JOURNAL OF THE OPERATIONAL RESEARCH SOCIETY
https://doi.org/10.1080/01605682.2020.1718559

ORIGINAL ARTICLE

Stochastic surgery selection and sequencing under dynamic emergency break-ins

Mathieu Vandenberghe[a], Stijn De Vuyst[a,b], El-Houssaine Aghezzaf[a,b] and Herwig Bruneel[c]

[a]Department of Industrial Systems Engineering and Product Design, Ghent University, Zwijnaarde, Belgium; [b]Industrial Systems Engineering (ISyE), Flanders Make, Ghent, Belgium; [c]Department of Telecommunication and Information Processing, Ghent University, Ghent, Belgium

ABSTRACT

Anticipating the impact of urgent emergency arrivals on operating room schedules remains methodologically and computationally challenging. This paper investigates a model for surgery scheduling, in which both surgery durations and emergency patient arrivals are stochastic. When an emergency patient arrives he enters the first available room. Given the sets of surgeries available to each operating room for that day, as well as the distributions of the main stochastic variables, we aim to find the per-room surgery sequences that minimise a joint objective, which includes over- and under-utilisation, the amount of cancelled patients, as well as the risk that emergencies suffer an excessively long waiting time. We show that a detailed analysis of emergency break-ins and their disruption of the schedule leads to a lower total cost compared to less sophisticated models. We also map the trade-off between the threshold for excessive waiting time, and the set of other objectives. Finally, an efficient heuristic is proposed to accurately estimate the value of a solution with significantly less computational effort.

ARTICLE HISTORY
Received 30 April 2019
Accepted 12 January 2020

KEYWORDS
Operating room scheduling; emergency arrivals; uncertain surgery duration; sample average approximation; genetic algorithm

1. Introduction

The growing demand on healthcare services in the developed world has fuelled a vast amount of research in providing care more efficiently and effectively, much of it within the context of Operations Research. A particular focus has been put on the operating room (OR), as it forms a central nexus in the hospital, while also being responsible for a large share of expenses. Despite these research efforts, however, surveys have found that the level to which research solutions are implemented, remains low (Brailsford, Harper, Patel, & Pitt, 2009). Van Riet and Demeulemeester (2015) note in their recent literature review on surgery scheduling that if Operations Research models are to become widely adopted, it is crucial that they tackle the components that differentiate traditional (industrial) scheduling models from those used in the OR. The authors specify the arrival of emergency patients as such a component; while last-minute orders may exist in an industrial context, they rarely have the same level of urgency as medical emergencies.

However, emergency patients are no monolith. Most patients who arrive at an emergency department do not suffer from life-threatening conditions, and their treatment can be deferred for several hours. In contrast, a minority of cases are highly urgent, and the imperative of beginning emergency surgery within 1 h of diagnosis is established by various triage systems (Eitel, Travers, Rosenau, Gilboy, & Wuerz, 2003). Emergency classification systems and other regulations often stipulate a maximum waiting time for patients (Eitel et al., 2003), especially for "emergent" patients, the highest category of urgency. The link between rapid intervention and medical outcomes is further expressed by the maxim of "the golden hour" (Fleet & Poitras, 2011; Newgard et al., 2010).

In general, the criticality of timely interventions for emergent patients is well established in the medical literature but rarely featured in the OR planning literature. The following papers represent some notable exceptions. The BIM (Break-In-Moment) model articulated by van Essen, Hans, Hurink, and Oversberg (2012) optimises the sequencing of surgeries in anticipation of an emergent patient, by minimising the maximum interval between any two surgery completion times. This establishes a worst-case bound on the maximum time until any OR becomes available, allowing the emergency to "break into" the schedule without interrupting the previous surgery. This first investigation into bounding emergency waiting times used the assumption of deterministic surgery times, and made no assumptions regarding emergency arrivals. Our previous paper on the Stochastic BIM problem (Vandenberghe, De Vuyst, Aghezzaf, & Bruneel, 2019)

CONTACT Mathieu Vandenberghe mathieu.vandenberghe@ugent.be Department of Industrial Systems Engineering and Product Design, Ghent University, Technologiepark 903, 9052 Zwijnaarde, Belgium

extended this problem to the more general case where surgery durations are only known stochastically, however, the actual break-in of emergencies and their impact, i.e. the way they disrupt the intended OR schedule, remained unexplored. In this paper, our intention is to include and investigate this dynamic component. Latorre-Núñez et al. (2016) also sought to restrict the worst-case waiting time for emergencies by penalising excessive distance between consecutive completion times, but without considering the stochasticity of emergency arrivals themselves. Paul and MacDonald (2013) used non-preemptive multi-priority queuing models to determine the optimal number of ORs needed to ensure that emergent patients (of various levels of severity) do not exceed their acceptable waiting time. This study, however, focused exclusively on OR performance related to emergent patients, rather than including elective patients as well. To the best of our knowledge, it has not been investigated how to create surgery schedules where the "golden hour" is robust for a series of stochastically arriving emergency patients, and under general stochastic surgery durations.

Anticipating the arrival of emergencies is hardly the only focus of OR planning, however. The schedule used by the OR department has multiple downstream effects on the intensive care unit, the nursing schedule, the amount of hospital beds required, etc. As a consequence, OR planning must consider a number of objectives: over- and under-utilisation of resources, overtime, risk of cancellation, as well as sequencing constraints (e.g. children are generally planned early; infected patients are planned at the end). However, as alluded to earlier, break-ins by emergent patients can have a disruptive impact on these objectives as well: additional surgeries enter the schedule and receive priority, which can lead to subsequent elective surgeries being delayed or cancelled. This then again affects the break-in potential for future emergencies. Unfortunately, emergent patients are rarely modelled in enough detail to capture this impact.

Our goal in this paper is to optimise a subset of the more common performance criteria for OR scheduling (expected over-utilisation, under-utilisation and patient cancellations), while taking into account the dynamic disruptions caused by emergencies. At the same time, we ensure that the minimal requirements for emergency waiting times are not exceeded, by adding waiting cost penalties to the objective function. The combination of these two elements has not been investigated yet, and represents a more fine-grained analysis of the impact of emergencies than is common.

This paper investigates the resulting scheduling problem, where the focus is on the creation of a one-day operational schedule. We start from a large set of patients (waiting list) that would ideally be scheduled on the following day, and we decide which surgeries should be performed, and in what order. The objective is to minimise the total operating cost, which includes over-utilisation, under-utilisation, cancellation risk, as well as the need to service emergent patients before a certain time threshold is exceeded. One limitation is that we do not include the assignment of surgeries to ORs. This is because, in practice, higher-level schedules such as a master surgical schedule (Hans & Vanberkel, 2012; Vanberkel et al., 2011) typically pre-assign OR time blocks to certain surgical specialties. As each patient belongs to one surgical specialty, the possibility of swapping patients between rooms tends to be limited. We thus work on the assumption that each OR has a dedicated set of surgeries available for scheduling. This mimics the reality of waiting lists: the set of surgeries assigned to each OR represents a larger workload than they can (on average) process in one day. The scheduling decisions thus comprise the *selection* of surgeries from each OR's waiting list, and the *sequence* of performing them within each room. We further assume that all ORs are equipped to deal with emergent patients, and so due to their urgency, the only criterion for their OR assignment will be which room becomes available first.

Our contribution is three-fold: (i) modelling the various stochastic components of emergent arrivals and associated break-in mechanics, and proposing solution methods for this problem (ii) estimating the stochastic value of a detailed modelling of emergent patients, as contrasted with more common methods of modelling emergency impact (iii) developing an efficient heuristic that allows practitioners to accurately estimate the impact of emergencies on a schedule, without needing to replicate our entire model.

2. Related literature

Given its importance in the hospital's value chain, the scheduling of ORs has been analysed from multiple perspectives. Typically, OR planning is divided into three subproblems, each one associated with a different decision level (Testi, Tanfani, & Torre, 2007). The strategic level forms the most long-term view, and sets the number of ORs, staff size, and general hospital policies. The tactical plan is medium-term, and typically deals with the amount of rooms and working hours assigned to various specialties over weeks or months. Finally, the operational level covers with both the creation and adaptation of short-term (often daily) surgery schedules;

referred to as offline operational and online operational, respectively. As this manuscript focuses on the creation of daily schedules, our literature review centres on contributions to offline operational planning, and in particular to how emergencies were modelled.

We briefly survey the field of online operational planning first, however, as it often contains the most detailed modelling of emergency patients. The online operational level deals with adapting daily schedules to unexpected events, which may include finding the least costly way to adjust an ongoing schedule in the face of emergency arrivals. Erdem, Qu, and Shi (2012) were one of the first authors to tackle this real-time rescheduling aspect. They proposed a Mixed Integer Program (MIP) and a genetic algorithm that support rescheduling decisions upon the arrival of emergency patients, over a planning horizon of a few days. Surgery on emergency patients must begin at the moment of arrival, and total cost calculations determine which OR will be required to free up its schedule. van Essen, Hurink, Hartholt, and van den Akker (2012) developed a set of decision support systems in the case of emergency arrivals. For any emergency arrival, the system returns three options for rescheduling, which each balance the competing desires of the various stakeholders in different ways. While the online operational level provides a necessary layer for immediate adjustments to an existing schedule, it provides no guidance for creating an initial schedule that *anticipates* the arrival of emergency patients. This remains the purview of the offline operational level, to which we turn now.

The offline operational layer deals with the construction of daily schedules over a planning horizon, so the layer can be broadly divided into first assigning patients a provisional surgery date (referred to as advance scheduling) and then finalising the daily schedules (allocation scheduling). Allocation scheduling thus involves OR assignment, decisions to rebalance surgery loads, and the precise sequence and starting times of surgeries. While many papers (including ours) lie squarely within one category, others straddle both or have sought to combine them. Below we list recent contributions that involve (but may not be limited to) selection and sequencing decisions, as these form the best comparison for our specific planning problem. Landa, Aringhieri, Soriano, Tànfani, and Testi (2016) created a joint optimisation model that assigns surgery dates and OR blocks to patients at a high level, and then determines the optimal room sequences, in order to balance the trade-off between OR utilisation and surgery cancellations. The problem is solved using a combination of neighbourhood search techniques and Monte Carlo simulation. The model recently proposed by Moosavi and Ebrahimnejad (2018) consists of a set of MIPs which roughly correspond to the various operational planning stages, each with its own objective function. The first minimises the deferral of patients to the next planning period, the second minimises the waiting cost of scheduled patients, and the last adjusts sequencing to minimise idleness and OR overtime. Molina-Pariente, Fernandez-Viagas, and Framinan (2015) address a scheduling problem where surgeries are performed by a team of surgeons, and where their respective experience influences the surgery duration. The authors developed both a MIP model and an approximate algorithm which tackle date and OR assignment, as well as sequencing. Meskens, Duvivier, and Hanset (2013) created a model that includes the real-life constraints of availability, staff preferences, and affinities among members. Sequencing decisions are made in order to minimise makespan and overtime. Recently, Eun, Kim, Yih, and Tiwari (2018) considered a single-OR system in which patients must be assigned a date of surgery and a sequence number in the OR. Elective patients each have a time-dependent health status, which may deteriorate if their case is delayed; emergency patients are, however, not considered.

Next, we wish to survey offline operational planning papers that have specifically modelled emergency patients; for this, both advance and allocation scheduling papers may be of interest. As described in the introduction, few papers have considered the criticality of timely interventions for emergent patients, or sought to ensure them. Yet researchers are well aware of the *impact* of emergency patients: their arrivals can significantly disturb the operational surgery schedule by delaying other patients and taking up extra capacity and resources. This may in turn cause OR teams to run overtime (incurring personnel costs and dissatisfaction) or cancel some elective patients and postpone them to another day (reducing patient satisfaction). As reducing overtime and the likelihood of patient cancellations are central objectives in OR scheduling, several papers have included emergencies in their analysis. In Jebali and Diabat (2017), elective surgeries must be assigned intervention dates within a planning horizon, but sufficient resources (OR capacity and recovery beds) must be reserved to satisfy a daily stochastic demand of emergency capacity. A two-stage chance-constrained stochastic program is used to minimise costs while limiting the risk of surgery cancellations. Rachuba and Werners (2014) propose a multi-objective approach that takes the differing needs of patients, staff and management into account. As the study is focused on a longer planning horizon, emergency demand is modelled per day as a stochastic demand, which may be spread out over several ORs as required. The model of Adan, Bekkers, Dellaert, Jeunet, and Vissers (2011) creates tactical master plans that are also feasible at the operational level, when accounting for both elective and emergency patients. An estimate of emergency patients informs the creation of the operational schedule, and once it is in motion, emergency patients may be admitted to ORs (leading to potential cancellations of elective surgeries) or be deferred to other hospitals. Emergency arrivals do not have a precise arrival time during the day, but do have an internal sequence; later emergency arrivals are more likely to be deferred. The extensive patient admission model of Ceschia and Schaerf (2016) schedules patients across an extended planning horizon, and its objective function assigns costs to OR over- and under-utilisation, and the use of ORs by various specialties. A local search method is used to come to solutions, even in a dynamic environment where unexpected events occur; these events include the arrival of a stochastic number of urgent patients, which must receive surgery in any OR on that day. In Molina-Pariente, Hans, and Framinan (2016), the authors focus on an open scheduling strategy, starting from a waiting list that exceeds the total OR time available in the planning horizon. Their goal is to assign an intervention date and an OR in a way that minimises undertime and overtime costs, and surgery cancellations. During a Monte Carlo simulation, a number of emergency patients are generated each day, who are then randomly assigned across the available ORs.

The above papers show two broad strategies to depict emergencies: modelling them as a single amount of emergency capacity which must be fit across the various ORs (Adan et al., 2011; Jebali & Diabat, 2017; Rachuba & Werners, 2014), or modelling them as discrete patients who require intervention sometime during the date of admission (Ceschia & Schaerf, 2016; Molina-Pariente et al., 2016). Both of these methods help to model the impact of emergencies on total available capacity, but do not consider the time-sensitive needs of emergent patients and the (also time-specific) resultant disruptions they cause. Offline operational models in which emergency patients have a more specific arrival time are comparatively rare. Moosavi and Ebrahimnejad (2018) use a robust optimisation approach to predict the capacity required for emergency patients. Emergency patients must be scheduled in the same time block (of which there are 2–5 per day) as they arrive, but there is no decision process for what room they enter. Duma and Aringhieri (2015) create a simulation model to solve the online rescheduling problem. They discuss an extension of their main model to deal with time-specified emergency surgeries as well, which tracks (but does not optimise) the number of emergencies operated on within 1 h.

3. Model formulation

This section expounds the various components used in our model. We begin by discussing the mechanics of how emergency arrivals disrupt the initial schedule, and the assumptions regarding emergency distributions. We then address the various components of the objective function, and the impact of some related design choices. Figure A1 serves as a visual illustration of the main model components.

3.1. Model dynamics

Let $\bar{\mathcal{I}}$ be the set of patients who require elective surgery (with $|\bar{\mathcal{I}}| = M$), which are available to be scheduled on that day in one of K ORs, $\mathcal{K} = \{1, \ldots, K\}$. OR k has an available time capacity D_k, though each is permitted some allowed overtime D_k^{OT}. We will assume that ORs open at the same time ($t = 0$), and a regular workday lasts for 8 h ($t = 8$). Unless stated otherwise, we initialise $D_k = 8\ , k \in \mathcal{K}$; i.e. each OR is available for a full workday. Allowed overtime is initialised to $D_k^{OT} = 0.75$.

The assignment of surgeries to ORs is assumed to be given, for reasons specified in Sections 1 and 2. That is, $\bar{\mathcal{I}}$ is already partitioned in subsets $\bar{\mathcal{I}}_k, k \in \mathcal{K}$, being the patients assigned to OR k. Not all available surgeries must be scheduled, however, we distinguish between the set $\mathcal{I}$ of surgeries taken up in the day's operational schedule, and the set $\mathcal{I}^Q$ of annulled patients, where $\mathcal{I} \cup \mathcal{I}^Q = \bar{\mathcal{I}}$. The selection of surgeries to either $\mathcal{I}$ or $\mathcal{I}^Q$ is part of the model's decision variables.

The problem is specified by several random variables. First, we model surgery durations of elective patients $\mathbf{P} = (P_1, \ldots, P_M)$ as independent random variables, each having a known distribution with $\mu_i = \mathrm{E}[P_i]$ and $\sigma_i^2 = \mathrm{Var}[P_i], i \in \bar{\mathcal{I}}$. Second, a number of emergencies J will arrive during schedule execution, which is a random variable with a known distribution. Third, for each emergency $j \in \mathcal{J} = \{1, \ldots, J\}$, let A'_j be the arrival time of emergency j with density $f(t)$ and distribution function $F(t) = \mathrm{Prob}[A'_j \leqslant t]$, where it is assumed that $F(t)$ has no discontinuities in its domain. We assume these arrival times A'_j to be independent and identically distributed. Then, let $\mathbf{A} = (A_1, \ldots, A_J)$ be the increasing order statistics of $\mathbf{A}' = (A'_1, \ldots, A'_J)$, i.e. we have, almost surely, $A_1 < A_2 < \ldots < A_J$. So to be clear, A_j is the arrival time of the jth emergency that arrives during the day, not necessarily the arrival time of emergency j. Finally, the emergency surgery durations $P_j^e, j \in \mathcal{J}$ are also independent and identically distributed with a known distribution.

Surgeries are scheduled according to the (initial) global schedule $\pi = (\pi^1, ..., \pi^K) \in \Pi$. The schedule π is the main decision variable, and π^k forms a *partial* permutation of the surgery set $\bar{\mathcal{I}}_k$ allocated to OR k. Each room schedule π^k thus consists of:

1. A *selection* decision, namely the partitioning of the surgery set $\bar{\mathcal{I}}_k$ into a set $\mathcal{I}_k$ of patients selected for surgery and a set $\mathcal{I}_k^Q$ of surgeries that are annulled.
2. A *sequencing* decision, namely a permutation of the selected surgeries $\mathcal{I}_k$.

Time in the OR is highly valuable, so we assume that surgeries take place contiguously without intentional idle time. Given the surgery durations $\mathbf{P}$, any particular schedule of surgeries π then results in a set of completion times. However, the break-in of emergency patients in specific rooms may delay subsequent elective surgeries, disrupting the initial schedule π. Completion times are thus characterised as C_{ij}, $i \in \mathcal{I}, j \in \mathcal{J}$, which is the completion time of surgery i after the arrival of j emergencies. Note that C_{i0} are the completion times under the condition that no emergencies have arrived. Figure A1(c) illustrates how $J = 1$ emergency disrupts an initial schedule.

We then define the BIM sequence B_{ij} as the completion times C_{ij} in ascending order. So, because of the reordering, BIM B_{ij} is not necessarily related to the completion time of surgery i. The intervals between these BIMs form the Break-In-Intervals (BII)

$$S_{ij} = B_{ij} - B_{i-1,j} \ , B_{0j} = 0 \ , i \in \mathcal{I} \ , j \in \mathcal{J}.$$

The BIMs constitute potential times when emergency patients can break into the operating schedule. From the perspective of each emergency, it is not known when the next BIM will occur; but once it does, the assignment is immediate. By default, emergency patient j takes advantage of the first BIM after its arrival time, denoted by

$$B_j^e = \min\{B_{ij} : i \in \mathcal{I} \ , \ B_{ij} > A_j\} \ , j \in \mathcal{J}.$$

The jth emergency will therefore incur a waiting time $W_j = B_j^e - A_j$. Furthermore, future completion times in the corresponding OR will be incremented by the emergency surgery duration P_j^e. Note that while multiple emergency patients may be waiting for a BIM, only one (the earliest arrival) can take advantage of it: the next possible BIM in that room takes place when the emergency surgery is completed.

The mechanic of emergencies entering the first available room creates an interdependency between the various ORs. In Figure A2, we take a schedule π, and focus only on one particular room. Within this room, we generate different sequences of the elective surgeries. When excluding emergencies, the different sequences do not change the total workload in that room. When we do include emergencies, the break-in rules above imply that a certain room sequence may increase or decrease the chance that the room will have to service an emergency. This in turn affects other ORs. This interdependence renders a room-by-room decomposition of the problem impossible.

3.2. Emergency arrival process

The use of a homogeneous Poisson process is likely to be a good fit for the day-to-day arrival variability of emergent patients, and its use for this purpose is established (Cardoen, Demeulemeester, & Beliën, 2010). However, no similar convention exists for a nonstationary arrival process with an arrival rate that changes from hour to hour according to a known daily pattern. This time-dependent arrival rate is likely to depend on factors such as country, season, day of the week, type of hospital, and so on.

The arrival model for the emergency arrivals as explained in Section 3.1 is a fairly broad class of models which encompasses, for example, the nonhomogeneous Poisson processes. Such a nonhomogeneous Poisson process $X(t)$, $0 \leqslant t \leqslant 1$ with bounded rate function $\lambda(t)$ is a counting process with $X(0) = 0$ and with independent increments $X(t+u) - X(t)$ having a Poisson distribution with mean $\int_t^u \lambda(\tau)\mathrm{d}\tau$. Let X_i be the event times in which $X(t)$ makes a jump, then as discussed in e.g. Kao (1997), the expectation function

$$\Lambda(t) = \mathrm{E}[X(t)] = \int_0^t \lambda(\tau)\mathrm{d}\tau \ , \quad 0 \leqslant t \leqslant 1 \ ,$$

transforms these event times into a (homogeneous) Poisson process with rate 1. That is, the points $U_i = \Lambda(X_i)$ form a regular Poisson process with fixed rate 1. This property allows us to show that nonhomogeneous Poisson processes can be characterised by the emergency arrival model explained above. It is known (Pyke, 1965) that to generate the event times U_i' of a Poisson process $U(t)$ with rate 1 confined to $t \in [0, u]$, one can first generate the total number of events J as a sample of the Poisson distribution with mean u and then generate $U_i', i = 1, ..., J$, as independent samples from the uniform distribution between 0 and u. From this, the transform property gives the (still unordered) event times $X'_i = \Lambda^{-1}(U'_i)$ of a nonhomogeneous Poisson process on the unit interval with expectation function $\Lambda(t)$ so that $\Lambda(1) = u$. Now, since the U'_i are independent, so are the X'_i. Therefore, in the emergency arrival model above, if we choose J to be Poisson with mean $\Lambda(1)$ and $F(t) = \Lambda(t)/\Lambda(1)$, the described emergency arrival model will be equivalent to a nonhomogeneous Poisson process with expectation function $\Lambda(t)$.

3.3. Objective function

The collective waiting times $W_j, j \in \mathcal{J}$, of the emergency patients will feature as a component of the objective function, under a cost function $h(W)$ and cost factor c_W:

$$g_{\mathrm{Wait}}(\pi) = c_W \mathrm{E}\left[\sum_{j\in\mathcal{J}} h(W_j)\right] \tag{1}$$

The other components of the objective function are more common in OR planning literature. To start, we penalise the number of cancellations that occur during the day's execution. Note the difference between annulling a surgery (deciding in advance not to include a surgery into the day's schedule) versus cancelling a surgery (the online decision not to perform surgery due to lack of capacity). We only penalise the latter, not the former. Emergency surgeries cannot be cancelled. An elective surgery may be cancelled if performing it is likely to result in an excessively late completion time, defined as the available capacity plus the "allowed overtime" capacity. Formally, surgery i will be cancelled from instant t, provided that $t + \mu_i > D_k + D_k^{OT}$. Cancellations are determined sequentially: if the ith surgery in the sequence is cancelled, a shorter $(i+1)$th surgery could still start. The decision to cancel is made at its would-be start time, but more frequent evaluations could be easily implemented. The cancellation variable Z_i equals 1 if in the state when all emergencies have arrived $(j = J)$, surgery i has been cancelled; and 0 otherwise. Cancellations receive a penalty in the objective function:

$$g_{\mathrm{Cancel}}(\pi) = c_Z \mathrm{E}\left[\sum_{i\in\mathcal{I}} Z_i\right], \tag{2}$$

Next, we add objectives related to OR utilisation. We assume that all ORs open at the same time in the morning; and that they close once both their emergency surgeries have been completed, and their elective surgeries have been either completed or cancelled. In other words, they are utilised until $C_k = \max\{C_{iJ}, B_j^e + P_j^e : i \in \mathcal{I}_k, Z_i = 0, j \in \mathcal{A}_k\}$ where $\mathcal{A}_k$ is the set of emergencies assigned to OR k, and C_{iJ} denotes the state when all emergencies have arrived $(j = J)$. Given the bottleneck status of the OR, under-utilising one day's OR capacity can create capacity problems in the future. Yet an OR running past closing time incurs costs as well. The variables UT_k and OT_k respectively track unused and over-used capacity in room k, defined as

$$UT_k = \max(D_k - C_k, 0), k \in \mathcal{K}$$
$$OT_k = \max(C_k - D_k, 0), k \in \mathcal{K}$$

Under-utilisation and over-utilisation are penalised in the objective function using the factors c_{UT} and c_{OT}. This leads to the final two objectives:

$$g_{\mathrm{OT,UT}}(\pi) = c_{OT}\mathrm{E}\left[\sum_{k\in\mathcal{K}} OT_k\right] + c_{UT}\mathrm{E}\left[\sum_{k\in\mathcal{K}} UT_k\right], \tag{3}$$

Summarising, the objective function in this paper takes the form:

$$\begin{aligned} &\min_{\pi\in\Pi} g(\pi), \quad \text{with} \\ &g(\pi) = c_W \mathrm{E}\left[\sum_{j\in\mathcal{J}} h(W_j)\right] + c_Z \sum_{i\in\mathcal{I}} \mathrm{E}[Z_i] \\ &\quad + c_{OT}\mathrm{E}\left[\sum_{k\in\mathcal{K}} OT_k\right] + c_{UT}\mathrm{E}\left[\sum_{k\in\mathcal{K}} UT_k\right], \end{aligned} \tag{4}$$

Our contribution in this paper relates mainly to the first term in this objective, as the other terms have featured in the literature already (see Section 2) and are understood to be important. For this reason, we will refer to the latter terms as the "core" objectives, and to the first term as the "emergency waiting time" objective:

$$g(\pi) = c_W \mathrm{E}\left[\sum_{j\in\mathcal{J}} h(W_j)\right] + \mathrm{E}\left[g_{\mathrm{Core}}(\pi)\right], \tag{5}$$

The core objectives can be combined in various ways, but a total cost evaluation as above is relatively common. How to include emergency waiting times is more open to debate. Some previous research has argued that the typical waiting time bounds set by emergency classification systems represent an underlying continuum of risk (Dexter, Macario, Traub, Hopwood, & Lubarsky, 1999). In this sense, low waiting times have an intrinsic medical value, which could be reflected by simply assigning a cost c_W to the term. But while shorter waiting times may lead to slightly more successful surgeries, avoiding the ramifications of *excessive* waiting time may be a more critical point of motivation for hospitals. As such waiting time optimisation may be better reflected by a threshold objective, which incurs a (large) penalty cost c_W when the threshold W_{thr} is exceeded, but is zero otherwise. This is the approach we will take in this paper.

$$\begin{aligned} &g_{\mathrm{Wait}}(\pi) = c_W \mathrm{E}\left[\sum_{j\in\mathcal{J}} h(W_j)\right] \\ &\text{with} \quad h(x) = \mathbb{1}\langle x > W_{thr}\rangle, \end{aligned} \tag{6}$$

3.4. Impact of design choices

It is clear that the objectives above can compete, and depending on the sizes of the respective cost factors, some may dominate others. In particular, other authors have commented on the trade-off between minimising under-utilisation on the one hand, versus the risk of cancellations and over-utilisation on the other (Adan et al., 2011; Rachuba & Werners, 2014). This dynamic applies to our model as well: when cost factors are chosen which punish over-utilisation far more than under-utilisation or vice versa, this results in schedules which are heavily skewed towards one objective over others.

A similar impact, more specific to our research, can be seen when over-emphasising the waiting time cost factors (W_{thr} and associated cost c_W); especially when the emergency arrival distribution is very concentrated. An illustration of this is shown in Figure A3(a,b). For two emergency arrival densities, we found the best schedule for a set of cost factors $W_{thr} = 15$ min and $c_W = 10{,}000$, and display the expected BIMs of this schedule across all scenarios. We can see that in the resultant schedules, a high emphasis has been put on centring expected BIMs around the density peaks.

We also note that the choice of a threshold objective is not the only mechanism to avoid excessive emergency waiting time. Other objectives could serve the same purpose, such as the minimisation of a set number of the largest BII intervals. However, as discussed in Vandenberghe et al. (2019), the impact of this decision is relatively limited: 1% of schedules with the best values for one of these objectives, also score (on average) in the top 3% of schedules for the other objectives.

4. Solution methods

4.1. Objective value estimation

In order to solve the problem $\min_\pi g(\pi)$ of selection and sequencing in each room, several methods are available, but because in general no closed-form expression for $g(\pi)$ is available we will resort to an approximate method. To capture the stochasticity of the problem, we can create a large set $\mathcal{N}$ of independent scenarios (with $|\mathcal{N}| = N$). Each scenario contains independent realisations of the surgery durations $\mathbf{P}$, the number of emergency arrivals J, emergency arrival times $\mathbf{A}$ and emergency surgery durations $\mathbf{P}^e$. These realisations are respectively referred to as $\mathbf{P}^n$, J^n, $\mathbf{A}^n, \mathbf{P}^{en}$. For solution π, let $g(\pi; \mathbf{P}^n, J^n, \mathbf{A}^n, \mathbf{P}^{en})$ be the deterministic value of the cost in the nth scenario, then the true expectation $g(\pi)$ in (4) is replaced by the unbiased estimator

$$\hat{g}_N(\pi) = \frac{1}{N}\sum_{n=1}^{N} g(\pi, \mathbf{P}^n, J^n, \mathbf{A}^n, \mathbf{P}^{en}) \qquad (7)$$

called the sample average function. The minimisation of $\hat{g}_N(\pi)$ serves as an approximation to the "true" problem (4). This method of tackling stochastic problems is referred to as sample average approximation (SAA) and its convergence to the true optimum was studied in great detail by Kleywegt, Shapiro, and Homem-de Mello (2002).

To be clear, the sample average evaluation of a schedule requires the following steps:

Algorithm 1. Sample average function for objective $g(\pi)$ (Full Estimator)

```
for each scenario $n \in \mathcal{N}$ do
    Generate surgery durations $\mathbf{P}^n$ for each of the $M$ elective surgeries
    Generate $J^n$ emergencies with ordered arrival times $\mathbf{A}^n$ and durations $\mathbf{P}^{en}$
    for each emergency arrival $j = 1, \dots, J^n$ do
        for each selected elective surgery $i \in \mathcal{I}$ do
            if surgery $i$ is not yet cancelled then
                Inspect the start time $t = C^n_{ij} - P^n_i$ of this surgery in the scenario
                If the cancellation rule is violated at start time $t$, cancel surgery $i$
    Identify the first available BIM ($B^e_j$) for $j$ among non-cancelled surgeries
    Enter emergency $j$ into the schedule at $B^e_j$, in respective room $k$
    Decrease room capacity $D_k$ by $P^{en}_j$, and update future BIMs and $C^n_{ij}$
  Calculate the four cost components to obtain $g(\pi, \mathbf{P}^n, J^n, \mathbf{A}^n, \mathbf{P}^{en})$
Calculate the average cost across scenarios $\hat{g}_N(\pi)$ as in (7)
```

We emphasise that the break-in of emergencies does not use “future” knowledge in its decisions: the interpretation of being assigned the next B_j^e is not that it knows when the next BIM will occur, but that it *waits* for the BIM to occur. Furthermore, emergencies must be handled sequentially, as each changes the state of the schedule for the next arrival. In computational terms, significant speed gains can be achieved by performing the evaluation in parallel across the N scenarios, and by limiting re-calculations (e.g. if no emergency has entered a room, cancellations within that room remain unchanged). Nonetheless, the evaluation is expensive for large sets of $\mathcal{J}$. In order to make a compromise between calculation time and accuracy, we will propose an approximation method in Section 5.6.

4.2. Genetic algorithm

Genetic algorithms (Goldberg, 1989) are local search methods that allow for a high degree of customisation. Solutions are represented as chromosomes, and follow a process mimicking genetic propagation. Promising chromosomes are selected to combine with others, and mutation operators help explore more of the search space. Algorithm 2 presents pseudo-code for the implementation of the genetic algorithm. The parameter values used throughout this paper are listed in Table A1, and were chosen based on a preliminary analysis on the benchmark set. Note that we limit the number of generations to 500 as our experiments require iterations over large numbers of instances and parameters; practitioners could comfortably pick a higher value. Performance results of these parameters on a mid-size ($K = 10$) test instance are displayed in Figure A4. The rest of this section consists of a detailed explanation of each step in the algorithm.

Algorithm 2. Genetic Algorithm (GA)

```
Generate population $\Pi_0$ of $\phi$ chromosomes through random partial permutation
Evaluate the fitness function value of all solutions in $\Pi_0$.
for $\ell = 1, \ldots, \ell_{\max}$ do
    for $\theta = 1, \ldots, \Theta$ do
        Choose 2 parent chromosomes from $\Pi_{\ell-1}$ for recombination, using rank-based selection.
        Apply appropriate crossover operator to obtain 2 offspring.
        Apply the mutation operator on both offspring.
        Evaluate fitness of both offspring.
        if an offspring has a similarity score of $> 90\%$ with either parent then
        Compare fitness of this offspring with the most similar parent
        if parent fitness is superior then
            Discard the offspring
    From the current population $\Pi_{\ell-1}$ plus non-discarded offspring, create the next population $\Pi_\ell$ by
    selecting $\nu$ chromosomes through elitist selection and $\phi - \nu$ through rank-based selection.
From the final population $\Pi_{\ell_{\max}}$, choose final solution $\hat{\pi}$ so that $\hat{g}_N(\hat{\pi}) < \hat{g}_N(\pi)$ ,$\forall \pi \in \Pi_{\ell_{\max}}, \pi \neq \hat{\pi}$
```

4.2.1. Representation of the solution

Our model uses an integer representation of candidate solutions. This allows surgery IDs to simply be transcribed into the representation. To retain information about the total amount of patients assigned to each room, we use “negative IDs” to represent annulled patients; these annulled surgery IDs are sorted to the end of a sequence to avoid symmetry. For a room k which has surgery IDs $\{1, 2, 3\}$ available to schedule, the planned sequence (1, 3) is thus represented as (1,3,−2) in the chromosome. For ease of interpretation, we represent the surgery schedule as a matrix. The various ORs are defined as rows, whereas columns represent places in the sequence. If not all rows are of the same length, we pad the schedule with zeroes in between selected and annulled patients. For instance, a possible schedule for the room assignments $\overline{\mathcal{I}_1} = \{1, 2, 3\}$;$\overline{\mathcal{I}_2} = \{5, 6\}$ can be rendered as (1,2,−3);(6,0,−5).

4.2.2. Selecting best-fit individuals for crossover

An exhaustive combination of all chromosomes in the population is typically not efficient. Instead, we first evaluate the respective fitness of each chromosome. As covered in Section 4.1, this can be done by evaluating (7). In all further experiments, we select $N = 2000$ as the number of scenarios to estimate.

We then use rank-based selection to select chromosomes for crossover. After sorting, each chromosome π_i receives a rank $1 \le R_i \le R_{\max}$ based on its fitness $\hat{g}_N(\pi_i)$, and an associated probability $R_i / \sum_{j=1}^{R_{\max}} (R_j)$; where $R_{\max}$ is determined by the number of candidates. As this is a minimisation problem, the lowest objective value would receive the best (i.e. highest) rank $R_{\max}$. A weighted random selection then determines the selected chromosomes. This mechanism ensures a balance between better solutions having a higher chance of propagating their content into the next generation, but maintaining genetic diversity as well. It is furthermore robust to high differences in fitness between chromosomes. Ranks are kept up-to-date as new chromosomes enter.

4.2.3. Crossover operator

We employ the two-dimensional substring crossover and associated repair mechanisms, initially proposed by Tsai, Hong, and Lin (2015). This operator is defined specifically to work with a two-dimensional matrix, making it well suited for our case of surgery sets assigned to ORs. Furthermore, the authors show that it performs well compared to competing operators, such as the widely used partially mapped crossover.

The essence of the operator is that a single crossover point in room k and sequence position m is chosen, as well as a crossover direction (horizontal or vertical). In the case of the horizontal direction, the first offspring is formed by copying the rooms $1, \ldots, k{-}1$ from parent 1, as well as the first part of the sequence in room k (namely the surgeries at indices $1, \ldots, m$); the second part of room k and any remaining rows $k+1, \ldots, K$ are copied from parent 2. The second offspring is created analogously, but swapping the roles of both parents. Finally, the case of vertical crossover works by copying columns rather than rows.

The crossover operator is likely to result in invalid schedules: the room sequences of offspring may contain the same surgery ID twice, or contain a surgery ID in both the positive (planned) form and negative (annulled) form. As an example, for the room assignment $\overline{\mathcal{I}_1} = \{1, 2, 3\}$, the room sequences (1, 3) and (3,2,−2) are both invalid. If an invalid offspring is detected, we apply a repair operator, which iterates over one half of the schedule to sequentially resolve duplicates. The precise steps of this repair operator are slightly different depending on whether it is repairing the 1st or 2nd offspring, and whether this offspring was constructed through the horizontal or vertical crossover direction. We refer the reader to Tsai et al. (2015) for the detailed repair steps.

4.2.4. Mutation operator

After crossover, both offspring are subjected to a mutation operator. We implement the equivalent of a bit-by-bit mutation operator, which acts on every part of a chromosome with probability p_{mut}.

First, the operator can make changes to the selection of surgeries in the given chromosome: scheduled surgeries may be annulled, and annulled surgeries may be added to the schedule. We do not iterate over each of the surgeries to determine this change, as this would lead to a "drift" towards having half the surgeries scheduled and half annulled. Instead, both the number of surgeries to add (N_{add}) and annul (N_{annul}) are determined independently, using two samples from the binomial distribution with success probability p_{mut} and $|\bar{\mathcal{I}}| = M$ number of trials. Afterwards, a number of surgery indices equal to N_{annul} are chosen from the pool of scheduled surgeries; and vice versa for N_{add}. If either N_{add} or N_{annul} exceeds the available number of surgeries to add or annul, we choose the available number instead.

Second, we iterate over all scheduled surgeries for a possible change to their sequencing. With probability p_{mut}, surgery i is moved to a random position in the room sequence (excluding its current position). Other surgeries are moved up accordingly.

4.2.5. Maintaining diversity

In order to limit the number of nearly identical chromosomes in the population, we compute a schedule similarity score for each offspring with each of its parents. If an offspring is judged as too similar to either parent, its fitness is compared to that of the most similar parent. If the offspring's fitness is superior, the offspring remains a candidate for the next population; otherwise, it is discarded.

The similarity score of two schedules is computed by iterating over all available surgeries $\mathcal{I}$. If a surgery is annulled in both schedules, this surgery is marked as a point of similarity. If a surgery is planned in both schedules *and* occupies the same absolute position in the room sequence, it is marked as a point of similarity. A percentage score is then obtained by dividing the number of points of similarity by the total number of surgeries available (M). As an example, the room schedules (1,2,3,−4,−5) and (3,2,4,1,−5) have two points of similarity, for a similarity score of 40%.

4.2.6. Replacing least-fit chromosomes with new offspring

After sorting all chromosomes of both the old and the new generation, we choose the amount ν of chromosomes with the best objective value, and these are directly copied into the new generation. The remaining $\phi - \nu$ are determined through a rank-based selection, as in Section 4.2.2.

5. Analysis

5.1. Surgery and emergency data

Our data regarding elective surgeries is mainly comprised of instances from the full benchmark set created by Leeftink and Hans (2018). This benchmark set was created by collecting 200,000 surgery realisations from five Dutch hospitals, and clustering these into 1018 surgical types. These surgery types are then sampled to create various surgery clusters and theoretical case mixes; we select the *RealLifeSurgeryTypes* mix of instances.

The resulting instances are composed of a set of surgeries, represented as three-parameter lognormal distributions, which we use to draw duration samples from; we truncate these to be between 10 and 400 min. Each instance is further characterised by a number of ORs $K = 5, \ldots, 40$, and by a surgery load $0.8, \ldots, 1.2$ representing total expected workload versus total capacity. As our algorithms are capable of selecting which surgeries must be performed, we construct instances with a high surgery load to simulate a waiting list. To do this we select all instances with surgery load 0.8, and with number of ORs $K = 10$, 20, 30. By then halving the number of ORs in each instance, we obtain a set of 30 instances with a surgery load of 1.6 and with $K = 5$, 10, 15.

As the benchmark set only concerns elective surgeries, we base the durations of emergency surgeries on the results of Huber-Wagner et al. (2009), in which the authors analyse data collected by the Trauma Registry of the German Trauma Society and calculate their mean and interquartile range (IQR). This registry records urgent and life-saving operations, making it a good fit for our focus on emergent patients. Huber-Wagner et al. (2009) list five classes of emergency operations with their own mean and IQR, and their relative proportion. Based on these data, we fit three-parameter lognormal distributions, the precise values of which can be found in Appendix A.

In the OR scheduling literature, emergency arrivals usually follow a Poisson distribution, which is well suited for modelling the total amount of emergencies per day. This is, however, insufficiently granular for our purposes. Further, while a number of papers modelling patient flow in the emergency department have been more explicit about the arrival distribution, it is clear that this distribution can vary significantly according to country, season, day of the week, type of hospital, and so on (McCarthy et al., 2008). Our methodology does not presuppose any particular distribution, and in fact, we would recommend practitioners to use their own historical data on emergency arrivals to create a distribution. To reflect this, we seek to estimate the added value of our model for various types of emergency arrival distributions, and so experiments are run for a variety of emergency arrival models:

- **UniformLow**, **UniformMid**, **UniformHigh**: these models share an underlying uniform distribution for arrival times (spread over the day between $t = 0$ and $t = 8$), but differ in the amount of emergency patients likely to arrive. Within any scenario n, the arrival amount J^n is obtained by drawing a sample from a discrete uniform distribution. The number of arrivals in “UniformLow” follows $U(0,\ 4)$; “UniformMid” follows $U(3,\ 7)$; “UniformHigh” follows $U(6,\ 10)$.
  After drawing J^n samples from the underlying distribution, we order the samples to obtain the increasing order statistics of $\mathbf{A}' = (A'_1, \ldots, A'_J)$, as per Section 3.2.
- **RealisticPoisson**: this model is based on the most granular estimation of emergency arrival times (McCarthy et al., 2008) that we were able to find. Based on this input, we defined a non-homogeneous Poisson process (depicted in Figure A5) which defined an arrival rate for each of the hours from 8 AM until 5 PM. To form each scenario, we sample across this range in increments of 0.5 min and a rate 1/120th the hourly rate. The resulting Poisson events determine the number of arrivals as well as their arrival time, precise to 0.5 min.
- **BimodalPoisson**: using a non-homogeneous Poisson process (also depicted in Figure A5), this model creates a bimodal distribution for arrival times, with the peaks roughly corresponding to periods of high traffic congestion (9 AM and 16 PM). As an emergency distribution with lower entropy, this provides insight into a case where emergency arrivals could be more clearly anticipated. Scenarios are generated according to the same method described for “RealisticPoisson”.

5.2. Cost parameters

We set our cost parameters in accordance with other research papers. Olivares, Terwiesch, and Cassorla (2008) report that healthcare practitioners perceive the costs associated with under-utilisation as 60% higher than over-utilisation costs. We couple this with the data from Argo et al. (2009) that under-utilisation costs amount to about \$600 per hour in American public hospitals; a figure in line with values reported for European hospitals (Lamiri, Xie, Dolgui, & Grimaud, 2008). Adjusting for inflation and currency, we set under-utilisation to €600 per hour, and over-utilisation at €375 per hour. Cancellations incur an additional €250 per cancelled patient, which reflects the cost of patient dissatisfaction and resources spent on preparing the patient and OR. We emphasise that the precise value of these parameters is hospital-dependent, and indeed the literature contains several sets of parameters. As our methods and estimators do not hinge on any particular parameter combination, we expect them to be fairly robust to parameter changes.

To our knowledge, this is the first paper that features a penalty cost for excessive waiting time in the objective function. While recent years have seen several papers argue for the importance of short response times for emergencies (McIsaac et al., 2017; Wilde, 2013), estimates on the impact of excessive delay are scarce. We base ourselves on the recent paper by McIsaac et al. (2017), in which the authors use propensity score-matched analyses to measure the association between surgical delay and death. Their results estimate the total cost difference between delayed and non-delayed emergency surgery to around €1325. This is an average across all five emergency classifications (using a very similar system to ESI), but it stands to reason that for the most urgent categories (which we focus on) this will function as a lower bound. Adjusting for this slightly, we set the excessive waiting time cost to €1500. While this is relatively high, our solution method will try to avoid incurring it in too many scenarios. The associated threshold for what is considered excessive waiting time is set to 45 min. This is the threshold analysed in McIsaac et al. (2017) and is consistent with the latest stipulations for appropriate waiting times.

5.3. Objective function of special cases

We also set a few special cases for the calculation of the objective function. Recall that ORs open at the same time in the morning, and that each room closes at the time C_k when emergency surgeries have been completed, and elective surgeries have been either completed or cancelled. Then $C_{\min} = \min_{k\in\mathcal{K}} C_k$ represents the first time an OR completes all its surgeries, and closes. $C_{\max} = \max_{k\in\mathcal{K}} C_k$ is the moment at which all ORs close, and this represents the end of the day shift. Emergency patients arriving after $C_{\max}$ and/or $C_{\min}$ represent a special case.

First, if emergency patients arrive after $C_{\max}$, this is an arrival outside of all scheduled operating blocks of the daily surgery schedule. We thus assume that these patients will be handled by the evening shift or an equivalent emergency readiness team, but do not see it as part of the planned schedule. These emergencies are discarded and not considered in the objective function.

Second, it is possible that emergency patients arrive when a few (but not all) ORs have finished their daily workload, i.e. between $C_{\min}$ and $C_{\max}$. One could presume that such emergencies can be handled in the vacant rooms, voiding the need for break-ins and for this part of the objective function. In fact, our experiments show that under certain parameters, the optimal schedule would be one that leaves one or more ORs entirely empty, in preparation for emergency arrivals. This encroaches on a common and popular research topic in surgery scheduling: is it better to reserve one or more ORs for emergencies only (dedicated OR policy), or reserve some capacity in each elective OR (flexible policy)? The results of comparative studies on both policies remain inconclusive, with some studies favouring the former (Ferrand, Magazine, & Rao, 2010) others the latter (Wullink et al., 2007), and still others proposing hybrid policies (Ferrand, Magazine, & Rao, 2014). As the pure problem (4) presupposes neither policy, our solution methods may choose to pursue either a dedicated or flexible policy depending on cost parameters, and provide evidence which is superior under what circumstances. While this topic has the potential for interesting findings, it is not within the scope of this paper. Given that the concept of BIMs is very aligned with the flexible OR policy, we will assume that it is in effect. Concretely, we assume that once an OR k finishes its daily workload, the room and its surgical team go offline and cannot be used for emergencies arriving after C_k. This disincentivises the creation of empty rooms, and reflects the fact that in a flexible policy, hospital staff is not idling in anticipation of emergencies.

5.4. Value of the stochastic solution

To gain insight into the value of the stochastic solution, we will seek to solve the problem (4) using different estimators: the Full Estimator, CapacityDet, and CapacityStoch. The latter two include stochastic information about emergency arrivals to various degrees of sophistication, and thus represent models that mimic emergency break-ins in less detail. Since CapacityStoch does not use $\mathbf{A}$, and CapacityDet uses neither $\mathbf{A}$, $\mathbf{J}$ nor $\mathbf{P}^e$, they estimate only the "core" cost components (i.e. $g_{\text{Core}}(\pi) = g_{\text{Cancel}}(\pi) + g_{\text{OT, UT}}(\pi)$). We refer to their estimations as $\hat{g}_{\text{Core, CS}, N}(\pi)$ and $\hat{g}_{\text{Core, CD}, N}(\pi)$ respectively. And as these estimators still use SAA, we use $g_{\text{Core, CS}}(\pi; \mathbf{P}^n, J^n, \mathbf{P}^{en})$ and $g_{\text{Core, CD}}(\pi, \mathbf{P}^n)$ for the deterministic values of the costs in the nth scenario for both estimators, respectively.

The estimators are defined as follows:

Algorithm 3. Sample average function of CapacityDet for $g_{\text{Core}}(\pi)$

```
for each scenario $n \in \mathcal{N}$ do
    Generate surgery durations $\mathbf{P}^n$ for each of the $M$ elective surgeries
    Calculate the rounded expected amount of emergencies $\lfloor \mathrm{E}[J] \rceil$ and their expected durations $\mathrm{E}[P^e]$
    for $j = 1, \ldots, \lfloor \mathrm{E}[J] \rceil$ do
        Choose room $k$ by weighted random selection (using room weights $RW_k$)
        Assign emergency $j$ to room $k$, and decrease available capacity $D_k$ by $\mathrm{E}[P^e]$
    for each selected elective surgery $i \in \mathcal{I}$ do
        if surgery $i$ is not yet cancelled then
            Inspect the start time $t = C^n_{i0} - P^n_i$ of this surgery in the scenario
            If the cancellation rule is violated at start time $t$ (with respect to updated $D_k$), cancel surgery $i$
    Calculate the three core components to obtain $g_{\text{Core, CD}}(\pi, \mathbf{P}^n)$
Calculate average cost over scenarios $\hat{g}_{\text{Core, CD}, N}(\pi) = \frac{1}{N}\sum_{n=1}^{N} g_{\text{Core, CD}}(\pi, \mathbf{P}^n)$
```

Algorithm 4. Sample average function of CapacityStoch for $g_{\text{Core}}(\pi)$

```
for each scenario $n \in \mathcal{N}$ do
    Generate surgery durations $\mathbf{P}^n$ for each of the $M$ elective surgeries
    Generate $J^n$ emergencies with durations $\mathbf{P}^{en}$
    for each emergency arrival $j = 1, \ldots, J^n$ do
        Choose room $k$ by weighted random selection (using room weights $RW_k$)
        Assign emergency $j$ to room $k$, and decrease available capacity $D_k$ by $P^{en}_j$
    for each selected elective surgery $i \in \mathcal{I}$ do
        if surgery $i$ is not yet cancelled then
            Inspect the start time $t = C^n_{i0} - P^n_i$ of this surgery in the scenario
            If the cancellation rule is violated at start time $t$ (with respect to updated $D_k$), cancel surgery $i$
    Calculate the three core components to obtain $g_{\text{Core, CS}}(\pi, \mathbf{P}^n, J^n, \mathbf{P}^{en})$
Calculate average cost over scenarios $\hat{g}_{\text{Core, CS}, N}(\pi) = \frac{1}{N}\sum_{n=1}^{N} g_{\text{Core, CS}}(\pi, \mathbf{P}^n, J^n, \mathbf{P}^{en})$
```

CapacityDet (detailed in Algorithm 3) optimises $\mathrm{E}[g_{\text{Core}}(\pi)]$, but only utilises knowledge about the expected number of emergencies that can arrive, as well as their expected surgery duration. The estimator randomly reserves $\lfloor \mathrm{E}[J] \rceil$ blocks of length $\mathrm{E}[P^e]$ across the ORs. This will lead to a higher workload in some ORs, and lead to commensurate changes in over-utilisation, under-utilisation and cancellation risk.

CapacityStoch (detailed in Algorithm 4) improves on the first estimator by actually drawing samples from the distributions of both the number and the duration of the emergencies. However, an emergency is still randomly assigned to an OR, instead of assigning it to the OR with the earliest completion time after the emergency's arrival. Therefore, no samples of the emergency arrival times are used. This estimator is similar to the methods used in some of the latest papers on surgery scheduling (Adan et al., 2011; Molina-Pariente et al., 2016; Moosavi & Ebrahimnejad, 2018).

Full Estimator is our own estimator (detailed in Algorithm 1) for the full objective (4), using all available stochastic information (surgery durations, emergency arrival times, and emergency surgery durations), as well as break-in mechanics.

Each estimator will make selection and sequencing decisions based on its own estimate of the objective function, which is in turn based on the stochastic information which it can process. Assuming that the dynamics regarding break-ins we described in Section 3 represent the most *realistic* environment, we can compare the performance of each estimator by taking their respective optimal schedules and inspecting performance in this most realistic environment. Note that in CapacityDet and CapacityStoch, we assign emergencies to rooms randomly (according to the uniform room weights $RW_k = 1/K \ , k \in \mathcal{K}$), rather than in ways that might help minimise their estimate of g_{Core} (e.g. by choosing the room with the lowest assigned workload). This is because the environment that the estimators will be tested against, will not allow such choices either. As CapacityDet and CapacityStoch do not track arrival times or available BIMs, choosing a random room k is a safe prediction; other choices would in fact disadvantage these estimators further.

Table 1. Average performance results reached for the benchmark set.

	Emergency arrival distributions									
ORs	Realistic Poiss.		Bimodal Poiss.		Uniform Low		Uniform Mid		Uniform High	
5	3598	$(287\vert.)$	3980	$(364\vert.)$	3197	$(187\vert.)$	4598	$(530\vert.)$	–	
	+188	$(-35\vert+223)$	+328	$(+37\vert+291)$	+163	$(+137\vert+26)$	+563	$(+463\vert+100)$	–	
	+398	$(+183\vert+215)$	+551	$(+275\vert+276)$	+306	$(+204\vert+102)$	+491	$(+381\vert+110)$	–	
10	6884	$(53\vert.)$	6645	$(47\vert.)$	6678	$(38\vert.)$	7037	$(71\vert.)$	7703	$(178\vert.)$
	+241	$(+55\vert+186)$	+337	$(-1\vert+338)$	+141	$(+2\vert+139)$	+223	$(+61\vert+162)$	+340	$(+84\vert+256)$
	+892	$(+20\vert+872)$	+808	$(+16\vert+792)$	+534	$(-3\vert+537)$	+878	$(+20\vert+858)$	+901	$(+136\vert+765)$
15	10,426	$(10\vert.)$	9719	$(3\vert.)$	10,002	$(5\vert.)$	10,373	$(15\vert.)$	10,805	$(30\vert.)$
	+132	$(+10\vert+122)$	+273	$(+25\vert+248)$	+165	$(+1\vert+164)$	+292	$(-3\vert+295)$	+316	$(+23\vert+293)$
	+737	$(+3\vert+734)$	+796	$(-2\vert+798)$	+711	$(+3\vert+708)$	+898	$(-3\vert+901)$	+1221	$(+5\vert+1216)$

Experiments where the expected number of emergencies exceeds the number of ORs, are omitted. Each triplet of cells lists the average objective value of the best schedules obtained by (from top to bottom) the Full Estimator, CapacityStoch, and CapacityDet. For the Full Estimator, we provide the absolute cost, and in brackets, its composition $(x|.)$; where x is the value of $\hat{g}_{\text{Wait}}(\pi)$, and the value of $\hat{g}_{\text{Core}}(\pi)$ can be inferred. For the other estimators we list respective deviations from the absolute cost, and between brackets $(x|y)$ the composition; where $x =$ changes in $\hat{g}_{\text{Wait}}(\pi)$, and $y =$ changes in $\hat{g}_{\text{Core}}(\pi)$.

In Table 1, we compare the three estimators. Using the genetic algorithm defined in Section 4.2, each instance of the benchmark set is solved for each of the three estimators, and each of the emergency arrival models. The resulting best schedules are then evaluated as $\hat{g}'_N(\pi)$ where π is the best schedule $\hat{\pi}_{\text{CD},N}, \hat{\pi}_{\text{CS},N}$ or $\hat{\pi}_N$ for CapacityDet, CapacityStoch and Full Estimator respectively. The final evaluation uses a larger number of scenarios $N' = 25000$ to obtain a more precise estimate. Note that as the genetic algorithm does not guarantee optimality, the comparison of the various estimators is inexact. However, as we consistently use the same genetic algorithm to search the same solution space, only with different objective estimators, we expect the relative performance differences between estimators to still be informative.

The results show that the cost composition of $\hat{g}(\hat{\pi})$ is clearly dependent on the parameters of the specific benchmark set, particularly on the ratio between ORs and the average number of emergency patient arrivals. When the number of ORs is low and the number of emergent patients comparatively high, waiting cost violations $\hat{g}_{\text{Wait}}(\hat{\pi})$ account for a significant percentage of the cost. However, as the ratio of ORs to emergent patients increases, the schedule contains more surgeries (and more BIMs). This increases the likelihood that at least one OR will complete its surgery before excessive waiting time occurs, and in turn reduces the importance of $\hat{g}_{\text{Wait}}(\hat{\pi})$.

In comparing the various estimators, we emphasise that even the first estimator ("CapacityDet") is already fairly sophisticated, as it uses the genetic algorithm to find a solution that (across $N = 2000$ scenarios) anticipates total emergency capacity and minimises cancellations, over- and underutilisation. Nevertheless, the schedules it finds tend to have a significantly higher cost than those found by CapacityStoch, which is in turn consistently outperformed by the Full Estimator.

The superiority of the full estimator can partly be explained by the tendency of simpler estimators to incur more frequent threshold violations for emergency waiting time. This is mainly true for the instances with $K = 5$, and is as expected, since the simple estimators could not anticipate this cost component. However, for most instances, the better prediction of emergency break-in dynamics leads to a gain for the core cost components as well, and particularly for the $K = 10$ and $K = 15$ instances, this accounts for the vast majority of improvement. Interestingly, the absolute added value of the full estimator is relatively consistent over the various instances, usually achieving an objective value between h Stoch, which iser than CapacityStoch. In relative terms, this means cost improvements are most significant for the $K = 5$ instances.

When comparing the various emergency arrival distributions, the general pattern is that the added value of the full estimator improves with the average number of emergencies. This is particularly visible for the three uniform distributions: CapacityStoch is quite competitive with the full estimator when applied to UniformLow ($\text{E}[J] = 2$), but gets progressively worse when applied to UniformMid and UniformHigh. Of particular interest is the full estimator's performance when applied to the BimodalPoisson distribution. Though its average number of emergencies is quite low ($\text{E}[J] = 2.9$), the cost improvements achieved by the full estimator are some of the highest across all instances. As BimodalPoisson is a low-entropy distribution (and the most dissimilar from the uniform distribution), its emergency arrivals are the most "predictable", and the easiest for the full estimator to guard against.

5.5. Threshold sensitivity and cost sensitivity

As shown in the previous section, using stochastic information about emergency arrivals can significantly reduce the cost associated with excessive waiting time. As the cost factors related to excessive waiting time (c_W and W_{thr}) become more strict, however, we might expect a more pronounced trade-off: reducing the occurrence of excessive waiting time becomes more critical, even at the expense of the core objectives.

We present an experiment to map this trade-off, and its sensitivity to the strictness of the waiting time cost factors. Using the genetic algorithm, we solve all instances of the benchmark set for the dual parameter ranges $c_W = 750, 1500, 2250$ and $W_{\text{thr}} = 15, 30, 45, 60, 75, 90$. In Figure A6, we record the average waiting time endured per emergency, as well as the value of core costs $\hat{g}_{\text{Core}}(\hat{\pi})$; both averaged over all $K=5$ and $K=10$ instances in the benchmark set. We do not display the value of $\hat{g}_{\text{Wait}}(\hat{\pi})$, as it of course fluctuates depending on the value of threshold and cost multipliers.

Though our algorithms only penalise emergency waiting time if it exceeds the threshold, this affects average waiting time per patient as well. If either the waiting time threshold or cost factor becomes especially strict, optimal schedules begin to put more emphasis on providing rapid break-in, though there appears to be a lower bound of what can be achieved. This also shows how practitioners can use particular cost combinations to accentuate hospital priorities. Accommodating stricter waiting time costs does require the expected trade-off versus core objectives: the best-found solutions for larger c_W and smaller W_{thr} have higher core costs, presumably to find a solution that scores better on $\hat{g}_{\text{Wait}}$. It's worth noting, however, that this trade-off is relatively limited: for $c_W = 1500$, moving the waiting time threshold from $W_{\text{thr}} = 90$ to $W_{\text{thr}} = 45$ leads to significantly lower waiting times while incurring around 5% more core costs.

5.6. Approximation of the objective function

Our experiments have shown that a detailed evaluation of emergency arrivals and the disruption they cause, allows to find solutions with lower total costs. But as may be evident from evaluation algorithm 1, this evaluation is computationally intensive: each emergency arrival can change the schedule state for the next arrival (by causing cancellations and changing the remaining BIMs), and so each emergency must be dealt with sequentially. Further, finding the first available BIM for an emergency arrival, always requires a small scenario-specific minimisation. This forms a dual bottleneck on the evaluation function, making approximate methods tempting.

One approximation method is to simply revert to a less sophisticated estimator (such as CapacityStoch), which does not model emergencies in detail. CapacityStoch assumes that each emergency has an equal chance to enter each of the K rooms; an assumption that comes with a cost, as per Section 5.4. However, we can expect the accuracy of this approximation to be dependent on how well this equal break-in chance reflects reality! Figure A7 supports this intuition by showing the correlation between the percentage error of CapacityStoch, and the value of the expected maximum BII of each schedule. Clearly, CapacityStoch gets progressively worse when schedules have larger maximum BIIs. This is because having a large expected maximum BII is an indicator for how (non-)uniformly BIMs are distributed across the available interval, and thus for how (non-)uniform the chance is that an emergent patient enters any particular room.

We can build an approximation heuristic from this general insight: the chance that an emergency enters a particular room k, is determined by (i) the length of the intervals where room k serves as the first available BIM, and (ii) the likelihood that an emergency will arrive during these intervals. Thus, the choice of which room to enter can be modelled by a weighted random number selection, determined by the schedule π and arrival time density $f(t)$. We can estimate these weights directly by inspecting a number of scenarios N and recording which rooms have the largest BIIs, adjusted for the arrival time density. Note that these estimates are most accurate at modelling the room that the first emergency will enter, and will deteriorate for subsequent arrivals. Still, since j emergency arrivals disrupt at most j rooms, and the size of the disruption $\mathbf{P}^e$ is identically distributed across all rooms, we expect the deterioration to be gradual.

The above method allows us to estimate the disruption caused by emergencies on overall capacity, i.e. the core objectives $g_{\text{Core}}(\pi)$. However, this leaves out waiting cost objective $g_{\text{Wait}}(\pi)$. Fortunately, the above inspection of the largest BIIs per scenario, can also identify the number of BIIs larger than W_{thr}. The length by which these intervals *exceed* W_{thr}, adjusted for arrival time density $f(t)$, provides an estimator for the chance that any emergency arrival will face excessive waiting time.

Detailed steps for the Weighted Room Break-in Heuristic (WRBH) are provided in Algorithm 5. For this, we supplement the notation in Section 3.1 with the following function. Recall that $\mathcal{I}$ contains all scheduled surgeries before any cancellations occur due to emergency arrivals, i.e. before disruption. Counting on the fact that simultaneity of completion times will almost surely not happen, we define the function ρ by

$$\rho(i) = k \Longleftrightarrow \exists i' \in \mathcal{I}_k : C_{i'0} = B_{i0} \ \ , i \in \{1, \ldots, |\mathcal{I}|\} \ .$$

That is, $\rho(i)$ is the room in which the ith BIM occurs, which is a stochastic function that depends on the surgery durations $\mathbf{P}$. In a specific scenario n with durations $\mathbf{P}^n$, it is fixed and we denote it as ρ^n. In Algorithm 5, only the Γ largest BIIs are considered. First, we compute improved room weights RW_k which are then substituted into the CapacityStoch estimator for $g_{\text{Core}}(\pi)$ of Algorithm 4. Then the waiting time cost $g_{\text{Wait}}(\pi)$ is estimated by assuming that the probability of an emergency having to wait longer than W_{thr} can be approximated by the probability that an emergency occurs in the critical zone of the undisrupted schedule. The critical zone is defined as all time points in the Γ largest BIIs that are further than W_{thr} removed from the next BIM. The discretisation of the emergency arrival time distribution $F(t)$ into slots of length δ in lines 11 and 12 is a feature that allows evaluation by look-up table, which can speed up the execution when $F(t)$ is hard to evaluate directly.

Algorithm 5. Weighted Room Break-in Heuristic (WRBH) estimator for $g_{\text{Core}}(\pi)$ and $g_{\text{Wait}}(\pi)$

```
1: Choose parameter ak-in Heuristic (WRBH) estimator for n by
2: Choose parameter $\delta$ (length of slots for discretisation)
3: for scenario $n \in \mathcal{N}$ do
4:     Evaluate scenario $n$ before emergency disruption: compute $C^n_{i0}, B^n_{i0}$ and $S^n_{i0}$
5:     Set $\mathcal{V}^n = \emptyset$
6:     for $\gamma = 1, \ldots, \Gamma$ do
7:         $I^n_\gamma = \arg\ \max_{i \in \mathcal{I} \setminus \mathcal{V}^n} S^n_{i0}$
8:         $\mathcal{V}^n \leftarrow \mathcal{V}^n \cup \{I^n_\gamma\}$
9: Obtain improved room weights
       $RW_k \ = \sum_{n \in \mathcal{N}} \sum_{\gamma=1}^{\Gamma} \mathbb{1}\langle \rho^n(I^n_\gamma) = k \rangle (F(\lfloor B^n_{I^n_\gamma,0}/\delta \rceil \delta) - F(\lfloor B^n_{I^n_\gamma - 1,0}/\delta \rceil \delta))$
       $RW_k \ \leftarrow RW_k / \sum_{q \in \mathcal{K}} RW_q$
10: Obtain single patient waiting time
       $W_{\text{Single}} = \frac{1}{N} \sum_{n \in \mathcal{N}} \sum_{\gamma=1}^{\Gamma} \mathbb{1}\langle S^n_{I^n_\gamma,0} > W_{\text{thr}} \rangle (F(\lfloor (B^n_{I^n_\gamma,0} - W_{\text{thr}})/\delta \rceil \delta) - F(\lfloor B^n_{I^n_\gamma - 1,0}/\delta \rceil \delta))$
11: Estimate $g_{\text{Core}}(\pi)$ as $\hat{g}_{\text{Core, WRBH}, N}(\pi)$, obtained by executing Algorithm 4 with room weights $RW_k$ instead
    of $1/K$ (possibly using the same scenario set)
12: Estimate $g_{\text{Wait}}(\pi)$ as $\hat{g}_{1, \text{WRBH}, N}(\pi) \leftarrow c_W \text{E}[J] W_{\text{Single}}$
```

We test the WRBH heuristic with the following experiments, described in Table 2. First, the full estimator provides the "true" estimates for $g_{\text{Core}}(\pi)$ and $g_{(\pi)}$, averaged across all instances in the benchmark set, and split up for the five emergency distributions defined earlier. We then record the accuracy (percentage error) with which the approximations WRBH and CapacityStoch are able to estimate these objectives. Note that CapacityStoch is not able to estimate the waiting time cost, so we only list $g_{\text{Core}}(\pi)$. We also list average computation time.

The results show that the estimates of $g_{\text{Core}}(\pi)$ made by CapacityStoch have an average error of around 4–17%. When also accounting for the excessive waiting time cost, this error would be larger still.

Clearly, using WRBH significantly improves accuracy: for three of the emergency distributions, WRBH is able to estimate the objective value with a percentage error of $< 1.5\%$; and for the other emergency distribution with $< 3.5\%$. Furthermore, while computation times do increase compared to CapacityStoch, they remain low overall and are virtually unaffected by the average number of emergencies. This in contrast to the computation times for the full estimator, which linearly increase with the number of emergencies, and are around 3.5× slower overall.

Table 2. Average performance results of the two approximations WRBH and CapacityStoch.

		Emergency distributions				
Estimators		Realistic-Poiss. E[J] = 2.9	Bimodal-Poiss. E[J] = 2.9	Uniform-Low E[J] = 1.5	Uniform-Mid E[J] = 4.5	Uniform-High E[J] = 7
Full Estim.	Obj. $g_{\text{Core}}(\pi)$	21,323	21,365	21,715	20,624	19,630
	Obj. $g(\pi)$	21,711	21,745	21,940	21,305	21,023
	Time (sec)	26.77	27.38	15.07	23.10	31.33
Cap. Stoch	Err. $g_{\text{Core}}(\pi)$ (%)	5.7	6.2	4.2	10.4	17.3
	Err. $g(\pi)$ (%)	–	–	–	–	–
	Time (sec)	4.38	4.33	3.76	3.94	4.24
WRBH	Err. $g_{\text{Core}}(\pi)$ (%)	0.3	0.9	0.5	1.6	1.7
	Err. $g(\pi)$ (%)	0.7	1.4	0.8	2.4	3.4
	Time (sec)	6.89	6.99	6.42	6.57	6.91

The full estimator provides the benchmark values for both $g_{\text{Core}}(\pi)$ and $g(\pi)$, averaged across 5000 randomly chosen schedules for each of the 30 analysed instances. We then record the total percentage error of WRBH for both objectives, and for CapacityStoch only for $g_{\text{Core}}(\pi)$. Solution times are the total time to evaluate 5000 schedules using $N = 2000$, averaged across the 30 analysed instances.

6. Conclusion

We have proposed a new model for OR scheduling at the offline operational level, which incorporates the arrival of emergent patients and their break-in to a more granular level. The model includes common objectives such as the minimisation of over- and under-utilisation, and introduces the minimisation of excessive waiting time for emergent patients as an additional objective. A genetic algorithm was developed to solve the resulting model by determining the set of selection and sequencing decisions required.

The added value of this approach is two-fold. First, avoiding excessive waiting time for emergent patients contributes significantly to their health outcomes, and second, including break-in decisions into the model allows a more accurate estimate of the total workload and disruption of the various ORs. Computational experiments quantify the objective gains under a variety of emergency distributions, and we map the sensitivity of these results to more and less strict time thresholds.

On a more practical note, we develop the WBRH heuristic to estimate the objective value faster and with high accuracy. This allows practitioners and other researchers to estimate the impact emergency arrivals will have on their schedule, without replicating all model dynamics related to break-ins.

Given the complexity of the problem, we limited the number of decision variables, and e.g. chose not to include ad-hoc deferrals of cancelled surgeries to other ORs, or patient-driven cancellations (no-shows). Further, we focused our analysis mainly on the allocation phase of operational scheduling (selection and sequencing decisions), and did not include advance scheduling (assigning patients to ORs over a longer time-horizon). Creating an integrated framework for both offline operational planning stages –that is similarly granular with respect to emergencies –remains an open challenge.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by Fonds Wetenschappelijk Onderzoek.

ORCID

Mathieu Vandenberghe http://orcid.org/0000-0002-3994-5584
Stijn De Vuyst http://orcid.org/0000-0003-2780-0588
El-Houssaine Aghezzaf http://orcid.org/0000-0003-3849-2218
Herwig Bruneel http://orcid.org/0000-0002-3739-327X

References

Adan, I., Bekkers, J., Dellaert, N., Jeunet, J., & Vissers, J. (2011). Improving operational effectiveness of tactical master plans for emergency and elective patients under stochastic demand and capacitated resources. *European Journal of Operational Research*, *213*(1), 290–308. doi:10.1016/j.ejor.2011.02.025

Argo, J. L., Vick, C. C., Graham, L., Itani, K. M., Bishop, M. J., & Hawn, M. T. (2009). Elective surgical case cancellation in the Veterans Health Administration system: Identifying improvement areas. *The American Journal of Surgery*, *198*(5), 600–606. doi:10.1016/j.amjsurg.2009.07.005

Brailsford, S. C., Harper, P. R., Patel, B., & Pitt, M. (2009). An analysis of the academic literature on simulation and modelling in health care. *Journal of Simulation*, *3*(3), 130–140. doi:10.1057/jos.2009.10

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

Ceschia, S., & Schaerf, A. (2016). Dynamic patient admission scheduling with operating room constraints, flexible horizons, and patient delays. *Journal of Scheduling*, *19*(4), 377–389. doi:10.1007/s10951-014-0407-8

Dexter, F., Macario, A., Traub, R. D., Hopwood, M., & Lubarsky, D. A. (1999). An operating room scheduling strategy to maximize the use of operating room block time: Computer simulation of patient scheduling and survey of patients' preferences for surgical waiting time. *Anesthesia & Analgesia*, *89*(1), 7–20. doi:10.1097/00000539-199907000-00003

Duma, D., & Aringhieri, R. (2015). An online optimization approach for the real time management of operating rooms. *Operations Research for Health Care*, *7*, 40–51. doi:10.1016/j.orhc.2015.08.006

Eitel, D. R., Travers, D. A., Rosenau, A. M., Gilboy, N., & Wuerz, R. C. (2003). The Emergency Severity Index triage algorithm version 2 is reliable and valid. *Academic Emergency Medicine*, *10*(10), 1070–1080. doi:10.1197/S1069-6563(03)00350-6

Erdem, E., Qu, X., & Shi, J. (2012). Rescheduling of elective patients upon the arrival of emergency patients. *Decision Support Systems*, *54*(1), 551–563. doi:10.1016/j.dss.2012.08.002

Eun, J., Kim, S-P., Yih, Y., & Tiwari, V. (2018). Scheduling elective surgery patients considering time-dependent health urgency: Modeling and solution approaches. *Omega*, *86*, 137–153. doi:10.1016/j.omega.2018.07.007

Ferrand, Y., Magazine, M., & Rao, U. (2010). Comparing two operating-room-allocation policies for elective and emergency surgeries. Proceedings of the Winter Simulation Conference (pp. 2364–2374), Baltimore, MD, December 5–8, 2010.

Ferrand, Y. B., Magazine, M. J., & Rao, U. S. (2014). Partially flexible operating rooms for elective and emergency surgeries. *Decision Sciences*, *45*(5), 819–847. doi:10.1111/deci.12096

Fleet, R., & Poitras, J. (2011). Have we killed the golden hour of trauma? *Annals of Emergency Medicine*, *57*(1), 73–74. doi:10.1016/j.annemergmed.2010.08.003

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning* (1st ed.). Boston, United States: Addison-Wesley Longman Publishing.

Hans, E. W., & Vanberkel, P. T. (2012). Operating theatre planning and scheduling. In R. Hall (Ed.), *Handbook of healthcare system scheduling* (1st ed., vol. 168, chapter 5, pp. 105–130). Heidelberg, Germany: Springer, US.

Huber-Wagner, S., Lefering, R., Kay, M. V., Stegmaier, J., Khalil, P. N., Paul, A. O., … Kanz, K.-G. (2009). Duration and predictors of emergency surgical operations – Basis for medical management of mass casualty incidents. *European Journal of Medical Research*, *14*(12), 532–539. doi:10.1186/2047-783X-14-12-532

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. doi:10.1016/j.cie.2017.07.015

Kao, E. P. C. (1997). *Introduction stochastic processes Kao Edward*. Belmont, United States: Duxbury Press.

Kleywegt, A. J., Shapiro, A., & Homem-de Mello, T. (2002). The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, *12*(2), 479–502. doi:10.1137/S1052623499363220

Lamiri, M., Xie, X., Dolgui, A., & Grimaud, F. (2008). A stochastic model for operating room planning with elective and emergency demand for surgery. *European Journal of Operational Research*, *185*(3), 1026–1037. doi:10.1016/j.ejor.2006.02.057

Landa, P., Aringhieri, R., Soriano, P., Tànfani, E., & Testi, A. (2016). A hybrid optimization algorithm for surgeries scheduling. *Operations Research for Health Care*, *8*, 103–114. doi:10.1016/j.orhc.2016.01.001

Latorre-Núñez, G., Lüer-Villagra, A., 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. doi:10.1016/j.cie.2016.05.016

Leeftink, G., & Hans, E. W. (2018). Case mix classification and a benchmark set for surgery scheduling. *Journal of Scheduling*, *21*(1), 17–33. doi:10.1007/s10951-017-0539-8

McCarthy, M. L., Zeger, S. L., Ding, R., Aronsky, D., Hoot, N. R., & Kelen, G. D. (2008). The challenge of predicting demand for emergency department services. *Academic Emergency Medicine*, *15*(4), 337–346. doi:10.1111/j.1553-2712.2008.00083.x

McIsaac, D. I., Abdulla, K., Yang, H., Sundaresan, S., Doering, P., Vaswani, S. G., … Forster, A. J. (2017). Association of delay of urgent or emergency surgery with mortality and use of health care resources: A propensity score-matched observational cohort study. *Canadian Medical Association Journal*, *189*(27), E905–E912. doi:10.1503/cmaj.160576

Meskens, N., Duvivier, D., & Hanset, A. (2013). Multi-objective operating room scheduling considering desiderata of the surgical team. *Decision Support Systems*, *55*(2), 650–659. doi:10.1016/j.dss.2012.10.019

Molina-Pariente, J. M., Fernandez-Viagas, V., & Framinan, J. M. (2015). Integrated operating room planning and scheduling problem with assistant surgeon dependent surgery durations. *Computers & Industrial Engineering*, *82*, 8–20. doi:10.1016/j.cie.2015.01.006

Molina-Pariente, J. M., Hans, E. W., & Framinan, J. M. (2016). A stochastic approach for solving the operating room scheduling problem. *Flexible Services and Manufacturing Journal*, *30*(1–2), 224–251. doi:10.1007/s10696-016-9250-x

Moosavi, A., & Ebrahimnejad, S. (2018). Scheduling of elective patients considering upstream and downstream units and emergency demand using robust optimization. *Computers & Industrial Engineering*, *120*, 216–233. doi:10.1016/j.cie.2018.04.047

Newgard, C. D., Schmicker, R. H., Hedges, J. R., Trickett, J. P., Davis, D. P., Bulger, E. M., … Nichol, G. (2010). Emergency medical services intervals and survival in trauma: Assessment of the "golden hour" in a North American Prospective Cohort. *Annals of Emergency Medicine*, *55*(3), 235–246.e4. doi:10.1016/j.annemergmed.2009.07.024

Olivares, M., Terwiesch, C., & Cassorla, L. (2008). Structural estimation of the newsvendor model: An application to reserving operating room time. *Management Science*, *54*(1), 41–55. doi:10.1287/mnsc.1070.0756

Paul, J. A., & MacDonald, L. (2013). Determination of number of dedicated OR's and supporting pricing mechanisms for emergent surgeries. *Journal of the Operational Research Society*, *64*(6), 912–924. doi:10.1057/jors.2012.92

Pyke, R. (1965). Spacings. *Journal of the Royal Statistical Society: Series B (Methodological)*, *27*(3), 395–449. doi:10.1111/j.2517-6161.1965.tb00602.x

Rachuba, S., & Werners, B. (2014). A robust approach for scheduling in hospitals using multiple objectives. *Journal of the Operational Research Society*, *65*(4), 546–556. doi:10.1057/jors.2013.112

Testi, A., Tanfani, E., & Torre, G. (2007). A three-phase approach for operating theatre schedules. *Health Care Management Science*, *10*(2), 163–172. doi:10.1007/s10729-007-9011-1

Tsai, M.-W., Hong, T.-P., & Lin, W.-T. (2015). A two-dimensional genetic algorithm and its application to aircraft scheduling problem. *Mathematical Problems in Engineering*, *2015*, 1–12. doi:10.1155/2015/906305

van Essen, J. T., Hans, E. W., Hurink, J. L., & Oversberg, A. (2012). Minimizing the waiting time for emergency surgery. *Operations Research for Health Care*, *1*(2–3), 34–44. doi:10.1016/j.orhc.2012.05.002

van Essen, J. T., Hurink, J. L., Hartholt, W., & van den Akker, B. J. (2012). Decision support system for the operating room rescheduling problem. *Health Care Management Science*, *15*(4), 355–372. doi:10.1007/s10729-012-9202-2

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. doi:10.1016/j.orhc.2015.05.005

Vanberkel, P. T., Boucherie, R. J., Hans, E. W., Hurink, J. L., Van Lent, W. A. M., & Van Harten, W. H. (2011). An exact approach for relating recovering surgical patient workload to the master surgical schedule. *Journal of the Operational Research Society*, *62*(10), 1851–1860. doi:10.1057/jors.2010.141

Vandenberghe, M., De Vuyst, S., Aghezzaf, E. H., & Bruneel, H. (2019). Surgery sequencing to minimize the expected maximum waiting time of emergent patients. *European Journal of Operational Research*, *275*(3), 971–982. doi:10.1016/j.ejor.2018.11.073

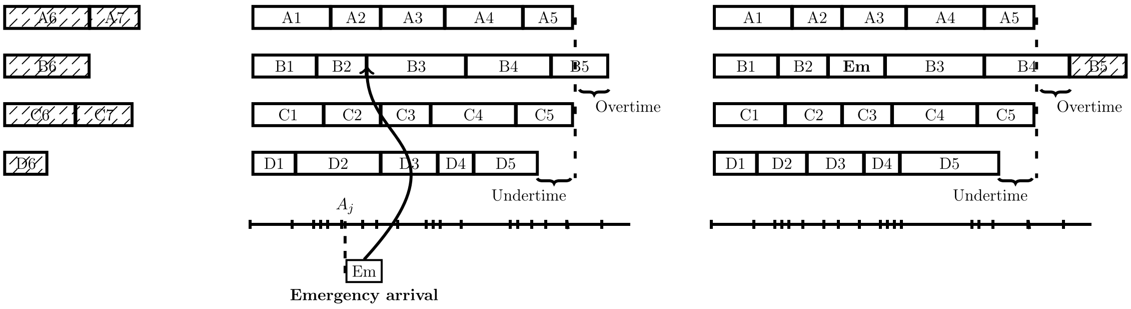
Wilde, E. T. (2013). Do emergency medical system response times matter for health outcomes? *Health Economics*, *22*(7), 790–806. doi:10.1002/hec.2851

Wullink, G., Van Houdenhoven, M., Hans, E. W., Van Oostrum, J. M., Van Der Lans, M., & Kazemier, G. (2007). Closing emergency operating rooms improves efficiency. *Journal of Medical Systems*, *31*(6), 543–546. doi:10.1007/s10916-007-9096-6

Appendix A. Supporting figures and tables

Table A1. Final choice of parameters for the genetic algorithm.

Parameters	Values
Max. generations	$\ell_{\max} = 500$
Population size	$\phi = 200$
Scenarios analysed	$N = 2000$
Mutation prob.	$p_{\text{mut}} = 0.01$
Crossover pairs	$\Theta = 0.075\ \phi$
Fraction of elite selection	$\nu = 0.1\ \phi$



(a) Annulled surgeries (b) Before emergency arrival (c) After emergency arrival

Figure A1. Visual example of the scheduling model, illustrated for a single scenario. (a) shows the surgeries that were annulled on this particular day. (b) displays the planned schedule for the day, and the arrival of an emergency. (c) shows the further execution of the schedule, which culminates in a cancellation. (b) and (c) also display the sets of BIMs for each situation. Note that the model reacts to events as they unfold in any particular scenario. (a) Annulled surgeries. (b) Before emergency arrival. (c) After emergency arrival.

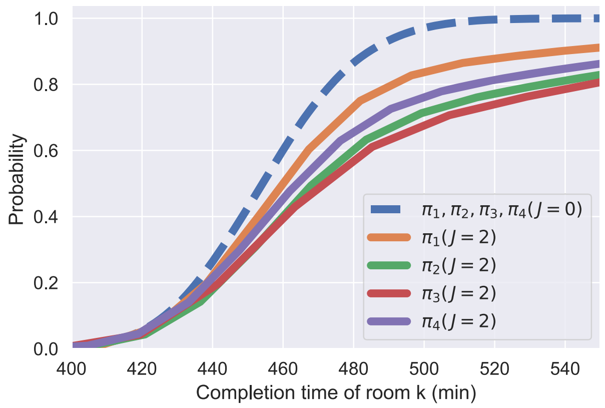


Figure A2. Illustration of OR interdependence. For an instance of five ORs, we compare four schedules $(\pi_1, \pi_2, \pi_3, \pi_4)$ which are identical apart from the sequences in the first room. We display the empirical distribution of this single room's completion time, under the four sequences and under either $J=0$ or $J=2$ emergency arrivals. Without emergencies, all sequences lead to the same completion time distribution (dashed line). If emergencies do arrive, however, different sequences change the likelihood that an emergency will enter the first room and delay completion time.

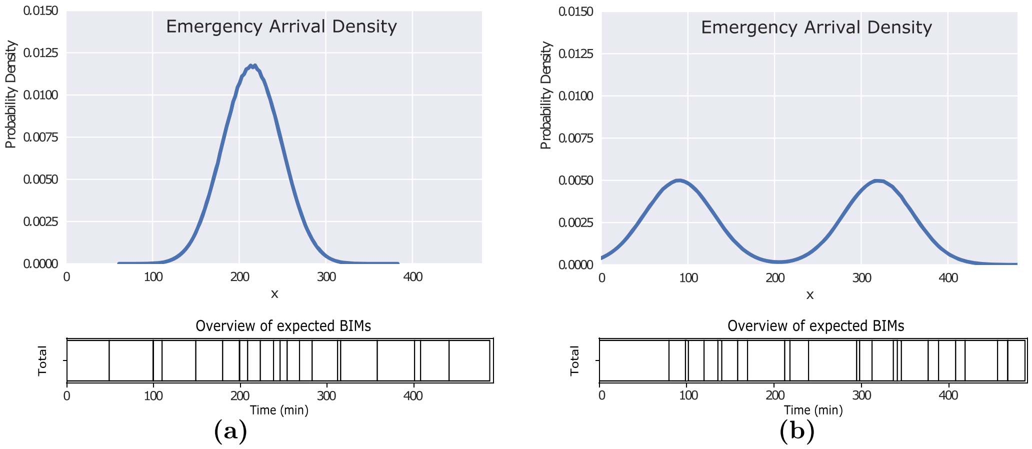


Figure A3. Comparison of the best-found schedules for two different emergency arrival distributions, for an instance of $K=5$ and $M=35$, divided equally across rooms. We use a combination of cost factors which heavily penalises excessive waiting times; thus incentivising schedules which usefully spread the set of BIMs. We illustrate the two resulting sets of BIMs using expected surgery durations.

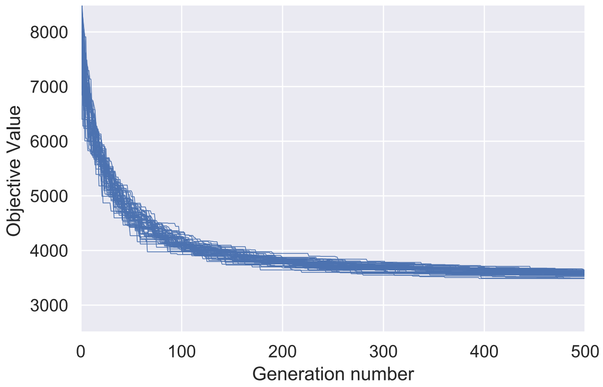


Figure A4. Genetic algorithm performance results, for 50 executions on the same mid-size problem instance. For each execution, we display the best objective value encountered, and plot its evolution across 500 generations.

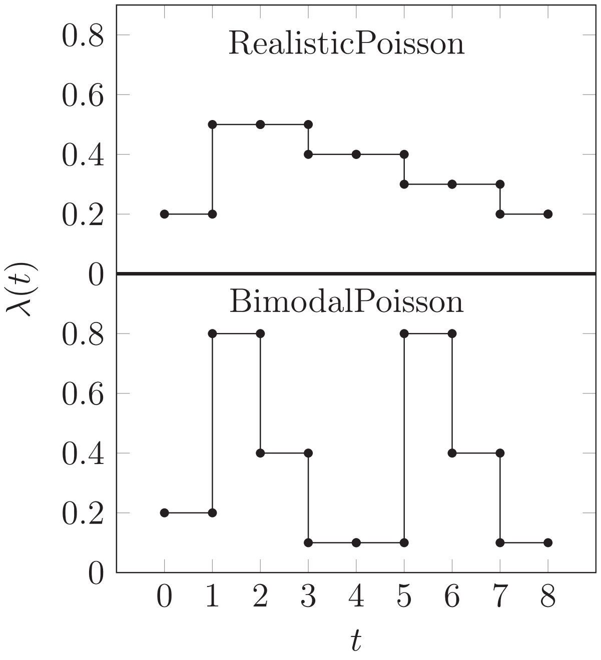


Figure A5. Visualisation of the non-homogeneous Poisson processes used for two of the emergency distributions. $\lambda(t)$ represents the hourly rate of emergency arrivals. The expected amount of emergency arrivals $\mathrm{E}[J] = \int_0^8 \lambda(t)dt$ is 2.9 in both cases.

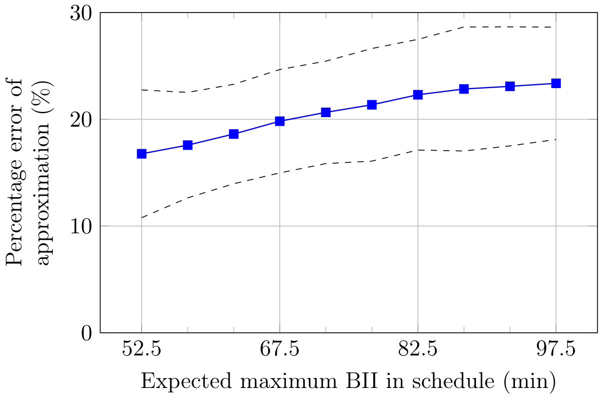


Figure A7. An illustration of the correlation between the expected maximum BII in a schedule, and the percentage error of the CapacityStoch approximation (against the value of the full estimator). For a single benchmark instance, we record both properties for 5000 random schedules (that is, randomly chosen partial permutations of the fixed assignment). Based on the size of the expected maximum BII, we sort the 5000 points into horizontal bins of 5 min. The three lines represent the average percentage error, and 5% and 95% empirical quantiles of the average percentage error.

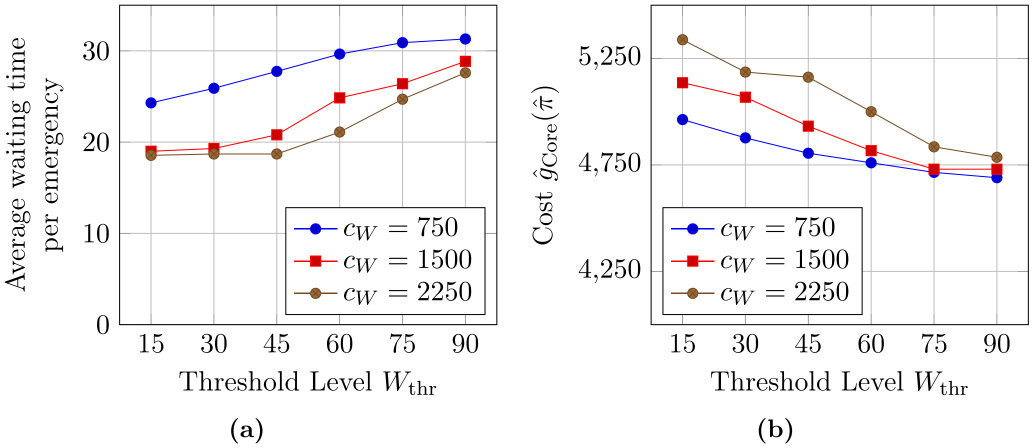


Figure A6. Cost evolution for different values of the waiting time threshold W_{thr} and cost factor cW, for the emergency model "RealisticPoisson", and averaged across all $K=5$ and $K=10$ instances in the benchmark set. Subfigure (a) shows the average waiting time endured per emergency arrival, i.e. $\sum_{j=1}^{J} W_j/\mathrm{E}[J]$, (b) shows the value of the core objectives $\hat{g}_{\text{Core}}(\hat{\pi})$.

Appendix B. Lognormal distributions for emergency surgeries

Three-parameter lognormal distributions are characterised by μ, σ and γ, corresponding with mu, sigma and location (threshold). Their distribution is characterised as:

$$Y = \gamma + e^X \quad \text{with} \quad X \sim N(\mu, \sigma)$$

Since the maximum size of our experiments is $N = 25{,}000$ scenarios with at most $J^n = 10$ emergencies per scenario, we created 250,000 emergency durations by sampling from each surgery category in the proportion provided. The resultant samples were truncated between 10 and 420 min.

The lognormal parameters for emergency surgeries are based on the estimates of Huber-Wager et al. (2009) and displayed in Table A2.

Table A2. Details of the emergency surgery durations used in experiments.

				Lognormal fit		
Region	Proportion (%)	Mean (min)	IQR (min)	μ	σ	γ
Head	26.3	110	55–140	4.46	0.71	0
Thorax	5.0	91	8–146	3.95	1.2	0
Abdomen	54.1	137	70–175	4.6	0.75	10
Pelvis	11.5	136	60–185	4.6	0.85	5
Extremities	3.1	142	80–180	4.8	0.6	0

The first four columns are taken from the analysis of Huber-Wagner et al. (2009), and in the last three columns, we show the parameters of the three-parameter lognormal distributions that gave the best fit to the mean and IQR range provided.