-based* formulation. The objective function is approximated as well, regarding the associated terms in the *ORS*. This *ORS-A* significantly reduces the number of variables, and thus the resulting CPU time. Then, other decision variables (i.e., the start time of each surgery and the exact value of the accompanying terms in the objective functions) are calculated based on a calculation model (*ORS-C*). The feasibility of the obtained solution in terms of the overlaps between surgeries of a surgeon in different OR blocks on a day is checked through two checking models (*CHECK1* and *CHECK2*) and resolved by means of applying *feasibility combinatorial cuts* in the form of constraints (FC1) and (FC2). If the solution obtained by the *ORP* is the source of the infeasibility, constraint (FC1) is incorporated into the *ORP*, and in the case the solution obtained by *ORS-A* is the source of the infeasibility, constraint (FC2) is added to *ORS-A* in order to exclude the infeasible solution.

$$\sum_b \sum_{k \in \{K_b | \bar{x}_{qbdk}^l = 1\}} x_{qbdk} + \sum_{\bar{b}} \sum_{\bar{k} \in \{K_{\bar{b}} | \bar{x}_{\bar{q}\bar{b}\bar{d}\bar{k}}^l = 1\}} x_{\bar{q}\bar{b}\bar{d}\bar{k}} \leq 1, \quad \forall q, \bar{q} \in Q^{list}, l \in L: \bar{z}_{q\bar{q}d}^l = 1, \quad (FC2)$$

where $\bar{z}_{q\bar{q}d}^l = 1$ indicates the patients who caused the infeasibility, and $\bar{x}_{qbdk}^l$ is the given solution which is infeasible as detected by the *CHECK1* model.

The details of the proposed assigning phase proceeded by two different sequencing approaches are described in Algorithm 1 (*H1*) and Algorithm 2 (*H2*), respectively. In fact, this iterative recourse approach applied in *H1* and *H2* by incorporating *feasibility combinatorial cuts* to the corresponding models is inspired from a Benders' decomposition method, more specifically from Logic-Based Benders' Decomposition method, as they do not impose structural restrictions (e.g. linearity) on different decomposition components (Hooker, 2000; Hooker and Ottosson, 2003). The corresponding mathematical models, along with the validity proofs of *feasibility combinatorial cuts* (FC1) and (FC2) are deferred to the Appendix.

Algorithm 1 Pseudo-code of the 2-phase heuristic (*H1*): Multi-stage assigning and sequencing phases

```

Initialization
    Generate and sort the patients based on their release date in the patients' waiting list
    Include the patients who have the possibility to be operated on day  $d = 1$ 
For  $d := 1$  to  $D$  do
    Set the feasibility criteria as false: these feasibility criteria consider the overlaps between surgeries of a surgeon in different OR blocks on a day
    While the feasibility criterion is false do
        Solve ORP for day  $d$ 
        Solve ORS for day  $d$ : according to the results of ORP in hand
        If ORS is infeasible
            Insert feasibility combinatorial cuts (FC1) to ORP
        Else
            Set the feasibility criteria as true
        End If
    End While
    Update the patients' waiting list: remove the operated patients and include the patients who have the possibility of being operated on day  $d + 1$ 
End For

```

Algorithm 2 Pseudo-code of the 2-phase heuristic (*H2*): Multi-stage assigning phase and approximate sequencing phase

```

Initialization
    Generate and sort the patients based on their release date in the patients' waiting list
    Include the patients who have the possibility to be operated on day  $d = 1$ 
For  $d := 1$  to  $D$  do
    Set the feasibility criteria 1 & 2 & 3 as false: these feasibility criteria consider the overlaps between surgeries of a surgeon in different OR blocks on a day
    While feasibility criterion 1 is false do
        Solve ORP for day  $d$ 
    While feasibility criterion 2 is false do
        Solve ORS-A for day  $d$ : according to the results of ORP in hand
        Solve ORS-C
        Solve CHECK1 model
        If the objective function of the CHECK1 model is  $> 0$ 
            If feasibility criterion 3 is false

```



```

        Solve CHECK2 model
        If CHECK2 is infeasible
            Insert feasibility combinatorial cuts (FC1) to ORP
            Break
        End If
        Set the feasibility criterion 3 as true
        End If
        Insert feasibility combinatorial cuts (FC2) to ORS-A
    Else
        Set the feasibility criteria 1 & 2 as true
        End If
    End While
End While
Update the patients' waiting list: remove the operated patients and introduce the patients who have the
possibility of being operated on day  $d + 1$ 
End For

```

5. Computational Study

In this section, several illustrative numerical examples are solved through the proposed two 2-phase heuristics and the results are compared with the commercial solver. All the models and solution approaches are programmed by GAMS 24.8.5. and solved with CPLEX 12.6.3 on a computer with Core (TM) 2 Duo CPU with 2.50 GHz speed and 4 GB of RAM.

5.1. Illustrative Test Problems

In order to design different test problems, we considered different hospitals from 1 to 5 ORs, each room consisting of 2 blocks of specialty groups, for 1 to 7 days from 16 to 371 patients. Regular and overtime capacity of the blocks are set to 5 and 1hrs, respectively. 10% of the patients of different specialty groups are authorized to be operated in some other randomly selected blocks. It is acknowledged that the surgery duration fits best to a Log-Normal distribution (May et al., 2000; Zhou and Dexter, 1998; Min and Yih, 2010). Thus, estimating the surgery durations of different surgery/specialty groups, 20 surgery duration scenarios for each surgery group are simulated from its associated Log-Normal distribution, $\mu = 25, 30, 35, 40, 45, 70$ and $\sigma = 0.5$, by a Monte-Carlo simulation. We determined the weights of the terms in the objective function based on various sets of parameter values, partly based on a priori preferences. We compared the performance outcomes depending on these sets and defined the best set which in our case is 1, 3, 1, 3, 30, 2, 0.5 and 1, respectively. In real cases, these weights should be defined in cooperation with hospital management and based on the hospital's specific preferences. Other parameter setting details are: $U_{q\epsilon P} = (U[2,4] + (D_q - A_q))^{-1}$; $U_{q\epsilon I} = (1 + (D_q - A_q))^{-1}$; $A_q = U[1, |D|]$; $D_{q\epsilon P} = \min(A_q + U[|D|, 2|D|], |D|)$; $D_{q\epsilon I} = A_q$; $N_s^{\max} = 4, 3, 2$; $T_{q[\omega]}^{Pre} = U[8, 10]$ min; $T_{q[\omega]}^{Cl} = 10$

$\min; T_{q[\omega]}^{Post} = 0.1 T_{q[\omega]}^O; T_{q[\omega]}^S = \max(T_{q[\omega]}^{Cl}, 0.05 T_{q[\omega]}^O)$, where $U[\tilde{u}, \hat{u}]$ represents a random uniform number between $\tilde{u}$ and $\hat{u}$.

Table 2 summarizes the results and provides a comparison between the performance of the two proposed heuristics and a commercial solver in solving such test problems.

Table 2. Summary of the results of the proposed 2-phase heuristics vs CPLEX through different small-to large-sized test problems

Test Problems		Obj. Func.			Relative Gap* (%)			CPU time** (s)		
No.	(R,D,I, P,Q)	H1	H2	ORPS	H1	H2	ORPS	H1	H2	ORPS
1	(1,1,1, 15,16)	63.94	64.61	62.857	1.72	2.79	0	1.7	2.1	1000.8
2	(1,2,2, 30,32)	120.55	121.27	116.80	3.21	3.82	0	1.42	4.29	1000.48
3	(1,3,3, 45,48)	173.83	174.83	173.12	0.41	0.99	0	11.89	9.90	1002.75
4	(1,4,4, 60,64)	228.27	230.14	222.06	2.80	3.64	0	30.40	15.85	1005.54
5	(1,5,5, 75,75)	280.57	282.43	284.11	0	0.66	1.27	19.48	24.95	1012.65
6	(1,6,6, 90,96)	326.37	329.67	327.97	0	1.01	0.49	82.71	40.25	1021.83
7	(1,7,7, 105,112)	356.55	360.30	388.16	0	1.05	8.87	62.61	54.41	1040.66
8	(2,1,1, 21,22)	53.89	55.016	51.037	5.59	7.80	0	10.35	3.88	2001.99
9	(2,2,2, 42,44)	121.64	129.89	115.42	5.38	12.53	0	32.53	17.23	2010.97
10	(2,3,3, 63,66)	149.88	171.81	143.81	4.22	19.47	0	562.14	21.52	2033.94
11	(2,4,4, 84,88)	227.06	251.29	252.40	0	10.67	11.16	530.89	42.44	2088.03
12	(2,5,5, 105,110)	263.17	288.96	295.38	0	9.80	12.24	246.71	78.56	2187.79
13	(2,6,6, 126,132)	318.87	293.17	291.03	9.57	0.74	0	32.04	126.45	2040.07
14	(2,7,7, 147,154)	312.20	272.68	452.28	14.49	0	65.86	666.23	160.77	2477.50
15	(3,1,2, 30,32)	92.645	99.198	77.67	19.26	27.71	0	230.18	8.141	3013.04
16	(3,2,4, 60,64)	182.10	199.44	173.73	4.82	14.80	0	45.35	25.99	3089.35
17	(3,3,6, 90,96)	256.29	204.88	270.81	25.09	0	32.18	567.36	73.75	3404.44
18	(3,4,8, 120,128)	353.39	372.29	438.51	0	5.35	24.09	608.90	136.67	3983.81
19	(3,5,10, 150,160)	381.83	340.59	446.61	12.11	0	31.13	638.59	289.61	6300.4
20	(3,6,12, 180,192)	435.40	345.88	1204.74	25.88	0	248.31	1698.64	271.46	7241.88
21	(3,7,14, 210,224)	485.52	388.35	1255.83	25.02	0	223.38	2580.21	348.36	8704.43
22	(4,1,2, 40,42)	127.94	13.89	100.71	821.12	0	625.03	508.46	34.15	5062.24
23	(4,2,4, 80,84)	233.28	128.19	226.70	81.99	0	76.86	533.62	498.43	5451.69
24	(4,3,6, 120,126)	306.57	348.36	340.29	0	13.63	11.00	859.56	160.24	6465.34
25	(4,4,8, 160,168)	399.17	230.11	556.64	73.47	0	141.90	1269.61	271.83	8551.12
26	(4,5,10, 200,210)	496.46	312.07	1510.75	59.08	0	384.10	867.11	367.22	11876.02
27	(4,6,12, 240,252)	577.45	230.72	1892.14	150.28	0	720.10	1400.01	573.42	19734.79
28	(4,7,14, 280,294)	620.72	216.43	825.52	186.80	0	281.43	2502.26	1253.96	54930.81
29	(5,1,3, 50,53)	166.23	182.71	135.33	22.83	35.00	0	56.89	46.48	5211.76
30	(5,2,6, 100,106)	327.92	217.2	302.55	50.98	0	39.30	1062.67	258.27	6714.85
31	(5,3,9, 150,159)	433.83	220.37	451.05	96.86	0	104.67	879.00	2844.76	10895
32	(5,4,12, 200,212)	486.16	100.60	699.63	383.28	0	595.49	1779.93	851.98	20461.03
33	(5,5,15, 250,265)	608.66	377.00	1832.6	61.45	0	386.10	2269.27	1327.94	31875.82
34	(5,6,18, 300,318)	754.50	402.08	—	87.65	0	—	2700.70	2102.92	70000
35	(5,7,21, 350,371)	829.13	535.45	—	54.85	0	—	2951.00	2090.34	70000

* The relative gap, $((Obj. Func. - \min(H1, H2, ORPS)) / \min(H1, H2, ORPS)) \times 100$

** A time limit of 500 seconds is set for the ORS at each iteration, and different time limits are set for ORPS for different-sized test problems

From Table 2, it can be inferred that in small-sized problems, the CPLEX solver beats the proposed heuristics. It is plain to see that the size of $|D|$ considerably affects the problem size. Meanwhile, the 2-phase heuristic (H1) is capable of generating solutions more efficiently than the 2-phase heuristic approximation (H2), with a relative gap of less than 10% in most cases. For growing sizes of test

problems, the proposed heuristics outperform the CPLEX solver both in terms of solution quality and CPU time. In fact, in medium- and real-sized problems, the *ORPS* almost completely fails in detecting the optimal solution, even in finding a feasible solution (see test problems No. 34 & 35). The 2-phase heuristic (*H1*) relatively preserves its quality in medium-sized problems, but it loses its quality in large-sized ones. The 2-phase heuristic approximation (*H2*) keeps its performance even in large-sized problems and strongly outperforms the solver. It can be argued that the two proposed heuristic algorithms operate on the basis of a daily approach, so why the size of $|D|$ affects the performance of the algorithms and why does this result in such a difference between the performance of the two algorithms? In response, it should be noted that solving a planning phase problem and/or a scheduling phase problem for $d = 1$ and for $d = D$ is not the same in terms of size and complexity. In fact, in the problem of $d = D$ all the patients who are kept outside of the plans/schedules of the previous days of the planning horizon ($d = 1, \dots, D-1$) are considered in the problem of $d = D$. Thus, by proceeding in the proposed algorithms in a day-by-day fashion, the size and the complexity of the problem either in the planning phase problem or in the scheduling phase problem increases. The difference between the *H1* and *H2* lies in the *ORS* and the *ORS-A*, where the original sequencing problem is not capable of handling this size and complexity increment, but the approximation of the sequencing problem is. Figures 4 to 6 provide better insight by depicting the results.

Figure 4: The objective functions obtained through three different approaches of *ORPS*, *H1*, and *H2*

Figure 5: The relative gaps of the three different approaches of *ORPS*, *H1*, and *H2*

Figure 6: The CPU time of solving the test problems through three different approaches of *ORPS*, *H1*, and *H2*

5.2. Real Data Experiment

In order to demonstrate the efficiency of the formulation that is proposed in this paper, real data obtained from the general surgery department of Radboud University Medical Center, Nijmegen, The

Netherlands (Goedhart, 2014) are applied. The Radboud University Medical Center works under the block scheduling policy and plans and schedules the patients based on the expected surgery times. It also arranges the arrival of emergency patients through a dedicated policy. In order to examine the performance of the hospital under the proposed features, a new Master Surgical Schedule (MSS) is designed for the hospital (Figure 7), the uncertainty in surgery duration is considered by a finite set of (discrete) random durations (i.e., scenarios), and the reserved slack policy is applied instead of the dedicated policy to arrange the arrival of emergency patients. In this regard, to compare the performance of the hospital under the current and the new features, a one-week schedule, according to an iteration of the heuristic algorithm, is generated for the hospital based on a real data set of records.

Designing the experiment, a set of patients including 83 elective patients and 13 emergency patients (in total 96 patients) are considered. Elective patients are of four different surgery groups, namely abdominal & oncology surgery (aos), child surgery (chs), trauma surgery (trs) and vascular & transplantation surgery (vts). The patients should be planned and scheduled, as much as possible, through a one-week schedule (five days). Different surgery duration scenarios are generated through Log-Normal distributions where μ equals the expected surgery duration of the different surgeries. The expected surgery durations of different surgeries and the realizations of the surgeries are all based on hospital records. According to our modeling requirements, some necessary adjustments are made to the data set as follows: the patients' release and due dates and priority coefficients are generated randomly. A set of surgeons is defined and randomly assigned to the surgeries. The duration of the pre- and post-operation activities and the time needed for cleaning the ORs as well as the surgeons' turnover time between surgeries are not reported in the data set. Thus, the pre- and post-operation durations are considered part of the surgery durations and the time needed for cleaning the ORs and surgeons' turnover times are assumed to be 0. $N_s^{\max}$ is set to 3 for all surgeons of all specialty groups. It is also assumed that emergency patients and aos surgeries could be operated in all ORs. The results of the obtained schedule under the current and the new features are reported in Table 3.

	Mon	Tue	Wed	Thu	Fri
OR1	trs	aos	aos	chs	trs
OR2	vts	vts	aos	vts	aos
OR3	aos	aos	aos	aos	chs
OR4	emr	emr	emr	emr	emr

(a)

	Mon	Tue	Wed	Thu	Fri
OR1	trs	aos	trs	aos	trs
OR2	vts	vts	aos	vts	vts
OR3	aos	aos	aos	aos	aos
OR4	aos	aos	chs	chs	chs

(b)

Figure 7: The MSS of Radboud University Medical Center: (a) current situation with a slight adjustment, (b) the proposed MSS*

* The way the MSS is obtained is not the concern of this paper, but as a brief explanation, the dedicated OR capacity for emergency surgeries is distributed between aos, chs, trs, and vts; and very small changes were made in the block locations.

Table 3: Comparison of the schedules obtained under the current features (a) and under the new proposed features (b) for Radboud University Medical Center

ORs*	Mon	Tue	Wed	Thu	Fri
P. No.	65 62	28 16 38 26 33 37 12 36 8 15 18	44 53 50 40 46 48 56 42 52 55 41 60 61 67 66 63 58 68		
1: Ex. Sur. Fin.	1 3.5	1 2 3 4 5 6.5 8 2 4 6 8	0.8 1.8 2.8 3.5 4.3 5 5.75 6.5 7.5 8.2 8.95 1.5 2.5 3.5 4.5 5.5 6.5 8.5		
Realization	0.95 2.72	1 1.55 2.77 3.45 4 5.33 6.78 1.78 3.66 5.56 7.36	0.67 1.5 2.27 3.07 3.97 4.7 5.83 6.41 7.31 8.08 8.85 1.98 2.66 3.84 4.86 5.78 6.13 8.73		
P. No.	79 80	83 69 70 82	29 25 9 2 11 77 72 73 75 71		3 34 24 5 14 17
2: Ex. Sur. Fin.	0.5 2	1 2.5 4 7	1 2 4 6 8 0.5 2 4.5 6 8		1 2 3 4.2 6.3 8.8
Realization	0.5 1.8	0.93 4.35 5.78 9.16	1.63 2.6 4.35 5.63 7.91 0.52 1.49 3.04 4.49 7.02		0.28 0.85 1.33 1.68 4.66 5.68
P. No.	4 7 22 23 21 59		32 1 10 6 20 30 13 35 19		49 43 45 47 54 51 57
3: Ex. Sur. Fin.	0.74 1.7 3.7 6.3 8.3 1.5		1.5 3.5 5 7.5 0.5 2.5 4.5 6.5 9		1 1.7 2.5 3.3 4.3 5.8 7.1
Realization	0.23 0.76 1.64 4.01 4.74 1.4		1.42 2.72 4.07 6.87 0.42 3.6 5.72 7.24 9.19		0.68 1.91 2.59 3.47 4.15 5.97 7
P. No.	94 92 91 87 85		95 86 88 89 90		93 84
4: Ex. Sur. Fin.	1 2.5 4 6 2.5		1 3 1.5 3 4.5		1 4
Realization	1.01 2.51 4.02 6.01 2.5		1 2.99 1.49 3 4.52		0.99 3.99

(a)

ORs*	Mon	Tue	Wed	Thu	Fri
P. No.	65 62 94 92	28 26 38 12 25 66 63 1 64 58 95	30 13 35 18	3 60 61 67 39 68	
1: Ex. Sur. Fin.	0.98 3.50 4.50 6.00	1.00 2.00 3.00 4.50 1.00 2.00 3.00 5.00 6.50 7.50 8.50	2.00 4.00 6.00 8.00	1.00 2.50 3.50 4.50 6.50 8.50	
Realization	0.95 2.72 3.73 5.23	1 1.68 2.9 4.35 0.97 1.99 2.91 4.21 5.43 5.78 6.78	3.18 5.3 6.82 8.62	0.28 2.26 2.94 4.12 5.99 8.59	
P. No.	79 4 7 80 21 83 69 70 82 8 6 11 86		77 75 72 73 71	74 5 81 76 78	
2: Ex. Sur. Fin.	0.50 1.25 2.40 3.74 5.75 1.00 2.50 4.00 7.00 2.00 4.50 6.50 8.50		0.50 2.00 3.50 6.00 8.00	1.00 2.30 3.70 6.20 8.20	
Realization	0.5 0.73 1.26 2.56 3.29 0.93 4.35 5.78 9.16 1.88 3.96 6.24 8.23		0.52 1.97 2.94 4.49 7.02	0.95 1.3 2.57 4.69 7.59	
P. No.	22 87	16 33 37 59 29 36 9 32 2	20 19 15 90 89	24 14 17 84	
3: Ex. Sur. Fin.	2.00 4.00	1.00 2.00 4.50 6 1.00 3.00 5.00 6.50 8.50	0.50 3.00 5.00 6.50 8.00	1.00 3.30 5.70 8.70	
Realization	0.88 2.87	0.55 1.1 2.43 3.83 1.63 3.41 5.16 6.58 7.86	0.42 2.37 4.27 5.79 7.3	0.48 3.46 4.48 7.48	
P. No.	23 91	85	53 40 46 48 56 42 54 55 41 10 44 50 57 52 31 88 45 47 49 43 34 51 27 93		
4: Ex. Sur. Fin.	2.50 4.00	2.50	1.00 1.75 2.50 3.24 4.00 4.75 5.72 6.50 7.30 8.80 0.75 1.75 3.00 4.00 6.50 8.00 0.75 1.50 2.50 3.30 4.20 5.75 7.70 8.70		
Realization	2.37 3.88	2.5	0.83 1.63 2.53 3.26 4.39 4.97 5.65 6.42 7.19 8.54 0.67 1.44 2.47 3.37 5.77 7.26 0.68 1.56 2.24 3.47 4.04 5.86 8.33 9.32		

(b)

* P.No. stands for Patient number, Ex. Sur. Fin. Stands for expected finish time of the surgery, and Realization is the realization of the surgery duration and finish time.

The results show that the assumptions and features made to the formulation will result in a much more efficient schedule. In fact, on the one hand, the modified block scheduling policy and the reserved slack policy provide the flexibility and the competency of a more efficient use of the resources and the time capacities; and on the other hand, stochastic surgery durations and the wide variety of situations (scenarios), make the obtained schedule to remain valid and feasible most often at the time of realizations. To be more strict, according to the new proposed features, as can be seen in Table 3(b), all 83 elective patients and 13 emergency patients are planned and scheduled to be operated during the week. This is while according to the Radboud University Medical Center, 8 elective patients (patients 27, 31, 39, 64, 74, 76, 78, and 81) are kept outside from the schedule of the week (Table 3(a)). With respect to both schedules obtained under the current and the new proposed features, only in two cases out of all realizations, the total surgery durations exceeded the allowed working time in the ORs (indicated in boldface text in Tables 3(a) and 3(b)). However, this is while the schedule under the new features is more congested and more patients receive their required surgical services. Taking all these into consideration, it comes to the conclusion that the new proposed features will lead us to more impressive results.

6. Conclusions

This study investigates the operational level decisions in the territory of operating room capacity management problems called the *ORs planning and scheduling problem (ORPS)* which concerns the allocation of a proper date and OR/block to the on-hand surgeries, as well as the sequencing of the allocated surgeries in ORs/blocks. A *modified block scheduling policy* is considered as the patients' booking policy. To be specific, patients regularly are planned to be operated in their related blocks according to the designed *Master surgical schedule*, but in the proposed modified block scheduling policy, taking into consideration medical directions and discretions, some patients are authorized to be operated in some blocks different from the related one. A *reserved slack policy* is considered to take care of emergency patient arrivals. In fact, the capacity required for devising emergency patient arrivals is predicted beforehand while planning for the ORs and embedded into the schedule as slacks. Surgery durations are supposed to be stochastic variables. A stochastic mixed integer linear programming model is proposed to formulate the problem. Diverse preferences of patients, staff, and surgeons are considered in the objective function including the minimization of the total patient waiting time, the tardiness, the number of cancellations, the patient surgery start times, the block overtime, the number of surgeon's surgery days within the planning horizon and the surgeon's idle time. A rolling horizon solution framework is designed to update and adapt the planned schedule over time. In order to tackle such an adaptive *ORPS*, two different 2-phase heuristic solution approaches are proposed. The efficiency of the solution framework is surveyed through numerical experiments by applying real data obtained from

hospital records. The results demonstrate that the developed solution framework immensely outperforms the commercial solver CPLEX in terms of solution quality and CPU time, in medium- and large-sized problems. Furthermore, the results show that the assumptions and features such as the modified block scheduling policy, the reserved slack policy, and the stochastic surgery durations will significantly cooperate in coming up with more efficient solutions.

Appendix A. The mathematical models applied in the proposed solution frameworks

In this section, six different mathematical formulations employed in the developed solution framework are presented. These formulations include (I) the ORs planning model (*ORP*), (II) the ORs scheduling and sequencing model (*ORS*), (III) the ORs scheduling and sequencing approximation model (*ORS-A*), (IV) the calculation model (*OPS-C*), (V) & (VI) and finally, the checking models (*CHECK1* & *CHECK2*).

Further notations applied in the models are presented in Table 4.

Table 4. Further notations used in developing the mathematical models

Indices, sets, and parameters:

K_b	set of positions of surgeries in block b ($k, k_b \in K_b$)
L	set of iterations <i>ORS-A</i> that is needed to be solved to come up with a feasible solution ($l \in L$)
Q^{list}	set of patients updated per each day
T_q^{Pre}	regular pre-operation duration of patient $q, q \in P \cup I$, mean value scenario
T_q^O	regular surgical duration of patient $q, q \in P \cup I$, mean value scenario
T_q^{Post}	regular post-operation duration of patient $q \in P \cup I$, mean value scenario
T_q^{Cl}	time needed for cleaning the ORs after the surgery of patient q and preparing it for the next surgery, mean value scenario
$\bar{x}_{qbd}$	the solution obtained by the <i>ORP</i> which is used as input in the later models; it indicates if patient q is assigned to block b on day d (=1) or not (=0)
$\bar{x}_{qbdk}$	the solution obtained by <i>ORS-A</i> which is used as input in the coming later models; it indicates if patient q is assigned to block b on day d in position k (=1) or not (=0)
$\bar{x}_{qbdk}^l$	the solution obtained by <i>ORS-A</i> in previous iterations which indicate if patient q is assigned to block b on day d in position k in iteration l (=1) or not (=0)
$\bar{z}_{q\dot{q}d}^l$	the solution obtained by the <i>CHECK1</i> which specifies the part of the solution that brings about infeasibility, and should be excluded in <i>ORS-A</i> in the next iterations
$\bar{s}_{qd[\omega]}$	start of the surgical care of patient q on day d in scenario ω obtained by the <i>ORS-C</i> which is used as input in the <i>CHECK1</i> model

Decision variables:

s_{sd}^S	start time of the surgeries of surgeon s on day d in mean value scenario
e_{sd}^S	end time of the surgeries of surgeon s on day d in mean value scenario
i_{sd}	idle time of surgeon s between his surgeries on day d in mean value scenario
x_{qbdk}	=1 if patient q is assigned to block b on day d in position k ; 0 otherwise

I) Mathematical model of the ORs planning problem (*ORP*)

In the first phase of the proposed heuristic solution framework, it is needed to allocate a date and an operating room for each surgery. The mathematical formulation applied for such an ORs planning problem (*ORP*) is as follows:

$$Obj_{ORP} = \min \left\{ \begin{aligned} & \left(\sum_q \sum_b [m_1(D_d^D - A_q) + m_2(D_d^D - D_q)^+] x_{qbd} U_q \right) + \\ & \left(\sum_q [m_3(D_q - A_q) + m_4((D+1) - D_q)] \left(1 - \sum_b x_{qbd} \right) U_q \right) + \\ & m_5 \left(\sum_q \left(U_q - \sum_b U_q x_{qbd} \right) \right) + m_6 \left(\sum_s n_{sd} \right) + m_7 \left(\sum_b \sum_\omega P_{[\omega]} o_{bd[\omega]} \right) \end{aligned} \right\} \quad (P1)$$

Subject to:

$$x_{qbd} \leq 1, \quad \forall q \in Q_b \cap Q^{list}, b \in B, \quad (P2)$$

$$\sum_b x_{qbd} = 1, \quad \forall q \in I \cap Q^{list}, \quad (P3)$$

$$\sum_b x_{qbd} \leq n_{sd}, \quad \forall q \in P \cap Q^{list}, s \in S, \quad (P4)$$

$$\sum_{q \in \{P \cap Q^{list} | S_q^P = s\}} \sum_b x_{qbd} \leq N_s^{\max}, \quad \forall s \in S, \quad (P5)$$

$$\sum_{q \in Q_b \cap Q^{list}} (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_q^{Cl}) x_{qbd} \leq (E_b^B - S_b^B) + o_{bd[\omega]}^B, \quad \forall b \in B, \omega \in \Omega, \quad (P6)$$

$$o_{bd[\omega]}^B \leq o_{bd[\omega]}^{\max}, \quad \forall b \in B, d \in D, \omega \in \Omega, \quad (P7)$$

$$\sum_{b \in \{B | \bar{x}_{qbd}^l = 1\}} x_{qbd} + \sum_{\dot{b} \in \{B | \bar{x}_{\dot{q}\dot{b}d}^l = 1\}} x_{\dot{q}\dot{b}d} \leq 1, \quad \forall q, \dot{q} \in Q^{list}, l \in L, \quad (FC1)$$

$$x_{qbd}, n_{sd} \in \{0, 1\}, \quad \forall q, \dot{q} \in Q, b \in B, d \in D, s \in S, \quad (P8)$$

$$o_{bd[\omega]}^B \geq 0, \quad \forall b \in B, d \in D, \omega \in \Omega. \quad (P9)$$

The objective function and the constraints of the *ORP* are similar to their counterparts in the *ORPS*, except that they are written with consideration of the updated patients' list, Q^{list} , and they are compiled for a specific day. Accordingly, objective function (P1) is similar to objective function (1) in terms of the first to seventh components. Constraints (P2)–(P5) are equal to constraints (2), (3), (5), and (6), and constraints (P6) and (P7) are the counterparts of constraints (18)–(20). Constraint (FC1) is incorporated into the *ORP* in order to exclude infeasible solutions in terms of the overlaps between surgeries of a surgeon in different OR blocks on a day.

II) Mathematical model of the ORs Scheduling problem (*ORS*)

In the second phase of the proposed heuristic solution framework, in the first approach (H1), it is needed to schedule and sequence the allocated surgeries of each day. The ORs scheduling and sequencing model (ORS) employs the ORP solutions as inputs.

$$Obj_{ORS} = \min \sum_{\omega} P_{[\omega]} \left\{ m_8 \sum_s i_{sd[\omega]} + m_9 \sum_{q \in Q^{list}} U_q s_{qd[\omega]} \right\} \quad (S1)$$

Subject to:

$$y_{q\dot{q}bd} + y_{q\dot{q}bd} = 1, \quad \forall q, \dot{q} \in Q_b \cap Q^{list}, \dot{q} \neq q, b \in B, \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \quad (S2)$$

$$y_{q\dot{q}bd} - y_{q\dot{q}bd} \geq 0, \quad \forall q \in P, \dot{q} \in I, \forall q, \dot{q} \in Q_b \cap Q^{list}, b \in B, \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \quad (S3)$$

$$s_{qd[\omega]} \geq S_b^B(\bar{x}_{qbd}), \quad \forall q \in Q^{list}, b \in B, \bar{x}_{qbd} = 1, \omega \in \Omega, \quad (S4)$$

$$s_{qd[\omega]} \leq (E_b^B + O_b^{\max}) - (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl}), \quad \forall q \in Q_b \cap Q^{list}, b \in B, \bar{x}_{qbd} = 1, \forall \omega \in \Omega, \quad (S5)$$

$$s_{\dot{q}d[\omega]} \geq s_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl} - M(1 - y_{q\dot{q}bd}), \quad \forall q, \dot{q} \in Q_b \cap Q^{list}, \dot{q} \neq q, b \in B, \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \omega \in \Omega, \quad (S6)$$

$$s_{\dot{q}d[\omega]} + T_{q[\omega]}^{Pre} \geq s_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S - M(1 - z_{q\dot{q}d}), \quad \forall q, \dot{q} \in P \cap Q^{list}, \dot{q} \neq q, \quad (S7)$$

$$S_q^P = S_{\dot{q}}^P, \forall b, \dot{b} \in B, \dot{b} \neq b, \forall b \in B_r^R, \forall \dot{b} \in B_r^R, \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}\dot{b}d} = 1, \forall \omega \in \Omega,$$

$$s_{qd[\omega]} + T_{q[\omega]}^{Pre} \geq s_{\dot{q}d[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S - M(z_{q\dot{q}d}), \quad \forall q, \dot{q} \in P \cap Q^{list}, \dot{q} \neq q, \quad (S8)$$

$$S_q^P = S_{\dot{q}}^P, \forall b, \dot{b} \in B, \dot{b} \neq b, \forall b \in B_r^R, \forall \dot{b} \in B_r^R, \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}\dot{b}d} = 1, \forall \omega \in \Omega,$$

$$s_{sd[\omega]}^S \leq s_{qd[\omega]} + T_{q[\omega]}^{Pre}, \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in B, \bar{x}_{qbd} = 1, \omega \in \Omega, \quad (S9)$$

$$e_{sd[\omega]}^S \geq s_{qd[\omega]} + (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S), \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in B, \bar{x}_{qbd} = 1, \omega \in \Omega, \quad (S10)$$

$$i_{sd[\omega]} \geq (e_{sd[\omega]}^S - s_{sd[\omega]}^S) - \sum_b \sum_{q \in P \cap Q^{list}, S_q^P = s} (T_{q[\omega]}^S + T_{q[\omega]}^O) x_{qbd}, \quad \forall s \in S, \omega \in \Omega, \quad (S11)$$

$$y_{q\dot{q}bd}, z_{q\dot{q}d} \in \{0,1\}, \quad \forall q, \dot{q} \in Q^{list}, b \in B, s \in S, \quad (S12)$$

$$s_{qd[\omega]}, s_{sd[\omega]}^S, e_{sd[\omega]}^S, i_{sd[\omega]} \geq 0, \quad \forall q \in Q^{list}, s \in S, \omega \in \Omega. \quad (S13)$$

In the same vein as for the ORP, the objective function and constraints of the ORS are similar to their counterparts in ORPS, except that they are written with consideration of the updated patients' list, Q^{list} , and they are compiled for a specific day. Accordingly, objective function (S1) is similar to objective function (1) in terms of the eighth and ninth components. Constraint (S2) is the equivalent of constraints (7) and (8), and constraints (S3)–(S11) are the counterparts of constraints (9)–(17), respectively.

III) Mathematical model of the ORs scheduling approximation problem (ORS-A)

In the second phase of the proposed heuristic solution framework, in the second approach (H2), instead of solving the *ORS*, due to CPU time considerations, an approximation of the ORs scheduling and sequencing model (*ORS-A*) is solved for each day based on the solution obtained in the planning phase. According to the context of scheduling and sequencing problems, the *ORS-A* is a *position-based* formulation in contrast to the *relation-based* formulations of the *ORPS* and the *ORS*. The mathematical formulation applied for such a scheduling approximation problem (*ORS-A*) is as follows:

$$Obj_{ORS-A} = \min (m_8 \sum_s i_{sd} + m_9 \sum_b \sum_{k \in K_b} \sum_{q: \bar{x}_{qbd}=1} k U_q x_{qbdk}) \quad (SA1)$$

Subject to:

$$\sum_{k_b} x_{qbdk} = 1, \quad \forall q \in Q_b \cap Q^{list}, b \in B, \bar{x}_{qbd} = 1, \quad (SA2)$$

$$\sum_{q \in \{Q_b \cap Q^{list} | \bar{x}_{qbd}=1\}} x_{qbdk} = 1, \quad \forall b \in B, k \in K_b, \quad (SA3)$$

$$\sum_k k x_{qbdk} + 1 \leq \sum_k k \bar{x}_{qbdk}, \quad \forall q \in P \cap Q^{list}, \dot{q} \in I \cap Q^{list}, b \in B: \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \quad (SA4)$$

$$s_{sd}^S \leq (k-1)(T_q^{Pre} + T_q^O + T_q^{Post} + T_q^{Cl}) + M(1 - x_{qbdk}), \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in \{B | S_b^B = 0\}, k \in K_b: \bar{x}_{qbd} = 1, \quad (SA5)$$

$$s_{sd}^S \leq \sum_{b \in \{B | b, \dot{b} \in B_r^R \wedge b \leq b-1\}} (E_b^B + O_b^{\max}) + (k-1)(T_q^{Pre} + T_q^O + T_q^{Post} + T_q^{Cl}) + M(1 - x_{qbdk}), \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in \{B | S_b^B \neq 0\}, k \in K_b: \bar{x}_{qbd} = 1, \quad (SA6)$$

$$e_{sd}^S \geq k(T_q^{Pre} + T_q^O + T_q^{Post} + T_q^{Cl}) x_{qbdk}, \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in \{B | S_b^B = 0\}, k \in K_b: \bar{x}_{qbd} = 1, \quad (SA7)$$

$$e_{sd}^S \geq \sum_{b \in \{B | b, \dot{b} \in B_r^R \wedge b \leq b-1\}} (E_b^B + O_b^{\max}) \bar{x}_{qbd} + k(T_q^{Pre} + T_q^O + T_q^{Post} + T_q^{Cl}) x_{qbdk}, \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, b \in \{B | S_b^B \neq 0\}, k \in K_b: \bar{x}_{qbd} = 1, \quad (SA8)$$

$$i_{sd} \geq (e_{sd}^S - s_{sd}^S) - \sum_b \sum_{q \in \{P \cap Q^{list} | S_q^P = s \wedge \bar{x}_{qbd}=1\}} (T_q^S + T_q^O), \quad \forall s \in S, \quad (SA9)$$

$$\sum_b \sum_{k \in \{K_b | \bar{x}_{qbdk}=1\}} x_{qbdk} + \sum_b \sum_{k \in \{K_b | \bar{x}_{\dot{q}bdk}=1\}} x_{\dot{q}bdk} \leq 1, \quad \forall q, \dot{q} \in Q^{list}, l \in L: \bar{z}_{q\dot{q}d}^l = 1, \quad (FC2)$$

$$x_{qbdk} \in \{0,1\}, \quad \forall q \in Q^{list}, b \in B, k \in K, \quad (SA10)$$

$$s_{sd}^S, e_{sd}^S, i_{sd} \geq 0, \quad \forall s \in S. \quad (SA11)$$

The objective function and constraints of the *ORS-A* are designated to approximate the *ORS*, but still, they are written for a specific day with consideration of the updated patients' list, Q^{list} . Accordingly,

objective function (SA1) is an estimation of objective function (1) in terms of the eighth and ninth components. Constraints (SA2) and (SA3) are assignment constraints of patients to positions and vice versa. Constraint (SA4) represents constraint (9). Constraints (SA5) and (SA6), (SA7) and (SA8), and (SA9) estimate constraints (15)–(17), respectively. Constraint (FC2) is added to the *ORS-A* as feasibility combinatorial cuts. They are attached to the *ORS-A* throughout the time the solution obtained by the *ORS-A* is not feasible, and by means of such cuts, the infeasible solution is excluded in succeeding iterations.

IV) Calculation model (*ORS-C*)

Solving the *ORS-A* and obtaining the solution for each day, it is needed to calculate the start time of the surgeries and the associated terms in the objective function. Thus, based on the solutions of the *ORP* and the *ORS-A*, a calculation model (*ORS-C*) is needed to be solved in this step.

$$Obj_{ORS-C} = \min \sum_{\omega} P_{[\omega]} \left\{ m_8 \sum_s i_{sd[\omega]} + m_9 \sum_b \sum_k \sum_{q \in Q^{list}: \bar{x}_{qbdk}=1} U_q s_{qd[\omega]} \right\} \quad (SC1)$$

Subject to:

$$s_{qd[\omega]} = S_b^B(\bar{x}_{qbdk}), \quad \forall q \in Q_b \cap Q^{list}, b \in B, \omega \in \Omega: k = 1, \bar{x}_{qbdk} = 1, \quad (SC2)$$

$$s_{qd[\omega]} \leq (E_b^B + O_b^{\max}) - (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl}), \quad \forall q \in Q_b \cap Q^{list}, b \in B, \omega \in \Omega: k \leq |K_b|, \bar{x}_{qbdk} = 1, \quad (SC3)$$

$$s_{\dot{q}d[\omega]} = s_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^{Post} + T_{q[\omega]}^{Cl}, \quad \forall q, \dot{q} \in Q^{list}, q \neq \dot{q}, \omega \in \Omega: b \in B, \omega \in \Omega: k \leq |K_b| - 1, \bar{x}_{qbdk} = 1, \bar{x}_{\dot{q}bdk+1} = 1, \quad (SC4)$$

$$s_{sd[\omega]}^S \leq s_{qd[\omega]} + T_{q[\omega]}^{Pre}, \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, \omega \in \Omega, \sum_k \bar{x}_{qbdk} = 1, \quad (SC5)$$

$$e_{sd[\omega]}^S \geq s_{qd[\omega]} + (T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S), \quad \forall s \in S, q \in \{P \cap Q^{list} | S_q^P = s\}, \omega \in \Omega, \sum_k \bar{x}_{qbdk} = 1, \quad (SC6)$$

$$i_{sd[\omega]} \geq (e_{sd[\omega]}^S - s_{sd[\omega]}^S) - \sum_b \sum_{q \in \{P \cap Q^{list} | S_q^P = s \wedge \bar{x}_{qbd} = 1\}} (T_{q[\omega]}^S + T_{q[\omega]}^O), \quad \forall s \in S, \omega \in \Omega, \quad (SC7)$$

$$s_{qd[\omega]}, s_{sd[\omega]}^S, e_{sd[\omega]}^S, i_{sd[\omega]} \geq 0, \quad \forall q \in Q^{list}, s \in S, \omega \in \Omega. \quad (SC8)$$

Objective function (SC1) calculates the eighth and ninth terms in objective function (1), and constraints (SC2)–(SC7) calculate the associated variables in constraints (10)–(12), (15)–(17).

V & VI) Feasibility checking of the mathematical models (*CHECK1* & *CHECK2*)

As the final step of the second solution approach (*H2*), it is needed to be sure that the solution obtained by the *ORS-A* is feasible. Thus, feasibility checking models are recruited. First, *CHECK1* is solved. If

the objective function of the *CHECK1* takes a value greater than zero, it means either the solution of the *ORP* or the *ORS-A* is infeasible in terms of the overlaps between surgeries of a surgeon in different OR blocks **on** a day. Thus, *CHECK2* is solved to recognize the source of the infeasibility. If *CHECK2* is infeasible, the source of the infeasibility lies in the patients' planning phase (*ORP*). Otherwise, it lies in the patients' sequencing phase (*ORS-A*). In either case, feasibility cuts of type (FC1) and (FC2) are added to the *ORP* and the *ORS-A*, respectively, in order to exclude such infeasible solutions. These steps of solving the *ORS-A*, the *ORS-C*, the *CHECK1*, and the *CHECK2* are iterated until the solution framework comes up with a feasible solution.

(*CHECK1*):

$$Obj_{CHECK1} = \min \sum_{q \in Q^{list}} \sum_{\dot{q} \in Q^{list}} z_{q\dot{q}d} \quad (C1)$$

Subject to:

$$\bar{s}_{\dot{q}d[\omega]} + T_{\dot{q}[\omega]}^{Pre} \geq \bar{s}_{qd[\omega]} + T_{q[\omega]}^{Pre} + T_{q[\omega]}^O + T_{q[\omega]}^S - M(z_{q\dot{q}d}), \quad \forall q, \dot{q} \in P \cap Q^{list}, \dot{q} \neq q, S_q^P = S_{\dot{q}}^P, \\ b, \dot{b} \in B, \dot{b} \neq b, b \in B_r^R, \dot{b} \in B_r^R, \omega \in \Omega: \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \bar{s}_{\dot{q}d[\omega]} \geq \bar{s}_{qd[\omega]}, \quad (C2)$$

$$z_{q\dot{q}d} \in \{0,1\}, \quad \forall q, \dot{q} \in Q, s \in S. \quad (C3)$$

(*CHECK2*):

$$Obj_{CHECK2} = \min 0 \quad (C'1)$$

$$\text{Subject to: (S2)–(S8), S(12),} \quad \forall q, \dot{q} \in P \cap Q^{list}, \dot{q} \neq q, S_q^P = S_{\dot{q}}^P, b, \dot{b} \in B, \dot{b} \neq b, b \in B_r^R, \\ \dot{b} \in B_r^R, \omega \in \Omega: \bar{x}_{qbd} = 1, \bar{x}_{\dot{q}bd} = 1, \quad (C'2)$$

$$s_{qd[\omega]} \geq 0, \quad \forall q \in Q^{list}, \omega \in \Omega. \quad (C'3)$$

Appendix B. the validity proofs of feasibility combinatorial cuts (FC1) and (FC2)

In this section, the validity of the proposed *feasibility combinatorial cuts* (FC1) and (FC2) are discussed. It is worth noting that even by proving the validity of constraints (FC1) and (FC2), the nature of the proposed methodologies will not change from heuristic to exact. Breaking down the solution approach into a 2-phase one, we cannot claim that the approach is an exact one. In other words, in the absence of the related terms with the focus of surgeons' idle time and surgeries' start time in the objective function, the proposed constraints (FC1) and (FC2) could play the role of optimality cuts, but in the presence of such terms, this is not the case.

In order to prove the validity of constraint (FC1), we first show that constraint (FC1) cuts off the current infeasible *ORP* solution from the feasible set, but keeps all the other feasible solutions. Let Q^{list} be the set of patients, updated each day, and $\bar{x}_{qbd}^l$ be the given solution which is infeasible through *ORS*, in

terms of the overlaps between surgeries of a surgeon in different OR blocks on a day, in the previous iteration ($l \in L$). In other words, $\bar{x}_{qbd}^l$ includes a set (or some sets) of paired patients of a specific surgeon that causes the overlaps through his/her surgeries. It is noteworthy that those pairs of patients who caused infeasibility are known based on $z_{q\dot{q}d}^l$. Therefore, the current *ORP* solution with $\bar{x}_{qbd}^l$ does not satisfy constraint (FC1) since for each pair of patients causing overlaps, the left side of the inequality is 2 and it is not less than or equal to 1.

$$\sum_{b \in \{B | \bar{x}_{qbd}^l = 1\}} x_{qbd} + \sum_{\dot{b} \in \{B | \bar{x}_{\dot{q}\dot{b}d}^l = 1\}} x_{\dot{q}\dot{b}d} = 2 \not\leq 1, \quad \forall q, \dot{q} \in Q^{list}, l \in L, \quad (\text{FC1})$$

As for the next step, it is needed to show that constraint (FC1) does not remove any globally feasible solution which is quite trivial. This constraint is just applicable for a specific pair of patients, who are planned on a specific day in a specific block time, and just cuts off the current infeasible *ORP* solution from the feasible set, but keeps all the other feasible solutions. ■

The same proving procedure should be carried out for constraint (FC2). We first show that constraint (FC2) cuts off the current infeasible *ORS-A* solution from the feasible set, but keeps all the other feasible solutions. Let $\bar{z}_{q\dot{q}d}^l = 1$ indicate the pair of patients of a specific surgeon who caused the infeasibility, and $\bar{x}_{qbdk}^l$ be the given solution which is infeasible as detected by the *CHECK1* and *CHECK2* models. Similarly to constraint (FC1), here also infeasibility is in terms of the overlaps between surgeries of a surgeon in different OR blocks on a day, in the previous iteration ($l \in L$). Therefore, the current *ORS-A* solution with $\bar{x}_{qbdk}^l$ does not satisfy constraint (FC2) since for each pair of patients causing overlaps, the left side of the inequality is 2 and it is not less than or equal to 1.

$$\sum_b \sum_{k \in \{K_b | \bar{x}_{qbdk}^l = 1\}} x_{qbdk} + \sum_{\dot{b}} \sum_{\dot{k} \in \{K_{\dot{b}} | \bar{x}_{\dot{q}\dot{b}\dot{d}\dot{k}}^l = 1\}} x_{\dot{q}\dot{b}\dot{d}\dot{k}} = 2 \not\leq 1, \quad \forall q, \dot{q} \in Q^{list}, l \in L: \bar{z}_{q\dot{q}d}^l = 1, \quad (\text{FC2})$$

As for the next step, it is needed to show that constraint (FC2) does not remove any globally feasible solution which is quite obvious. In the same vein as for constraint (FC1), this constraint is just applicable for a specific pair of patients, who are planned for on a specific day in a specific block time on a specific position, and just cuts off the current infeasible *ORS-A* solution from the feasible set, but keeps all the other feasible solutions. ■

References

1. Addis B, Carello G, Grosso A, Tanfani E. (2016). Operating room scheduling and rescheduling: a rolling horizon approach. *Flexible Services and Manufacturing Journal*, 28(1-2), 206–232.
2. Bam M, Denton BT, van Oyen MP, Cowen ME. (2017). Surgery scheduling with recovery resources. *IIE Transactions*, 49(10), 942–955.
3. Batun S, Denton BT, Huschka TR, Schaefer AJ. (2011). Operating Room Pooling and Parallel Surgery Processing Under Uncertainty. *INFORMS Journal on Computing*, 23(2), 220–237.
4. Belien J, Demeulemeester E. (2008). A branch-and-price approach for integrating nurse and surgery scheduling. *European Journal of Operational Research*, 189(3), 652–668.
5. Cardoen B, Demeulemeester E, Beliën J. (2009). Sequencing surgical cases in a day-care environment: an exact branch-and-price approach. *Computers & Operations Research*, 36(9), 2660–2669.
6. Cardoen B, Demeulemeester E, Beliën J. (2010). Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, 201(3), 921–932.
7. Dellaert NP, Dashty Saridarq F, van Woensel T, Crainic TG. (2018). Branch & price based algorithms for the two-echelon vehicle routing problem with time windows (In press) In: *Transportation Science*.
8. Denton BT, Gupta D. (2003). A sequential bounding approach for optimal appointment scheduling. *IIE Trans*, 35, 1003–1016.
9. Denton BT, Miller A, Balasubramanian HJ, Huschka TR. (2010). Optimal Allocation of Surgery Blocks to Operating Rooms Under Uncertainty. *Inform Operations Research*, 58(4), 802–816.
10. Denton BT, Viapiano V, Voglet A. (2007). Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Science*, 10, 13–24.
11. Etzioni D, Liu J, Maggard M, Ko C. (2003). The aging population and its impact on the surgery workforce. *Annals of Surgery*, 238, 170–177.
12. Fei H, Chu C, Meskens N, Artiba A. (2008). Solving surgical cases assignment problem by a branch-and-price approach. *International Journal of Production Economics*, 112, 96–108.
13. Fei H, Chu C, Meskens N. (2009). 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.

14. Fei H, Meskens N, Chu C. (2006). An operating theatre planning and scheduling problem in the case of a block scheduling strategy. In: Service systems and service management, 2006 international conference, 1, 422–428.
15. Fei H, Meskens N, Chu C. (2010). A Planning and Scheduling Problem for an Operating Theatre Using an Open Scheduling Strategy. *Computers & Industrial Engineering*, 58(2), 221–230.
16. Ferrand YB, Magazine MJ, Rao US. (2014). Managing operating room efficiency and responsiveness for emergency and elective surgeries—A literature survey. *IIE Transactions on Healthcare Systems Engineering*, 4, 49–64.
17. Goedhart CE. (2014). Reliable and Acceptable Planning of Elective Patients for General Surgery on the Day of Diagnosis. MSc thesis, Department of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, Netherlands.
18. Gordon T, Lyles A, Fountain J. (1988). Surgical unit time review: Resource utilization and management implications. *Journal of Medical Systems*, 12, 169–179.
19. Guerriero F, Guido R. (2011). Operational research in the management of the operating theatre: a survey. *Health Care Management Science*, 14, 89–114.
20. Guinet A, Chaabane S. (2003). Operating theatre planning. *International Journal of Production Economics*, 85(1), 69–81.
21. Hashemi Doulabi SH, Rousseau LM, Pesant G. (2014). A Constraint Programming-Based Column Generation Approach for Operating Room Planning and Scheduling. 8451, 455–463, New York: Springer.
22. Healthcare Financial Management Association. (2003). Achieving operating room efficiency through process integration. *Healthcare Financial Management*, 57(3, Supplement) 1–8.
23. Hooker, J. N. 2000. *Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction*. John Wiley and Sons, New York.
24. Hooker, J. N., G. Ottosson. 2003. Logic-based Benders decomposition. *Math. Programming*, 96 33–60.
25. Jebali A, Alouane ABH, Ladet P. (2006). Operating room scheduling. *International Journal of Production Economics*, 99(1), 52–62.
26. Jebali A, Diabat A. (2017). A chance-constrained operating room planning with elective and emergency cases under downstream capacity constraints. *Computers & Industrial Engineering*, 114, 329–344.

27. Kamran MA, Karimi B, Dellaert N. (2018a). A Column-Generation-heuristic-based Benders' decomposition for solving Adaptive Allocation Scheduling of patients in operating rooms. *Computers & Industrial Engineering*, Under 1st revision.
28. Kamran MA, Karimi B, Dellaert N. (2018b). Uncertainty in Advance Scheduling Problem in Operating Room Planning. *Computers & Industrial Engineering*, 126, 252–268.
29. Lamiri M, Xie X, Zhang S. (2008). Column generation approach to operating theater planning with elective and emergency patients. *IIE Transactions*, 40(9), 838–852.
30. Latorre-Núñez G, Lüer-Villagra R, Marianov V, Obreque C, Ramis F, Neriz L. (2016). Scheduling operating rooms with consideration of all resources, post anesthesia beds and emergency surgeries. *Computers & Industrial Engineering*, 97, 248–257.
31. Magerlein J, Martin J. (1978). Surgical demand scheduling: a review. *Health Services Research*, 13(4), 418–433.
32. Marques I, Captivo M, Vaz Pato M. (2012). An integer programming approach to elective surgery scheduling. *OR Spectrum*, 34(2), 407–427.
33. Marques I, Captivo ME, Pato MV. (2014). Scheduling elective surgeries in a Portuguese hospital using a genetic heuristic. *Operations Research for Health Care*, 3(2), 59–72.
34. May JH, Strum DP, Vargas LG. (2000). Fitting the lognormal distribution to surgical procedure times. *Decision Sciences*, 31(1), 129–148.
35. Meskens N, Duvivier D, Hanset A. (2013). Multi-objective operating room scheduling considering desiderata of the surgical team. *Decision Support Systems*, 55: 650–659.
36. Min D, Yih Y. (2010). Scheduling elective surgery under uncertainty and downstream capacity constraints. *European Journal of Operational Research*, 206, 642–652.
37. Molina-Pariente JM, Fernandez-Viagas V, Framinan JM. (2015). Integrated operating room planning and scheduling problem with assistant surgeon dependent surgery durations. *Computers & Industrial Engineering*, 82, 8–20.
38. Noorizadegan M, Seifi A. (2018). An efficient computational method for large scale surgery scheduling problems with chance constraints. *Computational Optimization and Applications*, 69(2), 535–561.
39. Pham DN, Klinkert A. (2008). Surgical case scheduling as a generalized job shop scheduling problem. *European Journal Operational Research*, 185(3), 1011–1025.
40. Pulido R, Aguirre AM, Ortega-Mier M, García-Sánchez A, Méndez CA. (2014). Managing daily surgery schedules in a teaching hospital: a mixed-integer optimization approach. *BMC Health Services Research*, 14, 1–13.

41. Rachuba S, Werners B. (2017). A fuzzy multi-criteria approach for robust operating room schedules. *Annals of Operations Research*, 251(1-2), 325–350.
42. Roland B, Di Martinelly C, Riane F, Pochet Y. (2010). Scheduling an operating theatre under human resource constraints. *Computers & Industrial Engineering*, 58(2), 212–220.
43. Roshanaei V, Luong C, Aleman D, Urbach D. (2017a). Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling. *European Journal Operational Research*, 257(2), 439–455.
44. Roshanaei V, Luong C, Aleman D, Urbach D. (2017b). Collaborative Operating Room Planning and Scheduling. *INFORMS Journal on Computing*, 29(3), 558-580.
45. Samudra M, van Riet C, Demeulemeester E, Cardoen B, Vansteenkiste N, Rademakers FE. (2016). Scheduling operating rooms: achievements, challenges and pitfalls. *Journal of Scheduling*, 19(5), 493–525.
46. Van Riet C, Demeulemeester E. (2015). Trade-offs in operating room planning for electives and emergencies: A review. *Operations Research for Health Care*, 7, 52-69.
47. Xiang W, Yin J, Lim G. (2015). An ant colony optimization approach for solving an operating room surgery scheduling problem. *Computers & Industrial Engineering*, 85, 335–345.
48. Xiao G, van Jaarsveld W, Dong M, van de Klundert J. (2016). Stochastic programming analysis and solutions to schedule overcrowded operating rooms in China. *Computers & Operations Research*, 74, 78–91.
49. Wang Y, Tang J, Fung R. (2014). A column-generation-based heuristic algorithm for solving operating theater planning problem under stochastic demand and surgery cancellation risk. *International Journal of Production Economics*, 158, 28–36.
50. Wang S, Roshanaei V, Aleman D, Urbach D. (2016). A discrete event simulation evaluation of distributed operating room scheduling. *IIE Transactions on Healthcare Systems Engineering*, 6(4), 236–245.
51. Zhou J, Dexter F. (1998). Method to assist in the scheduling of add-on surgical cases-upper prediction bounds for surgical case durations based on the log-normal distribution. *Anesthesiology*, 89(5), 1228–1232.
52. Zhu S, Fan W, Yang S, Pei J, Pardalos PM. (2019). Operating room planning and surgical case scheduling: a review of literature. *Journal of Combinatorial Optimization*, 37(3), 757–805.