J Heuristics (2011) 17:389–414
DOI 10.1007/s10732-010-9139-x

Local search for the surgery admission planning problem

Atle Riise · Edmund K. Burke

Received: 23 June 2009 / Revised: 30 March 2010 / Accepted: 5 July 2010 /
Published online: 11 September 2010

Abstract We present a model for the surgery admission planning problem, and a meta-heuristic algorithm for solving it. The problem involves assigning operating rooms and dates to a set of elective surgeries, as well as scheduling the surgeries of each day and room. Simultaneously, a schedule is created for each surgeon to avoid double bookings. The presented algorithm uses simple Relocate and Two-Exchange neighbourhoods, governed by an iterated local search framework. The problem's search space associated with these move operators is analysed for three typical fitness surfaces, representing different compromises between patient waiting time, surgeon overtime, and waiting time for children in the morning on the day of surgery. The analysis shows that for the same problem instances, the different objectives give fitness surfaces with quite different characteristics. We present computational results for a set of benchmarks that are based on the admission planning problem in a chosen Norwegian hospital.

Keywords Surgery allocation · Surgery scheduling · Surgery admission planning · Operating theatre planning · Meta-heuristics · Iterated local search · Search space analysis

A. Riise (✉)
Department of Applied Mathematics, SINTEF ICT, P.O. Box 124 Blindern, 0314 Oslo, Norway
e-mail: atle.riise@sintef.no

A. Riise
Centre of Mathematics for Applications, University of Oslo, P.O. Box 1053 Blindern, 0316 Oslo, Norway

E.K. Burke
School of Computer Science, University of Nottingham, Nottingham NG8 1BB, UK

1 Introduction

Health services are a major cost item in national budgets world wide. The sector is continuously challenged to provide the best possible care and treatment with limited resources. In hospitals, a significant cost driver is surgery (Macario et al. 1995; Cardoen et al. 2009). This is partly because of the expensive resources that are directly involved in surgery. Also, the operating room management greatly influences other activities in the hospital, indirectly influencing the use of many associated resources, and hence cost. Surgery planning on different levels and time scales is thus crucial for cost efficient treatment and care, both in the surgery department and in other parts of the hospital. Better surgery planning can improve resource efficiency, level staff work load, reduce patients' waiting time, reduce the number of surgery cancellations, and improve overall performance in the hospital (Cardoen et al. 2009). Surgery planning is, however, a very complex task. This is because of the many resources that are involved, their use in other hospital activities, and the high degree of dynamics and uncertainty that is inevitable in most large modern hospitals. We believe that the development of decision support systems based on modern search technology has the potential to significantly improve the various planning processes that are involved under the headings of surgery planning or operating room management.

1.1 Surgery planning—introduction and context

"Surgery planning" is a rather ambiguous term which covers several different aspects related to the planning and management of a surgery department's operation. The topic includes strategic, tactical and operational aspects. In this paper, we shall focus on operational surgery scheduling. However, let us first introduce the context in which this problem is typically found. On a strategic level, we have the issue of deciding the desired mix of patients, typically maximising profit or targeting a budget based on the existing resources, infrastructure, and demand. The targets may also be set by external authorities, for example in the case of public hospitals. This strategic planning determines how much operating room time should be allocated to different patient groups (Blake and Carter 2002; Beliën 2007). The results of this planning provide input to tactical surgery planning, which often includes the pre-allocation of operating room time to the hospital's various specialities. The way in which different hospital resources are organised varies significantly, but by "specialty" we mean here the organisational unit that "owns" the operating room time. This planning results in a so called "master surgery schedule", which is a (typically cyclic) timetable for the operating rooms, where the operating room opening hours are divided into predefined blocks and allocated to the various specialties (see e.g. Blake et al. 2002; Santibanez et al. 2007; Dexter and Macario 2002). A master surgery schedule is typically created or adjusted a few times per year, as a result of changes in hospital budgets, the availability of staff and equipment, or fluctuations in demand. Note that the existence of a master surgery schedule is not necessary; to use such a pre-allocation of operating room time to specialties or not is a choice made by individual hospitals.

At the operational level we find a set of problems that we can jointly label as operational surgery scheduling. This includes several different stages of planning, each happening at a different point in time before the date of surgery. A common feature is that they deal with the allocation of critical resources such as surgeons, nurses, rooms, equipment, and operation time to known individual surgeries. They also all happen on a daily—or weekly—basis (see, e.g. Weiss 1990; Ozkarahan 1995; Ozkarahan 2000). We will elaborate on the different operational surgery planning tasks below. When such problems are formulated within the constraints given by a master surgery schedule, it is referred to as "Block scheduling" (see, e.g. (Hans et al. 2008; Ozkarahan 2000)).

In addition to the relationships between the above planning tasks at different levels, surgery planning is also strongly related to other planning activities in the hospital. Some connections are found along the care pathway that the patient follows through the various units in the hospital. For example, Santibanez et al. (2007) considers how peak post-surgical resource requirements can be reduced through better master surgery schedules. Other connections, in a sense orthogonal to the care pathway, come from the fact that many of the involved resources also contribute towards other activities in the hospital. The surgery plan, therefore, influences, or is influenced by, other planning processes in the hospital, such as staffing or personnel rostering. For example, a master schedule influences the variation in demand for surgeons and nurses (see for example Gendreau et al. (2006) concerning physician scheduling and Burke et al. (2004) dealing with nurse rostering). The work in Beliën and Demeulemeester (2008) shows how the utilisation of nurses, and thus staffing costs, can be improved by combining nurse rostering and master surgery schedule generation. In turn, at an operational level, the personnel rosters that are based on this demand will constrain the operational surgery planning. These examples illustrate that surgery planning should ideally not be considered in isolation, but that it should be viewed as an important part of the overall operational planning of hospital activities. In practical terms, however, organisational constraints and the shear complexity of the involved optimisation problems, make this an ambition for the future. For a recent survey on surgery planning and scheduling, see Cardoen et al. (2009).

1.2 Operational surgery scheduling

The operational surgery scheduling problem comes in many variations. In its most basic form it may be informally described as the task of allocating, to a set of surgeries, dates and times of surgery, while reserving capacity for these surgeries on a set of constrained physical resources. Such resources may, for example, be operating rooms, operation teams, surgeons, equipment, or post-operative bed capacity. Objectives are typically tardiness costs (overtime), hospitalisation costs, intervention costs, operating room utilisation, patient's waiting time, and patient or personnel preferences, among others.

Several variations and extensions of this problem arise from differences between hospitals and between planning situations at different time scales. We will provide some context through a typical example: Having been recommended for surgery by a specialist, the patient is registered on a waiting list. Based on such a list, the first step of surgery planning is the ‘admission planning’, in which a date of hospitalisation is decided upon and then communicated to the patient. The result is an allocation of an operating room, an operating team, a specialty and a date. Objectives are typically minimising the patient’s waiting time and/or maximising resource utilisation. The admission planning is typically performed at a point in time which can vary between anything from one or two weeks to several months before the hospitalisation date, depending on the urgency of the patient’s condition. Admission planning will be described in more detail in Sect. 2. About a week or two before hospitalisation, a ‘surgery scheduling’ process is performed, allocating specific operating room time, and all relevant resources. Due to the high degree of dynamics in the hospital operations, this surgery scheduling is typically re-done the day before surgery, at which time the plan may be finalised and approved by surgeons and other crucial staff. Finally, during the actual day of surgery, the plan will be dynamically adapted to any unforeseen development that will occur. Note that all these planning stages are in principle working on the same plan, but focusing on different sets of resources, and making slightly different decisions for activities on different time horizons.

Apart from these variations in time scale, various other extensions and specialisations of the basic surgery scheduling problem are treated in the literature. The operating rooms themselves are not necessarily the bottleneck in the care pathways. Sometimes, the bottleneck can be caused by *post-operative resources*. After surgery, patients are transferred to a recovery room, in which they are monitored until they wake up. If no unforeseen development takes place, they are then typically transferred to a regular ward or possibly to an intensive care unit. The recovery room is considered to be associated with a set of operating rooms, and the recovery room beds may be viewed as bottle neck resources in surgery scheduling, representing a causal link between the schedules for the different operating rooms. Several authors consider recovery bed limitations. The work in Jebali et al. (2006) considers the number of available beds in the intensive care unit, as well as recovery room beds, as bottle neck resources. The authors take into consideration the fact that if no recovery bed is available at the end of the intervention procedure, then the recovery procedure will have to start in the operating room, thus delaying future interventions. Considerable work has been done on predicting hospital bed availability, as this is an important factor, e.g. when scheduling interventions in an operating room (Blake and Carter 1997). A given type of intervention may require that certain types of *equipment* be present in the operating room. Some equipment may be moved between operating rooms, and may have to be sterilised between uses. The approach in Jebali et al. (2006) considers stationary equipment that is present in some, but not all, of the operating rooms in the operating theatre.

Hospital operations involve considerable *uncertainty*. This arises from various sources, such as uncertain surgery and recovery durations, cancellations, and unexpected disruptions due to emergency interventions. As pointed out in Przasnyski (1986), a good estimate of surgery durations is essential to successful surgery scheduling. Such durations are by their very nature stochastic, as well as dependent on the patient, the surgery type, the individual surgeon, and whether training for a surgeon is involved. The importance of this uncertainty in durations is acknowledged by most authors, although many disregard it in order to simplify the scheduling problem.

However, more and more authors seem to address this important issue (see for example Denton et al. 2007; Hans et al. 2008; Zhou and Dexter 1998; Marcon et al. 2001; Charnetski 1984). There may also be uncertainty in the capacity, as key resources, human or otherwise, may unexpectedly become reduced or unavailable.

Other extensions include considerations of employee preferences and work regulations, and operating room cleaning time.

As we have seen above, different versions of the problem arise from differences in time scale, and differences between hospitals. In our experience, it is difficult to find two hospitals with exactly the same surgery scheduling problem. Perhaps as a reflection of this, the surgery scheduling literature is lacking a common ontology for operational surgery scheduling problems, or even a common repository of benchmark problems. This, of course, makes it difficult to compare algorithmic approaches. Furthermore, it underpins the following important observation: For a decision support tool to be generally useful with a minimum of customisation to each individual hospital and planning situation, it must be able to solve a wide range of surgery scheduling problems.

1.3 Challenge and contribution

In the literature, the surgery scheduling problem is often considered to be two separate decision steps (Jebali et al. 2006; Fei et al. 2006). The first concerns the assignment of interventions to operating rooms and days (Noyan Ogulata and Erol 2003; Chaabane et al. 2006). We call this "intervention assignment" (it is also known as "advance scheduling" (Ozkarahan 2000)). In Hans et al. (2008), the authors argue that the intervention assignment problem is a generalisation of the so-called "general bin packing problem with unequal bins", and is thus strongly NP-hard.

The second step entails scheduling the interventions that have been assigned within the day and room (Sier et al. 1997; Cardoen et al. 2009). We call this "intervention scheduling" (it is also called "allocation scheduling" (Ozkarahan 2000)). This is usually studied in connection with planning closer to the day of surgery. As argued in Cardoen et al. (2009), this problem can also be NP-hard.

The above decomposition has distinct disadvantages. As pointed out in Cardoen et al. (2009), the quality of the surgery sequence that can be achieved in the intervention scheduling step is highly dependent on the assignment to days and rooms that was made in the intervention assignment step. This is, of course, because the scheduling is bound by the previous assignment decisions, which was typically made without consideration of the scheduling step objectives. This is supported by the work in Jebali et al. (2006), which shows that allowing the modification of previous assignments during intervention scheduling gives improved solutions, although at the price of a considerably higher calculation effort. Hospital admission planning is generally considered to include only intervention assignment. However, in many hospitals the admission planner actually includes scheduling considerations in planning. This is done to take various schedule dependent objectives into account. Examples are objectives that give some "time-of-day" preference for certain patients, such as children or patients with diabetes. Other considerations can be to schedule certain patients after all others to reduce the risk of infections, or to make sure that pre-assigned surgeons can always perform their surgeries without being in more than one room at a time. For a decision tool to be useful for the admission planner, it should, therefore, consider the fully combined admission planning problem, including both intervention assignment and intervention scheduling decisions.

We believe that the above decomposition into two separate steps is often motivated by a need to reduce problem complexity. Indeed, each of the two steps involve very complex problems, which at least in some versions are NP-hard. The combination of the two problems can therefore be expected to be very difficult to solve, especially for real size problem instances. This may be the reason that very few papers consider a combination of assignment and scheduling within the same problem. Those that do, do so in connection with short term planning, for example for one day or one week into the future (Guinet and Chaabane 2003; Roland et al. 2006; Pham and Klinkert 2008; Chaabane et al. 2006). In Roland et al. (2006), the authors combine assignment and scheduling for short term planning. They conclude that a meta-heuristic method, in their case a genetic algorithm, was necessary to solve even a one day problem in a reasonable time. In Pham and Klinkert (2008) the problem is viewed as a multi-mode blocking job shop problem, which the authors solve with CPLEX for one week problems. They suggest that meta-heuristic methods should be developed to solve larger problems.

The challenge is thus to develop search methods for the admission planning problem, including both intervention assignment and intervention scheduling. One of the contributions of this paper is that we investigate a search method for this combined problem, and for realistically sized problem instances. Such a method will of course also be applicable in short term planning where previous assignments may be reconsidered. Note, however, that admission planning problems are typically considerably larger than short term planning problems. The conclusions from the previous work mentioned above therefore convince us that a meta-heuristic approach is appropriate. We implement a local search method based on iterated local search and variable neighbourhood descent, and analyse its properties when applied to the described admission planning problem (see Sect. 2).

In addition, we include the constraint that the allocated surgeons can only be in one operating room at any one time. This means that in addition to creating a schedule for interventions in each operating room for each day, we also need to maintain a schedule for each surgeon (note that similar constraints may apply to other resources, such as anaesthesiologist or mobile equipment). This adds considerable complexity to the problem, which may be why this aspect has, to our knowledge, only been considered for smaller, short term, problems (Fei et al. 2006; Pham and Klinkert 2008). We develop a simple heuristic for handling this issue for realistically sized admission planning problems.

Our test cases are based on the characteristics of a surgery department of a selected Norwegian hospital. They are available for other researchers through our web pages at www.sintef.no/hospital.

In Sect. 2, we give a formal definition of the admission planning problem. Section 3 introduces the local search algorithm under investigation, and Sect. 4 describes the test data that were used. Section 5 contains an analysis of the search spaces for our test problems, and an assessment of the suitability of our algorithm. Some benchmark results are given in Sect. 6, and we conclude and indicate directions for future research in Sect. 7.

2 Problem definition

As explained in Sect. 1.3, the human admission planner often considers both intervention assignment and intervention scheduling aspects. Our problem model must therefore include a schedule of surgeries for each room and day. In manual planning, scheduling related objectives may not be readily visible, as the desired effect is often obtained by applying simple rules of thumb. When we want to state the problem formally, however, we want to include as many of the real, some times obscured, objectives as possible.

Typically, the planning is done from anything between 2 or 3 weeks to several months before the date of surgery, depending on the length of the queue, the capacity of the clinic and the urgency of the patient's diagnosis. A typical planning process can be outlined as follows. Each day, the planner receives new referrals of patients from specialists in or outside the hospitals. These are to be put into the surgery schedule for the coming months. This schedule already exists, as the result of previous planning. The admission planner, therefore, always works with the starting point of an existing plan in which surgeries may be "un-served", "planned but not fixed", or "fixed". Surgeries that are fixed are typically only fixed to dates, and this is because the date has been communicated or agreed on with the patient. Such surgeries can still be moved between rooms, or re-scheduled within the day. Regularly, say daily, the planner will communicate with some patients and fix some surgeries. Other surgeries may be put into the plan, or be left "un-served" until a later time. The admission planning problem therefore consists of:

- a set of surgeries to be performed,
- an initial plan, in which the date of some of these surgeries is fixed,
- a model of the hospital, including all relevant resources, their availability, relevant time constraints, and other requirements.

We will come back to a more formal and detailed description below.

The admission planning problem typically considers fewer resources than other stages of surgery scheduling that happen closer to the date of surgery. The level of detail about resource availability is usually lower, and the uncertainty related to this information is higher than more short-term surgery scheduling stages. For admission planning, critical resources are typically surgeons, operating rooms, surgery teams, or care capacities in some parts of the care pathway. In our version of the problem, operating rooms and surgeons are considered to be the critical resources. Only elective patients are considered, and all durations are treated as deterministic. Let us define the problem more formally.

Let N be the number of surgeries, D the number of days in the planning period, counted from the day after the day of planning, and R the number of operating rooms. We use a discrete time model, where P is the number of time periods in which an intervention may be performed during any day. We define our basic decision variable as

$$x_{ipdr} = \begin{cases} 1, & \text{if intervention } i \text{ starts in period } p \text{ on day } d \text{ in room } r \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

Here, $i \in [1, N]$, $p \in [1, P]$, $d \in [1, D]$, and $r \in [1, R]$. There are several objectives or preferences to consider, usually measurable in different dimensions. In this study, we have included three objectives that seem relevant; Patient waiting time, surgeon overtime, and the waiting time for children in the mornings. By a patient's waiting time, we refer to the number of days from when a patient is first referred for surgery until the planned date of surgery, relative to the number of days between the referral date and a set deadline for treating the patient. This deadline is set based on the medical urgency of the patient's ailment. In mathematical terms, we can present our waiting time objective as

$$O_W = \frac{1}{N} \sum_{i=1}^{N} \sum_{d=1}^{D} \sum_{p=1}^{P} \sum_{r=1}^{R} x_{ipdr} \left(\frac{d - R_i}{G_i - R_i} \right)^2 \tag{2.2}$$

Here, R_i is the referral date of surgery i, and G_i is the deadline for the same surgery. Note that there is no constraint in the problem that prevents the surgery to be performed after its deadline, in which case the surgery's contribution to O_W is larger than one. However, exceeding the deadline in this way is highly undesirable. In Norway, the number of such deadline violations is one measure of a hospital's efficiency. This objective is, therefore, relevant both to the patients, who wish to be treated as soon as possible, and to the hospital, which wishes to adhere to the surgery deadlines. The power of two in Eq. 2.2 is introduced to promote fairness between patients; it is better to have minor increases in waiting time for many patients, than to let a few patients carry the burden by getting very large waiting times.

An objective that is frequently used in the literature on surgery scheduling is the amount of overtime for surgeons. This is relevant in admission planning if surgeons are considered to be a critical resource at this stage. To calculate this objective, one needs to know which surgeon(s) will perform each surgery. This can be pre-defined as a part of the problem formulation, which typically happens in cases where the referring physician will perform the surgery. Alternatively, the choice of surgeon can be considered to be a part of the admission planning problem. Note that in some cases the admission planning problem may consider groups of surgeons, i.e. each intervention is assigned to a specialty, rather than to individual surgeons. In this case, the overtime objective as formulated in Eq. 2.3 only makes sense if all surgeons in a specialty have the same working hours. In this study, all surgeries have been pre-assigned to surgeons as a part of the problem input, and we only consider the total amount of overtime in a plan. We also consider only the responsible surgeons for each surgery; assistant surgeons, or surgeons in training, are ignored. This simplification does not alter the underlying nature of the problem. Our overtime objective can be written as:

$$O_O = \sum_{d=1}^{D} \sum_{k=1}^{K} \max_{i \in [1,N]} \left(\delta_{ik} \sum_{r=1}^{R} \sum_{p=1}^{P} \text{Max} \left((p + \Delta_i - 1)\, x_{ipdr} - E^e_{kd}, 0 \right) \right) \tag{2.3}$$

Here, K is the number of surgeons, E^e_{kd} is the end of working hours for surgeon k on day d, and Δ_i represents the duration of intervention i, given as an integer number of time periods. The input pre-assignment of surgeons is expressed in Eq. 2.3 through the constant

$$\delta_{ik} = \begin{cases} 1, & \text{if intervention } i \text{ is pre-assigned to surgeon } k \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

The objective in Eq. 2.3 is obviously important to surgeons, and will also directly influence hospital costs. Similar objectives can be expressed for other resources. The importance of overtime in our study is motivated by the hospital's need to assess the practical trade-offs that are available between overtime and other aspects of the plan, such as the other two objectives considered herein.

The last objective that we consider is a measure of how long young children will have to wait (on an empty stomach) in the morning before their surgery begins. Such waiting is considered to be increasingly difficult with decreasing age. The problem definition contains an upper limit on the age of children, L, and the age of each patient, A_i, both given in years. We write this objective as

$$O_C = \sum_{i=1}^{N} \sum_{p=1}^{P} \sum_{r=1}^{R} \sum_{d=1}^{D} p\, x_{ipdr}\, \text{Max}(L - A_i + 1, 0) \tag{2.5}$$

This objective is an example of a broader class of objectives relating to a preferred starting time for a given surgery. Other examples include time preferences for elderly people, people who have difficulties arriving at the hospital in the morning (but still arrive on the day of surgery), or people with specific ailments such as diabetes. Our results concerning the "children early" objective therefore also give insight about versions of the problem where such similar objectives are present. The objective in Eq. 2.5 is clearly focused on the well being of the patient and, therefore, comes under the heading "quality of care".

We are now ready to formulate a total, aggregate, objective function. Using a weighted sum for aggregation, this can be written as:

$$O = w_W \frac{O_W}{N_W} + w_O \frac{O_O}{N_O} + w_C \frac{O_C}{N_C} + \sum_{i=1}^{N} \left(1 - \sum_{p=1}^{P} \sum_{d=1}^{D} \sum_{r=1}^{R} x_{ipdr} \right) \Psi_i \tag{2.6}$$

Here, Ψ_i is the cost of not including a surgery i in the plan.

The normalisation factors N_W, N_O and N_C are set so that each normalised, but un-weighted, term is of the same order of magnitude. The value of these factors are set, so that $N_W = O_W$, $N_O = O_O$, and $N_C = O_C$, for some initial solution. Note that these factors are not normalisation factors in the strict mathematical sense, since for a given solution to the problem, each " 'normalised' " objective component value may be larger than one. However, the initial solution is typically from the day before the input problem arises. The number of surgeries that is already planned in the initial solution is therefore typically much larger than the number of new surgeries that has been added to the problem since the initial solution was made. Therefore, for a new solution, each normalised term in Eq. 2.6 is in practice still of the same order of magnitude. The objective weights w_W,w_O and w_C can then be set by a user to reflect the relative importance of their respective objective components, without the need to consider the magnitude of the values for the un-normalised components themselves.

Let us introduce a helping variable y_{ipdr}, which has a value of one if surgery i is under execution in periods p, in room r, and on day d, and zero otherwise:

$$y_{ipdr} = \sum_{p'=\max(p-\Delta_i+1,1)}^{p} x_{ip'dr} \tag{2.7}$$

The optimisation problem is to minimize the objective function in Eq. 2.6, subject to the following constraints:

$$\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{r=1}^{R} x_{ipdr} \leq 1, \quad \forall i \in [1, N] \tag{2.8}$$

$$\sum_{p=1}^{E^s_{kd}-1}\sum_{i=1}^{N}\sum_{r=1}^{R} x_{ipdr}\delta_{ik} = 0, \quad \forall d \in [1, D], k \in [1, K] \tag{2.9}$$

$$\sum_{i=1}^{N} y_{ipdr} \leq 1, \quad \forall p \in [1, P], d \in [1, D], r \in [1, R] \tag{2.10}$$

$$\sum_{p=1}^{P}\sum_{k=1}^{K}\sum_{s=1}^{S} y_{ipdr}\,\delta_{ik}\,\sigma_{ks}\,M_{rpds} = \Delta_i \sum_{p=1}^{P} x_{ipdr}, \quad \forall i \in [1, N], r \in [1, R], d \in [1, D] \tag{2.11}$$

$$\sum_{i=1}^{N}\sum_{r=1}^{R} y_{ipdr}\,\delta_{ik} \leq 1, \quad \forall k \in [1, K], p \in [1, P], d \in [1, D] \tag{2.12}$$

Note that the following parameters are part of the problem definition:

$$\sigma_{ks} = \begin{cases} 1, & \text{if surgeon } k \text{ belongs to specialty } s \\ 0, & \text{otherwise} \end{cases} \tag{2.13}$$

$$M_{rpds} = \begin{cases} 1, & \text{if the master surgery schedule allows operating room } r \text{ to be used by specialty } s \text{ on day } d \text{ and period } p \\ 0, & \text{otherwise} \end{cases} \tag{2.14}$$

The relation in Eq. 2.8 says that each intervention must be planned to start at most once and, if so, it starts in exactly one operating room. Equation 2.9 says that all surgeries performed by surgeon k at any day d must start after the start of working hours for that surgeon on that day, E^s_{kd}. In Eq. 2.10, we state that at most one intervention can happen simultaneously in any operating room. In Eq. 2.11, we say that no surgery can start too late to be completed within the operating room hours that have been pre-assigned to the specialty s of the surgeon k that is assigned to the surgery i. This is the block scheduling constraint, with M_{rpds} representing the master surgery schedule as described in Sect. 1. Note that even if there is no master surgery schedule (open scheduling), this constraint is still necessary to ensure that surgeries are carried out during room opening hours. In this case, M_{rpds} does not distinguish between different specialties, s; it simply reflects the opening hours of each room. The relation in Eq. 2.12 states that the maximum number of simultaneous interventions performed by any surgeon k is one. This constraint has a large impact on the complexity of the problem. It means that it is not enough to choose a room-day combination (denoted by "room-day" in the following) for each intervention, and a sequencing of the interventions in each room-day. We also need to select starting times for each intervention in a way that respects Eq. 2.12. If all surgeons have the same working hours, this constraint is the only reason why the actual sequence of surgeries can affect overtime. Note that in many real world variations of the problem, the constraint in Eq. 2.12 is not included because the availability of surgeons is not considered to be critical at the stage of admission planning. However, at some stage, this constraint always becomes relevant.

3 The algorithm

The literature on search methodologies for surgery scheduling reports the use of many different algorithmic approaches, including integer programming, column generation techniques, meta-heuristic methods and simple assignment heuristics. As explained in Sect. 1.3, problem complexity and size, and the conclusions from the relevant literature, suggest that we consider a meta-heuristic search method for our admission planning problem. A meta-heuristic method will also be easily adaptable to future changes in the problem, such as additional objectives or constraints. We present an iterative improvement approach based on local search. Our baseline algorithm is a steepest variable neighbourhood descent (Hansen and Mladenovic 2005), where each neighbourhood operator is used repeatedly until no improving move is found before changing to another neighbourhood operator in a cyclic fashion. Pseudo-code for this algorithm is given in Fig. 2. Here, the function *SelectNHO* selects the next type of neighbourhood operator to use, starting with the Relocate operator. The function *bestNeighbour* returns the best feasible and improving neighbour of s with the given operator, *nho*. If no such move is found, we have a local optimum for that operator, and *bestNeighbour* returns *NULL*. Enclosing the VND is an iterated local search framework (Lourenco et al. 2002), which diversifies the local optima returned from the variable neighbourhood search, and always continues from the last found local optimum. Figure 1 shows pseudo-code for the algorithm. The function *Accept* sets $b = s$ if s is better than b, and returns s. The function *Diversify* achieves diversification by removing surgeries from the plan (i.e. making them un-served). This is done at each iterated local search iteration by removing all surgeries that are assigned to one or two randomly chosen surgeons, or that are allocated to one or two randomly chosen operating rooms. Note that only unlocked ("served but not fixed") surgeries are affected by the diversification.

Fig. 1 Pseudo-code for our ILS implementation. See the text for details

ILS
input: solution s
returns: improved solution

1. $b = s$
2. while(! stop)
   (a) $s = VND(s)$
   (b) $s = Accept(s, b)$
   (c) $s = Diversify(s)$
3. return b

Fig. 2 Pseudo-code for our VND implementation. See the text for details.

VND
input: solution s
returns: improved solution

1. $nho = SelectNHO$
2. while($nhc! = NULL$)
   (a) $s* = bestNeighbour(nho, s)$
   (b) while($s*! = NULL$)
      i. $s = s*$
      ii. $s* = bestNeighbour(nho, s)$
   (c) $nho = SelectNHO$
3. return s

Initial solutions are generated by a greedy insertion heuristic. Un-served surgeries are inserted into the plan as early as possible, and in an order sorted according to increasing referral dates, increasing deadline, and decreasing surgery durations.

We model a solution as one sequence of surgeries for each room-day. For ease of implementation, we also keep any un-scheduled surgeries in a separate schedule, referred to below as the "un-served schedule". When the various diversification routines remove surgeries from the solution, these are put in the "un-served schedule". We use two different move operators, "Relocate" and "Two-Exchange". The Relocate neighbourhood of a solution contains all other solutions that can be reached by moving one surgery from its position in a room-day schedule, to any position in any (possibly the same) room-day schedule. The Relocate move can, in our implementation, also move surgeries from the "un-served schedule" to a real room-day schedule. The Two-Exchange neighbourhood of a solution contains all other solutions that can be reached by exchanging the positions of two surgeries, and this is independent of which room-day schedule each of them is allocated to. The Two-Exchange neighbourhood also includes solutions that result from exchanging a planned surgery with one in the un-served schedule.

Note that to evaluate each room-day sequence, it is necessary to find the best possible schedule for all room-day sequences that satisfy Eq. 2.12. This is, in itself, a search problem, which we have to solve efficiently since it has to be done at every single move evaluation. We solve this by applying the following heuristic to each day under evaluation:

1. Calculate for each surgeon the total workload on the day in question.
2. Sort a list of all rooms according to:
   (a) Decreasing workload for the surgeon of the first entry in the room's sequence on that day.
   (b) Decreasing utilisation of the room's opening hours on that day.
3. Free all time for all surgeons on that day.
4. For each room, in the sorted list of rooms:
   Schedule all surgeries as early as possible (without changing the sequence), and reserve time for the associated surgeons, respecting reservations of surgeon time that have already been made.

4 Test data

Our test data is based on the admission planning problem found in a Norwegian hospital. The number of resources, their capacity, and the number of surgeries are consistent with the operations at the hospital. Also, the weekly master surgery schedule is identical to the one used in their admission planning. The problem has the three objectives described in Sect. 2; waiting time, overtime, and serving children early. At the hospital, the waiting time objective is used actively, although they have not formulated it explicitly in mathematical terms. They simply strive to treat patients as soon as possible, and to avoid overriding any surgery deadlines. The overtime objective is not used explicitly by this hospital during admission planning, but it is still of interest because the hospital, in the future, will need to investigate the possible trade-offs between waiting time and staff size. Another reason for including the overtime objective is generality; as can be seen from the literature Cardoen et al. (2009), this is a frequently used objective. At present, the hospital in question avoids overtime by scheduling interventions based on rules of thumb that are designed to finish all surgeries within normal working hours. We introduce this objective as a replacement for these rules of thumb, and in the test data the opening hours of operating rooms are set to exceed the normal working hours for surgeons by three hours and twenty minutes. All surgeons have the same working hours. The objective of serving children early is in active use at the hospital, but is again handled by applying rules of thumb that ensure a plan that is satisfactory with respect to this objective. A real planner may choose to postpone the planning of a patient until a later day. In our tests, however, we set Ψ_i to a value high enough to totally dominate the other components in Eq. 2.6, to ensure that the algorithm's first priority will be to serve all patients.

To represent a realistic planning situation, each test case contains an initial plan in which many surgeries are locked or fixed to a certain day of intervention. These are the surgeries for which the dates have already been set. They are chosen to reflect the fact that admission planners at the hospital in question want to settle intervention dates within a certain time period after receiving the referral. This is a philosophy which is often applied in Norwegian hospitals. We implement this when generating test cases by locking all surgeries in the initial plans whose referral arrived more than 14 days before the day of planning. In addition, all surgeries that were planned (in the initial plan) to be performed in less than 14 days after the day of planning are also locked to reflect that one normally wants to give patients a certain minimum notice. In our test cases' initial plans, all un-locked surgeries are represented as un-served; that is, not planned in any room-day.

We define three different fitness surfaces through the following sets of objective weights:

1. **WT**: $w_W = 1.0$, $w_O = 0.0$ and $w_C = 0.0$
2. **WOT**: $w_W = 0.5$, $w_O = 0.5$ and $w_C = 0.05$
3. **OT**: $w_W = 0.0$, $w_O = 1.0$ and $w_C = 0.05$

We want to investigate the fitness surfaces of the two objectives, O_W and O_O, in addition to the fitness surface corresponding to a trade-off between them. The reason why we use such a small weight for the objective of serving children early is that our preliminary results showed that it is rather easy to satisfy, given a realistic number of children, and a significant weight on the overtime objective. This is because O_C and O_O are both in significant conflict with O_W, but not with each other. Reducing overtime must be done at the cost of increased patient waiting time, which means that the total period covered by the planned surgeries is increased. This in turn makes it easier to satisfy O_C, since there are more days available for planning children, and since the waiting time objective does not differentiate between children and adults.

We generate 10 test cases representing 10 consecutive days. Each case has about 150 surgeries in total, of which about 100 are already fixed to a date in the initial solution. Each case was tested for each of the above fitness surfaces, giving, in total, 30 benchmark problems.

For each generated case, we generated an initial solution by applying the construction heuristic described in Sect. 3. We then removed all unlocked entries, making them un-served in the initial plans.

To encourage comparative studies, all the test cases are made available in XML format at *www.sintef.no/hospital*.

5 Search space analysis

In order to understand the nature of the defined admission planning problem, and to assess the suitability of our algorithm, we perform some simple search space analysis. First, let us clarify our terminology. We define the *solution space*, as the space of all solutions, feasible or not. On this space, each move operator defines a topology, given as a graph where each solution is a vertex, and where edges between two vertices exist if and only if it is possible to reach one of the corresponding solutions from the other by a single step with the move operator. Note that this graph is not necessarily connected; this depends on the nature of the move operator in question (see for example Barnes and Colletti 2002). Since we have two move operators, they each define their own neighbourhood graph, or network topology. Each of the two graphs for our two operators have diameter $n-1$, where n is the number of surgeries. The graph diameter defines the maximum number of moves that is necessary to transform one solution into any other solution. By distance between two solutions, we refer to the minimum number of edges in the neighbourhood graph that must be traversed in order to move from one solution to the other. If two solutions are in separate, non-connected parts of the graph, we define the distance between them to be infinite.

Now, hard constraints may leave parts of the solution space infeasible. We define the *search space* of a given move operator as the feasible solutions that are included as vertices in the associated neighbourhood graph. Obviously, hard constraints may reduce the connectedness of the graph, converting it to a set of an increased number of isolated sub graphs. For each solution in the search space, there is an objective value as defined in Eq. 2.6. These values define what is often called a *fitness surface* over the search space (see for example Hoos and Stutzle 2005). Obviously, the shape of this surface will depend on our weights in Eq. 2.6. Moving in the search space corresponds to moving on the fitness surface.

The level of difficulty of solving a problem by local search methods depends on several things besides the size of the search space. One indication can be derived from how rugged, or how smooth, the fitness surface is. A rugged surface will have many local optima and finding the good ones becomes difficult. On the other hand, a smooth surface has fewer local optima, and is also more likely to serve as a guide for a steepest-descent method searching for better solutions. The existence of large plateaus on the fitness surface will present challenges to a local search algorithm, even if it is very smooth. In addition to ruggedness, it is useful to consider the distribution of local optima. Are they uniformly distributed throughout the search space, or are they clustered in a small region? Is there any relationship between their objective values and their distance to the closest optimal solution?

5.1 Ruggedness

Let us consider first a measure of ruggedness of the fitness surface. We would like to do this by applying a measure of the correlation between solutions' objective values and their dependence on the distance between the solutions. Since it is impractical to sample the entire solution space, we estimate this correlation measure by collecting objective values from a random walk through the search space (Weinberger 1990). Based on this information, we can calculate an empirical autocorrelation function (Hoos and Stutzle 2005; Reeves 2005), as

$$r(d) := \frac{1/(m-d)\sum_{k=1}^{m-d}(g_k - \bar{g})\,(g_{k+d} - \bar{g})}{1/m\sum_{k=1}^{m}(g_k - \bar{g})^2} \tag{5.1}$$

Here, d signifies the distance at which correlation is measured, m is the number of steps of the random walk, and g_k is the objective value that is observed at step k. $\bar{g}$ is the average objective value over all m steps. Note that in this formula, the distance between solutions is measured as a number of steps along a random walk. It may be that between two solutions at a given distance d in the random walk, it is actually possible to move from one to the other in less than d steps. That is, the actual distance between them in the search space can be less than d. This is more likely to happen if d is larger, and, of course, it cannot happen for $d = 1$. As a measure of correlation between objective values at different distances, the above empirical autocorrelation function is therefore primarily of interest for small d, and particularly

Table 1 $r(1)$ for each run. *WT*, *WOT* and *OT* refer to the names of the fitness surfaces defined in Sect. 4

	Relocate	Two-Exchange
WT	0.991	0.990
WOT	0.991	0.984
OT	0.995	0.987

for $d = 1$. A correlation close to one means that the landscape is smooth, in the sense that neighbouring solutions tend to have similar values. Smooth landscapes are particularly appropriate for local search algorithms such as variable neighbourhood descent, which depends on the local shape of the fitness surface to guide the search towards good solutions. At the other extreme, if $r(1)$ is zero, the fitness surface is extremely rugged, such as that which would be obtained with a random assignment of fitness values to solutions.

The observations one makes during such a random walk, of course, depend on the move operator that is used. We ran random walks of 20 000 iterations for each of the move operators Relocate and Two-Exchange. This was done for each of the three different fitness surfaces corresponding to the weighting of objective components given in Sect. 4. The case considered was, otherwise, identical in all runs. The results are summarised in Table 1, which shows $r(1)$ for each of these runs.

As we can see, $r(1)$ is close to one for all these tests, indicating that the corresponding fitness surfaces are indeed quite smooth. Caution should be applied when interpreting these results, however, since ruggedness is not always a good measure of the difficulty of addressing an optimisation problem (Hoos and Stutzle 2005). The results in Table 1 indicate that the Two-Exchange neighbourhood produces slightly more rugged fitness surfaces than the Relocate neighbourhood.

5.2 Fitness and distance between solutions

In addition to the ruggedness and number of local optima, the distribution of these in the search space is of critical importance to a local search algorithm. If all local optima are confined to a small connected region in the search space, finding a good solution is much easier than if they are more or less uniformly distributed.

In the following analysis, we require a thorough and rigorous way of calculating the actual distance between two solutions in the search space. For both our two operators, there are exact measures for their respective distances. Both are derived from permutation theory, and both can be computed efficiently (Schiavinotto and Stutzle 2007). We associate a distance measure with the Two-Exchange operator that can be calculated in linear time, using the algorithm given in Schiavinotto and Stutzle (2007). For the Relocate operator, we use Ulam's metric (Beyer et al. 1972), whose calculation is based on the calculation of the longest increasing subsequence in a permutation representing our solution. There are a number of algorithms for calculating this; we have used the one from Orlowski and Pachter (1989), which has worst case complexity $O(n \log(r))$, where r is the length of the longest increasing subsequence.

Now, consider the degree of correlation between inter-solution distance and the fitness value. Of particular interest are the average distances from each local optimum

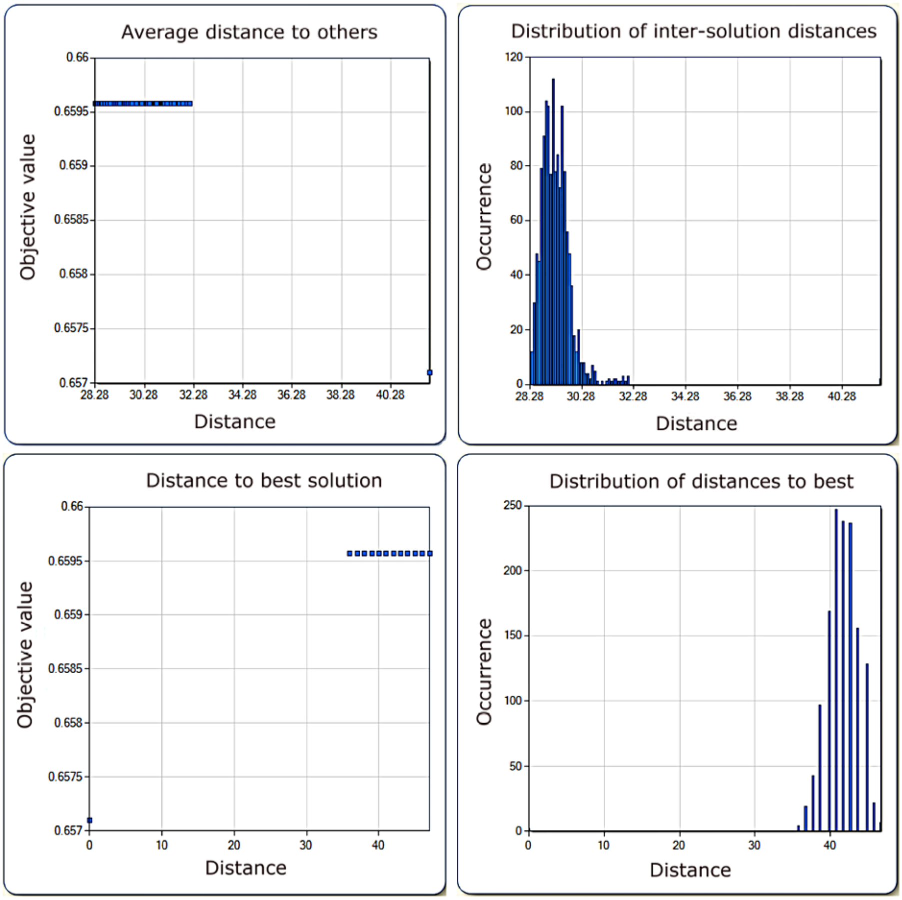

Fig. 3 "Relocate" distances for the "OT" fitness surface. See the text for details.

to all other local optima, and the average distances from each local optimum to the closest global optimum. If there is a clear correlation between distances and objective values, it would provide evidence for a "big valley" structure on the search space, in which good local optima are close together (relative to the neighbourhood graph diameter). This property generally makes a problem suitable for solving with local search methods. Lacking information about the global optima of our test problems, we have to use the best solutions we know from our experiments instead. One should be aware that this can lead to erroneous conclusions, since the best known solutions may in principle be far from any global optimum.

We wanted to investigate the distribution of local optima, and the relationship between distance and the fitness value. We performed ten runs, of 2 000 iterations each, for each of the three fitness surfaces, and for all test cases. It was confirmed that using both our operators in the variable neighbourhood search yields a more effective search than using only one of them. In this analysis, therefore, we logged all solutions that are local optima with respect to both operators. The results for one of

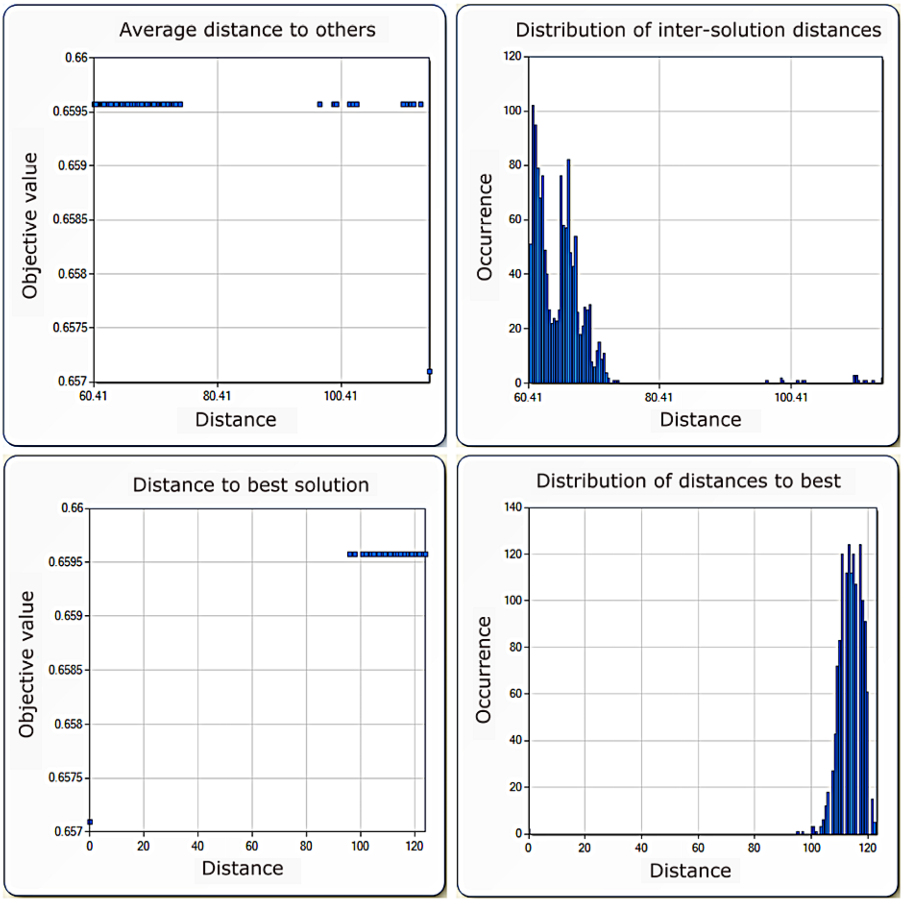

Fig. 4 "Two-Exchange" distances for the "OT" fitness surface. See the text for details

the test cases can be seen in Figs. 3, 4, 5, 6, 7, and 8. Each of these figures have the same structure: At the top left we present the average distance for each observed local minimum to the other local minima, and its relation to their objective value. At the top right, we have their observed distribution. At the bottom left, we see the distance from each local minimum to the best known solution, and at the bottom right, we outline the observed distribution of these distances.

In Figs. 3 and 4, we see the results for the "overtime" fitness surface. As we can see in Fig. 3, the average distances between the local optima is around 30 Relocate moves. This is rather small compared to the diameter of the neighbourhood graph for this test case, which is 151. This means that the local optima are not distributed evenly throughout the search space but, rather, they are concentrated in a smaller region. Notice that most local optima have the same (or very similar) normalized objective values. This kind of "value level" is observed for the other test cases as well, for this fitness surface. However, for the case illustrated in Figs. 3 and 4 we have found a best known solution that has a value below this level. Moreover, the

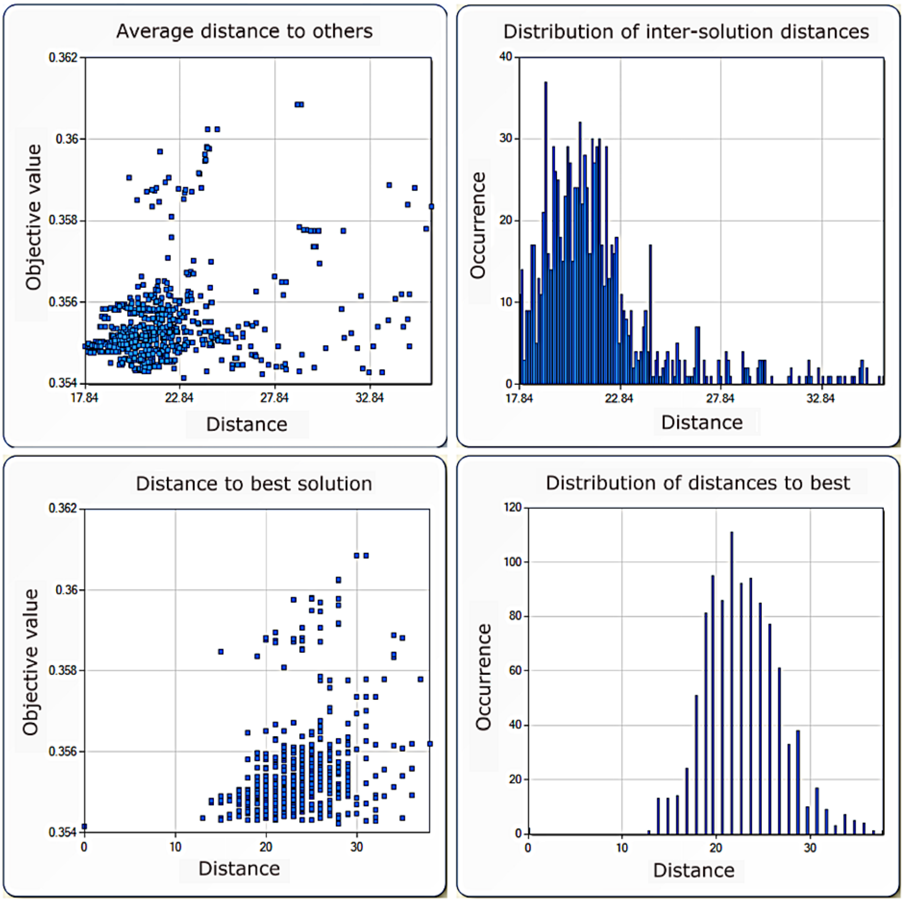

Fig. 5 “Relocate” distances for the “WT” fitness surface. See the text for details

best known solution is on average further from other local optima than are other local optima. It seems that the search easily finds a solution at the mentioned value level, but that finding the best known solution is very difficult. For the other test cases, we only find such “value levels”. That is, many solutions have the same—best known—value. This is reflected in the computational results in Sect. 6. It is quite likely that better solutions exist for those test cases as well, although we have not been able to find any. It seems that this fitness surface does little to guide the local search towards globally optimal solutions, but that many different solutions at a certain quality level can be found quite easily.

Figures 5 and 6 show the results of the same analysis for the “waiting time” fitness surface. When measured with the “relocate” distance measure (Fig. 5), the local optima of this surface is typically closer together than was the case for the “overtime” objective. Their objective values vary, but there is a tendency that this variation is smaller amongst the optima that are close to the others on average. The same observation can be made from Fig. 6, in which the “Two-exchange” distance measure

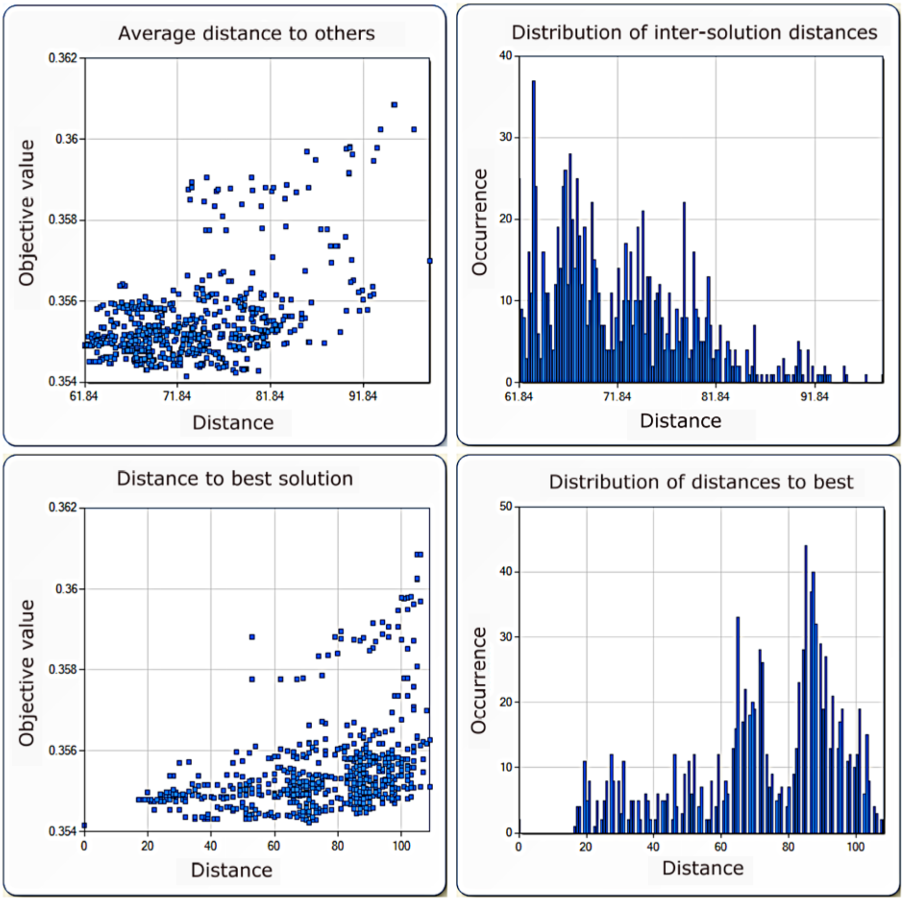

Fig. 6 "Two-Exchange" distances for the "WT" fitness surface. See the text for details

has been used. Here, we also see a more clear correlation between objective value and distance to the best known solution, which indicates that this fitness surface has a structure that can be exploited by a steepest-descent based algorithm. It also indicates that a stronger intensification might improve our algorithm, e.g. by letting the iterated local search accept the best solution as a starting point for the local search, rather than the latest local optimum, which is the current strategy. However, a note of caution is in order when interpreting these results. Even if the distance measures exactly reflect the minimum number of moves between local optima in the associated neighbourhood graph, it does not always equal the minimum number of moves in the (feasible) search space. This is because the shortest path between two solutions in the neighbourhood graph may lead through infeasible regions in the solution space. The actual distance in the search space will then be longer, or even infinite. Hard constraints may thus represent barriers that inhibit the search's progress towards better solutions even if this analysis shows that the fitness landscape has a benign shape.

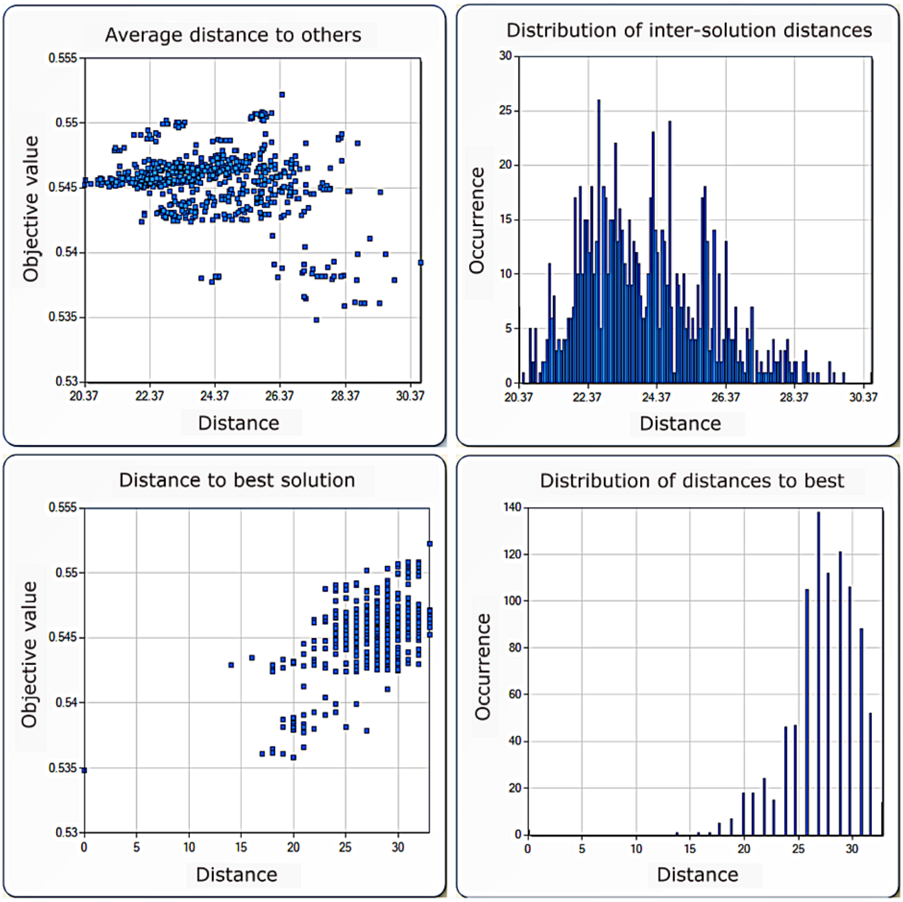

Fig. 7 "Relocate" distances for the "WOT" fitness surface. See the text for details

For the fitness surface of the combined objective function, "WOT", the results in Figs. 7 and 8 show properties that are a combination of those observed for the two fitness surfaces "OT" and "WT". The local optima are quite close together, especially for the Relocate neighbourhood. The correlation between objective value and distance to the best known solution is less clear than for the "WT" surface, while traces of the plateaus in the "OT" surface can be observed.

Note that while our analysis shows properties of the various fitness surfaces, the sample of local optima that is obtained is related to our algorithm. A different algorithm might sample different local optima, in different regions of the search space, and might (in principle) get different results. We are, however, interested in assessing the suitability of our algorithm for this problem, and are, therefore, content to analyse the fitness surfaces as they are experienced by our algorithm. From the above analysis, it is clear that the properties of the fitness surface that are experienced by our local search algorithm vary substantially with the weighting of the three objective components.

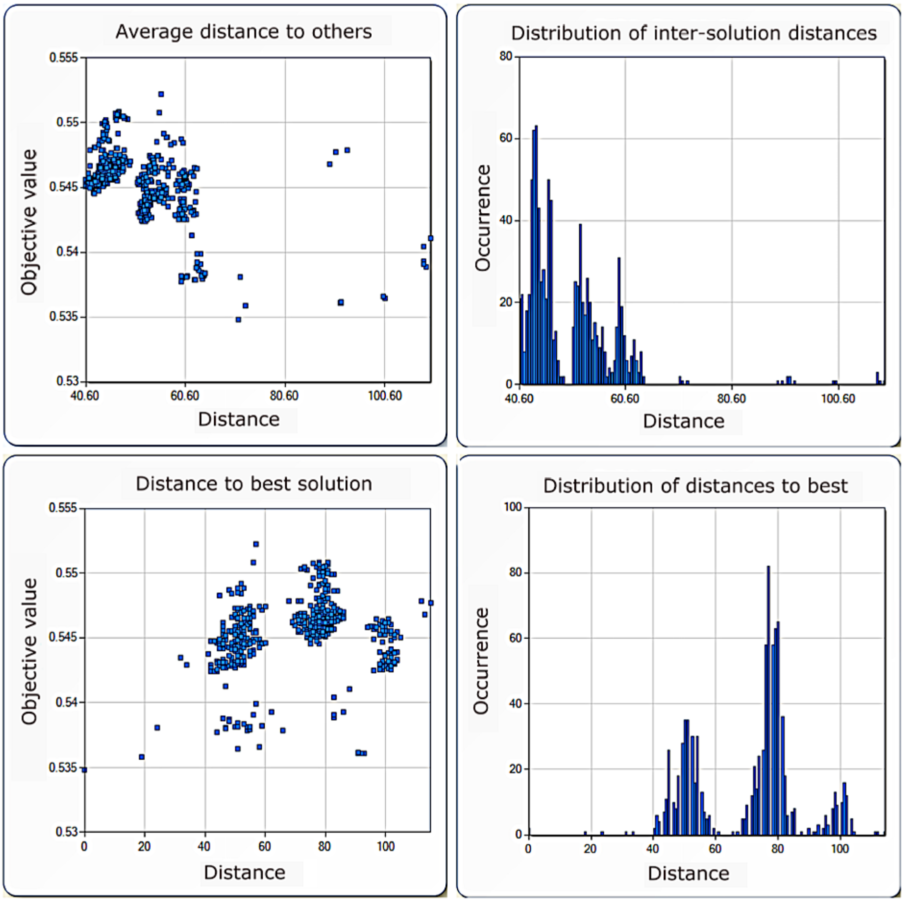

Fig. 8 "Two-Exchange" distances for the "WOT" fitness surface. See the text for details

6 Results

Our experiments were based on the admission planning problem found at a certain Norwegian hospital. Technical issues prevented us from retrieving historical surgery data with all the necessary details. For the time being, therefore, we have worked with an artificially generated set of test instances. These are, however, based on the details of the hospital's actual admission planning problem, as discussed in Sect. 4. For testing purposes, each case is represented in triplets, one for each of the three fitness surfaces as defined in Sect. 4. In this section, we report benchmarking results for each of these cases. All computations where carried out on a Hewlett-Packard laptop with a 2.37 GHz Intel Dual core processor and 1.98 GB of Ram.

The results that are summarised in Tables 2, 3, and 4, are based on ten runs for each problem, of 1 000 iterations each. All the tables have the same layout. The best known solutions values (*BKV*) are the objective values of the best solutions found during these and other experiments, using the same algorithm. *Att.* is the fraction of

Table 2 Benchmark results for the *WOT* fitness landscape. See the text for details

Cases	BKV	Att.	Att. 1%	Avg.it. 1%	Var.it. 1%	Avg.time 1%	Var.time 1%
WOT1	0.53152	0%	30%	63.7	0.0928	47.3	0.0243
WOT2	0.52764	10%	90%	54.8	0.0824	37.5	0.0424
WOT3	0.31494	20%	100%	40.3	0.0566	27.0	0.0524
WOT4	0.24893	20%	100%	32.3	0.0854	16.0	0.08400
WOT5	0.34333	0%	100%	44.8	0.345	21.0	0.316
WOT6	0.40024	20%	90%	60.6	1.00	29.4	0.959
WOT7	0.46814	80%	100%	47.1	0.377	21.3	0.358
WOT8	0.44070	50%	100%	109	0.951	50.628	0.878
WOT9	0.42898	10%	100%	71.3	0.878	32.6	0.795
WOT10	0.2947	10%	90%	33.7	0.0907	18.8	0.0809

Table 3 Benchmark results for the *OT* fitness landscape. See the text for details

Cases	BKV	Att.	Att. 0.5%	Avg.it. 0.5%	Var.it. 0.5%	Avg.time 0.5%	Var.time 0.5%
OT1	0.65115	0%	0%	–	–	–	–
OT2	0.65082	100%	100%	44.3	0.195	31.1	0.200
OT3	0.27067	100%	100%	34.3	0.0569	21.4	0.0551
OT4	0.13360	100%	100%	31.4	0.328	14.2	0.414
OT5	0.32065	100%	100%	23.8	0.0366	11.3	0.0439
OT6	0.43359	100%	100%	24.5	0.0524	11.9	0.0511
OT7	0.50260	100%	100%	99.1	1.06	49.4	1.08
OT8	0.46882	90%	90%	35.1	0.0474	18.2	0.0408
OT9	0.45308	100%	100%	31.7	0.0347	16.4	0.0426
OT10	0.2346	100%	100%	31.9	0.217	17.5	0.243

runs that obtained the best known solution. All surgeries were planned for all cases, and the objective values in the tables are calculated using the objective component weights and normalisation factors that are stored with each test case. $Avg.it.x\%$ is the number of iterations necessary to reach a $x\%$ deviation from BKV. $Var.it.x\%$ is the corresponding variation coefficient, which is the standard deviation divided by the mean. $Avg.time.x\%$ and $Var.time.x\%$ are the corresponding measures applied to calculation time (seconds). Averages and variations are calculated over those runs that achieved the given deviation, as indicated in the columns labelled "$Att.x\%$".

A small note is in order in connection with Table 3. The reason that the algorithm is not able to produce solutions with zero overtime is that there are surgeries that are planned for overtime in the initial solution. Some of these are locked, and cannot be moved by the algorithm.

Table 4 Benchmark results for the *WT* fitness landscape. See the text for details

Cases	BKV	Att.	Att. 0.5%	Avg.it. 0.5%	Var.it. 0.5%	Avg.time 0.5%	Var.time 0.5%
WT1	0.35410	60%	100%	19.4	1.50	15.6	1.88
WT2	0.34904	50%	100%	4.00	0.316	2.35	0.159
WT3	0.35024	40%	100%	4.80	0.664	1.63	0.52
WT4	0.35527	60%	100%	7.20	1.617	3.18	1.80
WT5	0.35775	60%	100%	2.80	0.915	0.85	0.534
WT6	0.35553	30%	100%	6.50	0.635	2.27	0.540
WT7	0.37957	10%	60%	3.83	0.681	1.62	0.693
WT8	0.34875	60%	100%	2.30	0.391	0.872	0.107
WT9	0.34373	20%	50%	2.20	0.340	0.841	0.0604
WT10	0.34527	30%	60%	2.17	0.560	1.03	0.313

7 Conclusions and future work

We have presented a model for the complex admission planning problem, in which intervention assignment and scheduling is combined. The problem also includes scheduling interventions for each surgeon. A meta-heuristic resolution method was presented, along with its underlying move operators and associated distance measures. The problem's resulting search space was analysed for three different fitness surfaces, representing different compromises between patient waiting time, surgeon overtime, and waiting time for children in the morning on the day of surgery. An analysis of fitness landscape ruggedness and a fitness-distance correlation between local optima has been presented. We have shown that although the fitness surfaces are comparatively smooth for all the fitness surfaces, with a high one-step fitness correlation function, the different fitness surfaces pose quite different challenges for a local search algorithm. The waiting time objective seems to present a fitness surface that is quite suitable for guiding the local search algorithm, and that could be exploited further by an increased intensification in the search. The overtime fitness surface, on the other hand, seems to contain distinct levels of similar objective values, and it seems difficult for a local search algorithm to find regions of the search space with better objective values. This indicates that more powerful diversification schemes, or indeed complete restarts, could be a way forward for such problems.

Finally we have presented computational results for a set of realistically sized benchmarks that were generated based on the characteristics of the admission planning problem in a Norwegian hospital. The benchmark problem description and test data are available at www.sintef.no/hospital.

Future work concerns the development of algorithms that are robust over an extended problem model, and to test such approaches on real world historical data. One of the issues of interest is to study robust and adaptable local search algorithms, in order to better handle the differing properties of different fitness surfaces. Problem extensions will include decision variables, constraints, and preferences that arise in planning closer to the day of surgery. At the stage of admission planning, extension regarding post-operative resources and uncertainty in surgery durations and emergency occurrence are important. The simultaneous assignment of operating rooms and surgeons should be generalised to any set of critical resources, including e.g. anaesthetists and mobile equipment. The master surgery schedule often implicitly represents several underlying constraints and preferences. In the case of open scheduling, a selection of these should be included to ensure that the resulting plans are still acceptable. We also believe that a proper multi-objective treatment of these optimisation problems can add to the power of decision support tools where the search for the best trade-offs is important.

Acknowledgements We wish to thank the reviewers for their constructive contributions.

This work is supported by the Research Council of Norway, through the HOSPITAL project.[1] and the UK Engineering and Physical Sciences Research Council (EPSRC).

References

Barnes, J.W., Colletti, B.W., Neuway, D.L.: Using group theory and transition matrices to study a class of metaheuristic neighborhoods. Eur. J. Oper. Res. **138**(3), 531–544 (2002)

Beliën, J., Demeulemeester, E.: Building cyclic master surgery schedules with leveled resulting bed occupancy. Eur. J. Oper. Res. **176**(2), 1185–1204 (2007)

Beliën, J., Demeulemeester, E.: A branch-and-price approach for integrating nurse and surgery scheduling. Eur. J. Oper. Res. **189**(3), 652–668 (2008). 0377-2217 DOI:10.1016/j.ejor.2006.10.060

Beyer, W.A., Stein, M.L., Ulam, S.M.: Metric in biology, an introduction. Preprint LA-4937, University of California, Los Alamos (1972)

Blake, J.T., Carter, M.: Surgical process scheduling: a structured review. J. Soc. Heal. Syst. **5**(3), 17–30 (1997)

Blake, J.T., Carter, M.W.: A goal programming approach to strategic resource allocation in acute care hospitals. Eur. J. Oper. Res. **140**(3), 541–561 (2002)

Blake, J.T., Dexter, F., Donald, J.: Operating room managers' use of integer programming for assigning block time to surgical groups: a case study. Anesth. Analg. **94**(1), 143–148 (2002)

Burke, E.K., De Causmaecker, P., Vanden Berghe, G., Van Landeghem, H.: The state of the art of nurse rostering. J. Sched. **7**(6), 441–499 (2004). 10.1023/B:JOSH.0000046076.75950.0b

Cardoen, B., Demeulemeester, E., Beliën, J.: Operating room planning and scheduling: a literature review. Eur. J. Oper. Res. **201**(3), 921–932 (2009). 0377-2217 DOI:10.1016/j.ejor.2009.04.011

Cardoen, B., Demeulemeester, E., Beliën, J.: Optimizing a multiple objective surgical case sequencing problem. Int. J. Prod. Econ. **119**(2), 354–366 (2009). 0925-5273 DOI:10.1016/j.ijpe.2009.03.009

Cardoen, B., Demeulemeester, E., Beliën, J.: Sequencing surgical cases in a day-care environment: an exact branch-and-price approach. Comput. Oper. Res. **36**(9), 2660–2669 (2009). 0305-0548 DOI:10.1016/j.cor.2008.11.012

Chaabane, S., Meskens, N., Guinet, A., Laurent, M.: Comparison of two methods of operating theatre planning: application in Belgian hospital. In: 2006 International Conference on Service Systems and Service Management, Troyes (2006)

Charnetski, J.R.: Scheduling operating room surgical procedures with early and late completion penalty costs. J. Oper. Manag. **5**(1), 91–102 (1984)

Denton, B., Viapiano, J., Vogl, A.: Optimization of surgery sequencing and scheduling decisions under uncertainty. Heal. Care Manag. Sci. **10**(1), 13–24 (2007)

[1] See www.sintef.no/hospital.

Dexter, F., Macario, A.: Changing allocations of operating room time from a system based on historical utilization to one where the aim is to schedule as many surgical cases as possible. Anesth. Analg. **94**(5), 1272–1279 (2002)
Fei, H., Combes, C., Chu, C., Meskens, N.: Endoscopies scheduling problem: a case study. In: IFAC Symposium on Information Control Problems in Manufacturing (2006)
Gendreau, M., Ferland, J., Gendron, B., Hail, N., Jaumard, B., Lapierre, S., Pesant, G., Soriano, P.: Physician scheduling in emergency rooms. In: Burke E.K., Rudov H. (eds.) PATAT 2006, pp. 2–14. Faculty of Informatics, Masaryk University Brno, The Czech Republic (2006)
Guinet, A., Chaabane, S.: Operating theatre planning. Int. J. Prod. Econ. Plan. Control Prod. Syst. **85**(1), 69–81 (2003)
Hans, E., Wullink, G., van Houdenhoven, M., Kazemier, G.: Robust surgery loading. Eur. J. Oper. Res. **185**(3), 1038–1050 (2008)
Hansen, P., Mladenovic, N.: Variable neighbourhood search. In: Burke, E.K., Kendall, G. (eds.) Search Methodologies—Introductory Tutorials in Optimization and Decision Support Techniques, vol. 1, pp. 211–238. Springer, Berlin (2005)
Hoos, H.H., Stutzle, T.: Stochastic Local Search: Foundations and Applications. Morgan Kaufmann, San Mateo (2005)
Jebali, A., Alouane, A.B.H., Ladet, P.: Operating rooms scheduling. Int. J. Prod. Econ. Control Manag. Prod. Syst. **99**(12), 52–62 (2006)
Dia Lourenco, H.R., Martin, O.C., Stutzle, T.: Iterated local search. In: Glover, F., Kochenberger, G. (eds.) Handbook of Metaheuristics, pp. 321–353. Kluwer Academic, Norwell (2002)
Macario, A., Vitez, T.S., Dunn, B., McDonald, T.: Where are the costs in perioperative care?: Analysis of hospital costs and charges for inpatient surgical care. Anesthesiology **83**(6), 1138–1144 (1995)
Marcon, E., Kharraja, S., Simonnet, G.: Minimization of the risk of no realization for the planning of the surgical interventions into the operation theatre. In: 8th IEEE International Conference on Emerging Technologies and Factory Automation, pp. 675–680. Antibes-Juan les Pins, France (2001)
Noyan Ogulata, S., Erol, R.: A hierarchical multiple criteria mathematical programming approach for scheduling general surgery operations in large hospitals. J. Med. Syst. **27**(3), 259–270 (2003)
Orlowski, M., Pachter, M.: An algorithm for the determination of a longest increasing subsequence in a sequence. Comput. Math. Appl. **17**(7), 10735 (1989)
Ozkarahan, I.: Allocation of surgical procedures to operating rooms. J. Med. Syst. **19**(4), 333–352 (1995). DOI:10.1007/BF02257264
Ozkarahan, I.: Allocation of surgeries to operating rooms by goal programing. J. Med. Syst. **24**(6), 339–378 (2000)
Pham, D.-N., Klinkert, A.: Surgical case scheduling as a generalized job shop scheduling problem. Eur. J. Oper. Res. **185**(3), 1011–1025 (2008). 0377-2217 DOI:10.1016/j.ejor.2006.03.059
Przasnyski Z.H.: Operating room scheduling—a literature review. AORN J. **44**(1), 67–76 (1986)
Reeves, C.R.: Fitness landscapes. In: Burke, E.K., Kendall, G. (eds.): Search Methodologies, pp. 587–610. Springer, Berlin (2005)
Roland, B., Di Martinelly, C., Riane, F.: Operating theatre optimization: a resource-constrained based solving approach. In: Service Systems and Service Management, 2006 International Conference on, vol. 1, pp. 443–448 (2006)
Santibanez, P., Begen, M., Atkins, D.: Surgical block scheduling in a system of hospitals: an application to resource and wait list management in a British Columbia health authority. Heal. Care Manag. Sci. **10**(3), 269–282 (2007)
Schiavinotto, T., Stutzle, T.: A review of metrics on permutations for search landscape analysis. Comput. Oper. Res. **34**(10), 3143–3153 (2007). 0305-0548 DOI:10.1016/j.cor.2005.11.022
Sier, D., Tobin, P., McGurk, C.: Scheduling surgical procedures. J. Oper. Res. Soc. **48**(9), 884–891 (1997)
Weinberger, E.: Correlated and uncorrelated fitness landscapes and how to tell the difference. Biol. Cybern. **63**(5), 325–336 (1990). DOI:10.1007/BF00202749
Weiss, E.N.: Models for determining estimated start times and case orderings in hospital operating rooms. IIE Trans. **22**(2), 143–150 (1990)
Zhou, J., Dexter, F.: Method to assist in the scheduling of add-on surgical cases-upper prediction bounds for surgical case durations based on the log-normal distribution. Anesthesiology **89**(5) (1998)