Expert Systems with Applications 42 (2015) 4029–4041

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Bi-criteria appointment scheduling of patients with heterogeneous service sequences

Alireza Saremi [a], Payman Jula [b,*], Tarek ElMekkawy [c], Gary G. Wang [d]

[a] Fraser Health Authority, Surrey, BC, Canada
[b] Beedie School of Business, Simon Fraser University, Vancouver, BC V5A 1S6, Canada
[c] Department of Mechanical and Industrial Engineering, Qatar University, Doha, Qatar
[d] Mechatronic Systems Engineering, Simon Fraser University, Vancouver, BC V3T 0A3, Canada

ARTICLE INFO

Article history:
Available online 13 January 2015

Keywords:
Outpatient and surgery scheduling
Simulation-based optimization
Mathematical programming
Multiobjective Tabu search
Multiagent optimization

ABSTRACT

This article addresses the challenges of scheduling patients with stochastic service times and heterogeneous service sequences in multi-stage facilities, while considering the availability and compatibility of resources with presence of a variety of patient types. The proposed method departs from existing literature by optimizing the scheduling of patients by integrating mathematical programming, simulation, and multiobjective tabu search methods to achieve our bi-objectives of minimizing the waiting time of patients and the completion time of the facility. Through intensive testing, the performance of the proposed approach is analyzed in terms of the solution quality and computation time, and is compared with the performance of the well-known method, Non-Dominated Sorting Genetic Algorithm (NSGA-II). The proposed method is then applied to actual data of a case study operating department in a major Canadian hospital and promising results have been observed. Based on this study, insights are provided for practitioners.

1. Introduction

Advances in healthcare in recent years accompanied several efforts to improve the efficiency of operations in healthcare in order to reduce the expenses in this sector. Accordingly, many studies have been carried out to improve the efficiency of scheduling in order to address the increase in demand for outpatient and inpatient services. In this article, we consider multi-stage facilities that serve patients of different types with non-identical stochastic service time at each stage. We assume heterogeneous service sequences; i.e., each patient type follows a specific order to visit stages of the facility that may vary from type to type. For instance, a surgical patient may go through stages such as reception, pre-operation, operation, and post-operation, while a checkup patient may undergo a different sequence of stages. In this manuscript, scheduling refers to the determination of the arrival time for each patient to the facility in order to minimize bi-objectives of the waiting time of patients and the completion time of the facility.

Relevant literature indicates that analytical methods and simulation studies have been used to solve the problem of appointment scheduling and planning in healthcare, including inpatient and surgical facilities. Typically, optimization methods use analytical approaches to achieve optimal (or near optimal) solutions. These approaches generally have difficulty addressing large and complex systems. On the other hand, simulation methods can address many complexities in large systems. However, simulation methods are time-consuming and often do not deliver a competitive optimization strategy (Cayirli & Veral, 2003). A gap still exists in the literature for efficient and effective methods to address the challenges in scheduling of such services. In this article, efficiency of a method refers to the amount of computation time required by the method to produce results, while effectiveness addresses the quality of solutions generated by the method. One of the contributions of this article is that we target this gap by integrating analytical methods and simulation. Another contribution is that in contrast with commonly used single objective optimization methods, our approach provides a Pareto front for bi-objectives of patients' average waiting time and the facility completion time. The average waiting time refers to the time in which patients have to wait to receive various services in the facility, while facility completion time refers to the time that the last patient leaves the facility. The Pareto front (also known as Pareto set, or Pareto frontier) is the set of choices that are not strictly dominated by another point in the objective space. Finally, we depart from existing literature by considering the

* Corresponding author.
E-mail addresses: Alireza.Saremi@fraserhealth.ca (A. Saremi), pjula@sfu.ca (P. Jula), tmekkawy@qu.edu.qa (T. ElMekkawy), gary_wang@sfu.ca (G.G. Wang).

http://dx.doi.org/10.1016/j.eswa.2015.01.013
0957-4174/

health centers that serve patients with heterogeneous service sequences.

This work proposes an optimization method termed multi-agent tabu search (MATS), which simultaneously addresses bi-objectives of minimizing patients' waiting time and the clinic's completion time. MATS uses mathematical programming (MP), tabu search, and simulation model. The MP model provides MATS with promising initial solutions. Tabu search then improves the initial solution by searching for optimal schedules by running a number of agents in parallel. The agents seek the non-dominated solutions of the problem and share information with each other to improve the search performance of the algorithm. In order to capture the complexity of multi-stage facilities, MATS is performed on a discrete event simulation. Therefore, MATS method benefits from the flexibility of simulation, and the power of mathematical programming optimization to find Pareto fronts.

In order to evaluate the performance of the proposed method, we developed several test problems with a range of important factors such as, the number of patients and patient types, and the coefficient of variation of service times. We compared the performance of MATS with Non-Dominated Sorting Genetic Algorithm (NSGA-II) in terms of quality of solutions and computation time. NSGA-II has been selected for evaluating performances since it is one of the most powerful methods in the field of multiobjective optimization, and has recently been used in healthcare appointment scheduling with promising results (Gul, Denton, Fowler, & Huschka, 2011). To measure the quality of solutions, we use the hyper-volume and spacing performance indicators, which will be explained later in this article.

This paper has been organized as follows: Section 2 presents a literature review for the appointment scheduling problem in surgical and outpatient settings. Section 3 discusses the problem definition. Section 4 describes the architecture of the proposed approach and describes different components of the algorithm. Section 5 gives the design of experiments and analysis of the results. Section 6 provides a case study of an OR department in a major Canadian hospital. Section 7 provides insights for practitioners. Finally, Section 8 provides the conclusions and directions for future work.

2. Literature review

This section presents a brief review of the relevant literature. Here, we consider either articles that study the appointment scheduling problem as a multi-stage facility, or those which use discrete event simulation or mathematical programming in their study. We further divide appointment scheduling into outpatient appointment scheduling and surgery scheduling. For a more comprehensive review of literature, readers are encouraged to refer to Cayirli and Veral (2003), and Gupta and Denton (2008) for general outpatient appointment scheduling, as well as Blake and Carter (1997) and Cardoen, Demeulemeester, and Belien (2010) for surgery scheduling.

We divided the relevant literature into three categories: optimization studies, simulation studies, and a combination of the two. Many articles in optimization studies use analytical methods to address the appointment scheduling problem. Although the major benefit of analytical methods is their ability to reach optimal solutions, they may not easily represent all the details and constraints of complex systems. Therefore, many analytical methods simplify the system or relax some of the constraints in order to solve the optimization problem. For instance, the queuing theory is an analytical method that is widely used to address the clinic appointment scheduling problem. Cayirli and Veral (2003) stated that most studies in this domain assumed steady state behavior for the system, which is hardly achievable in healthcare environments. They further added that many optimization methods considered only single-stage systems or made strong assumptions on the distribution of the service time. For instance, special properties of exponential or Erlang distributions used for service times in outpatient appointment scheduling. In addition, Klassen and Yoogalingam (2009) reported that most proposed analytical methods are only valid for problems with a few patients. Begen, Levi, and Queyranne (2012) addressed the problem of appointment scheduling with general discrete probability distributions. However, they considered a single stage facility. Recently, there have been several reports on applications of genetic algorithms in healthcare scheduling at individual departments (see, e.g., Petrovic, Morshed, & Petrovic, 2011).

Another analytical method is mathematical programming (MP) which has been used in patient scheduling and surgery departments. Similar to other analytical methods, MP cannot easily accommodate the complexities and environmental parameters arise in the complex–large systems. A major shortcoming of MP models (except for stochastic programming) is that they are incapable of addressing the stochastic nature of healthcare scheduling problem. Although stochastic programming can address the uncertainty in patient scheduling, most of the models in this area are either overly simplified or analytically intractable, and have been solved using approximation methods (e.g., see Lamiri, Grimaud, & Xie, 2009; Min & Yih, 2010). Another major concern in stochastic programming modeling methods is that the computation time of reaching the optimal solutions is significantly higher than that of deterministic models. In addition, almost none of the studies that consider multi-stage facilities cover patients with different service sequences. The only exceptions are Pham and Klinkert (2008) and Gartner and Kolisch (2014) which address the deterministic version (in terms of processing times) of the problem.

In the context of outpatient appointment scheduling, Fries and Marathe (1981) proposed a dynamic programming method to determine the number of patients to arrive at the beginning of each time block for their appointment scheduling rule. Wang (1999) studied scheduling using non-linear programming for both static and dynamic problems in a clinic. The author assumed that customer service times were independent and identical exponential distributions while customer arrivals were punctual. Denton and Gupta (2003) proposed a stochastic programming model in which appointment times were determined optimally for a fixed appointment sequence.

With respect to MP in surgery scheduling, Hsu, de Matta, and Lee (2003) presented a deterministic two-stage no-wait flow shop model for an ambulatory surgery clinic. The first stage addresses the operating room (OR), surgeons, and scarce resources; the second stage models the post anesthesia unit (PACU). They proposed a heuristic to solve the model with the goal of minimizing the number of PACU nurses and the makespan. Guinet and Chaabane (2003) developed a no-wait flow shop method for inpatient surgery. However, they did not provide any solution method. Ozkarahan (1995) introduced a deterministic mixed integer programming (MIP) model to assign the surgery cases to operating rooms (ORs) with the goal of minimizing the under-time and over-time. In addition, the author developed priority rules to sequence the patients. Sier, Tobin, and Mcgurk (1997) suggested a mixed integer non-liner programming model to assign surgery time blocks to patients. The model considered a penalty function including patient's age and resources such as scarce equipment and ORs. They proposed a simulated annealing approach to solve the model. Pham and Klinkert (2008) proposed a deterministic MIP model based on multi-blocking job shop scheduling problem to minimize criteria such as makespan in surgery-case scheduling. They defined each surgery as a sequence of predetermined jobs. They imposed precedence and priority relations to address the conflict of shared

resources. In addition, they allowed emergency cases by imposing deadline constraints and using job insertion methods. Testi and Tànfani (2009) developed a binary linear programming model to solve the master surgical schedule problem, together with the surgical case assignment problem. The model minimized overall patients' welfare loss calculated based on the waiting time of patients on the waiting list. They incorporated urgency levels for patients and investigated the impact of "what-if" scenarios such as adding ORs and different strategies of assigning additional ORs. Min and Yih (2010) proposed a stochastic programming model for case scheduling problem. They considered OR and surgical intensive care units that address different specialties. However, the model did not consider the intake procedure and other resources such as nurses, surgeons and equipment. They solved the model using the sample average approximation.

Güler (2013) studied appointment scheduling of the assignment of the care providers to outpatient rehabilitation clinics. They proposed a hierarchical goal programing approach in order to develop schedules that take into account care provider preferences and reduction of schedule disruptions. Tang, Yan, and Cao (2014) uses heuristic algorithms to solve outpatient appointment scheduling for two types of patients (i.e., routine, and urgent) considering no-show probability. They did not consider clinical paths of patients.

Zhao and Li (2014) addressed elective surgery scheduling in ambulatory setting and studied three aspects of daily scheduling including: the decision on the number of ORs to be enabled, the assignment of cases to ORs, and the sequencing of surgeries assigned to each OR. They assumed a sequence dependent setup time for surgeries, which further signifies sequencing of the cases in this environment. They used mixed integer nonlinear programing along with constraint programing to solve the scheduling problem. Their work considered deterministic surgery durations and concludes that constraint programing can offer better results than nonlinear programing.

Lamiri et al. (2009) proposed a stochastic programming model for surgery planning in order to minimize elective patients' assignment costs and expected overtime costs. They considered a mix of elective and emergency patients and developed an almost-exact Monte Carlo simulation method. They compared different heuristic and metaheuristic approaches (such as simulated annealing and tabu search) with their method. They reported that although their method outperformed the heuristic and metaheuristics methods for small to medium sized test problems, the computation time was significantly higher. In addition, they stated that tabu search was the best among the heuristic and metaheuristics approaches. For large problems, tabu search provided solutions better than those provided by the almost-exact method in a reasonable amount of time.

Simulation has also been used by researchers to address the appointment scheduling problem. Contrary to the analytical methods, simulation can easily accommodate the environmental factors and system parameters such as patient priorities, multi-stage facilities, and servers with different service time distributions. However, simulation models do not include any optimization strategies, and often authors restrict their analysis to the evaluation of pre-specified configurations and manually run the simulations. Since there are several instances of simulation studies in the literature, we review those that are most relevant to this article.

Dexter, Macario, Traub, Hopwood, and Lubarsky (1999) developed a method to assign the block time to surgeons and schedule patients to improve the utilization of ORs using simulation. Marcon and Dexter (2006) analyzed the impact of different sequencing rules on OR utilization and workload of the post anesthesia care unit. Tyler, Pasquariello, and Chen (2003) used a simulation model for an OR to improve its utilization. They also studied how other factors such as the average patient waiting time and variability of surgery duration affect the OR utilization. Robinson and Chen (2003) studied outpatient appointment scheduling using simulation-based techniques. They suggested that while the optimal appointment intervals present a dome pattern, a setting with equal intervals for intermediate appointments might result in better performance for some systems. M'Hallah and Al-Roomi (2014) studied scheduling of elective surgery cases in an OR department which aimed at reduction of ORs overutilization and minimizing completion time. They used a simulation model to evaluate the proposed schedules. They proposed multiple heuristics to achieve better schedules within multiple scenarios.

Although many studies used simulation to solve outpatient appointment scheduling, only few articles utilized simulation-based optimization methods to address the problem. Simulation-based optimization enjoys the flexibility of simulation in modeling complex systems, while systematically seeks optimal solutions through its optimization component. In this domain, Denton, Rahman, Nelson, and Bailey (2006) studied an endoscopy suit, which included two types of patients, and one surgeon type. They used simulated annealing as the optimization tool to schedule the start time of surgery cases to minimize the overtime and patient waiting time. Klassen and Yoogalingam (2009) examined a single stage outpatient clinic and used OptQuest to decide on the arrival time of patients. OptQuest is a simulation-based optimization package that accompanies several discrete event simulation tools. They studied dome patterns in appointment scheduling and suggested that practitioners could employ a "plateau-dome" type rule in many different environments. Chow, Puterman, Salehirad, Huang, and Atkins (2011) use simulation and mathematical programming to propose surgical schedule to reduce surgical ward congestions.

Gul et al. (2011) considered an outpatient surgical suite, and investigated the impact of several sequencing and scheduling heuristics on competing performance criteria. They developed a simulation model which incorporated a bi-criteria genetic algorithm. They demonstrated the impact of different surgery schedules on the competing objectives of the mean patient waiting time and amount of overtime of the outpatient surgical suite. They indicated that the arrival time schedules substantially influenced the expected overtime and patient waiting time, while surgery allocation and sequencing heuristics had a smaller effect.

Ewen and Mönch (2014) studied scheduling of surgeries in an eye clinic and developed a multi-objective scheduling method in order to minimize the patient wait time, and to maximize the utilization of the ORs. They used minimizing the weighted sum of overtime and idleness of ORs as a surrogate objective for maximizing OR utilization. They used a simulation-based optimization method by employing NSGA-II and discrete event simulation. They studied the impact of stochastic arrival patterns of patients and reported outperforming results gained by their proposed method compared with schedules developed by dispatching rules and ones manually created by medical staff.

More recently, Saremi, Jula, ElMekkawy, and Wang (2013) proposed a novel simulation-based optimization method by incorporating mathematical programming to schedule day surgeries. Building on the insights by Gul et al. (2011) that merely combining simulation and heuristics may not necessarily provide the most promising results, they reported that incorporation of mathematical programming significantly improves the performance of simulation-based optimization methods. However, similar to several other articles tackling multiobjective scheduling of patients, Saremi et al. (2013) and Gul et al. (2011) did not offer a Pareto front of solutions. Rather, they only provided a single solution for each scenario. In addition, their methodology only

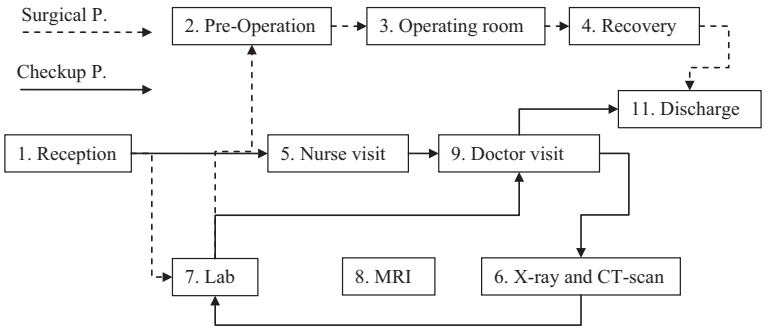

Fig. 1. Layout of a clinic depicting the service sequences of check-up patients and surgical patients.

addresses homogenous sequence of services and does not support multiple sequences.

Lee and Yih (2014) proposed a simulation-based optimization method to schedule surgery cases in surgical suits. They developed a multiobjective method to minimize patients' wait time, idealness of resources, and completion time of the schedules. Their scheduling process included two stages: sequencing of surgeries and determining the exact timing of each procedure. They used a genetic algorithm to fulfill the sequencing while the further scheduling of was done by a "decision-heuristic." They reported that the result of their method outperformed simple scheduling rules when used in regional hospitals.

Our review of the literature concludes with the following remarks:

1. Current optimization methods in outpatient appointment scheduling, although can provide optimal or near optimal solutions, mainly focus on elements/stages of the facility, or simplified analytical models of the system. On the other hand, simulation methods can address many complexities in large systems. They are, however, time-consuming and often do not deliver a competitive optimization strategy. Therefore, we observed a lack of methods in existing literature that provide optimal or near-optimal solutions while covering complexities of healthcare facilities.
2. Several performance criteria such as patient waiting time, clinic overtime, etc. have been studied in the literature, and several studies addressed problems with multiple criteria. However, most of these studies considered a weighted sum of objectives to generate a single function optimization problem. Therefore, a gap still exists in the literature for methods that provide Pareto fronts for the appointment scheduling problems.
3. To our best knowledge, Pham and Klinkert (2008) is the only article that proposed a method that can address patients with different service sequences. However, their method only considers the deterministic version of the problem in OR departments. Thus, we believe that the existing literature lacks methods that can address multi-stage facilities serving patients with different service sequences and stochastic service times.

To address these gaps we propose a multiobjective simulation-based tabu search method enhanced by MP model. Our work can be differentiated from the previous works in following directions: first, to our knowledge this work is the first attempt to address appointment scheduling of patients with stochastic service times, and heterogeneous service sequence in a multi-stage facility in which patients may revisit a stage. Second, our method takes advantage of the flexibility of simulation to model different complexities of systems. Our approach integrates MP and tabu search to find optimal or near optimal solutions. Third, this work offers the Pareto (near) optimal set of schedules, which shows the tradeoffs between the factors that influence patients and providers. Finally, this article includes a case study of an OR department of a major Canadian hospital. We apply the proposed methodology to the actual data and compare the result with the outcome of actual surgery schedules. In the understudy OR department each patient might take a different route which changes according to required type of procedures, and the availability of compatible ORs.

3. Problem description

In this work, we address facilities in which patients can have different sequences of services. A good example for this kind of facility is a private clinic that offers a large variety of outpatient services, from checkups to small surgeries. We consider the appointment scheduling of patients of different types with stochastic service times in each stage where each type of patients may not only require different service times, but also has a specific sequence of services (including the possible revisits to a stage). Here, appointment scheduling refers to the determination of the arrival time of each patient at the clinic in order to minimize the average waiting time of all patients, and the completion time of the facility. The availability and compatibility of resources such as doctors and nurses for each stage are considered.

For instance, consider a clinic that includes a surgical suite, a diagnostics section, and doctor consultation sections. Fig. 1 shows the layout of the clinic and the route that a checkup patient and a surgical patient may take. It is possible that some patients need to revisit a stage based on their type. Fig. 1 shows that the checkup patient first visits the doctor, goes to X-ray and Lab, and then returns to the doctor stage. Here OR stage includes multiple staffed ORs with the service time depending on the patient's type.

A predetermined number of patients of different types are considered to be scheduled in the horizon of one day. Parameters such as distribution of service time for each patient type, capacity of each stage, and sequence of services are given. Patients arrive according to the schedule, and we assume no tardiness in arrivals and no no-shows in the system. The stages in the clinic (except for the reception stage) work according to the first-come-first-served rule. The reception stage admits the patients according to schedule.

4. Methodology

Our approach, multi-agent tabu search (MATS), employs an MP model, and a simulation model along with the tabu search to generate Pareto (near) optimal schedules that minimize waiting time of patients and completion time of the clinic. The MP model provides MATS with promising initial solutions through solving

deterministic version of the problem. Tabu search then improves the initial solutions by searching for optimal schedules by running a number of agents in parallel. The agents seek the non-dominated solutions of the problem and share information with each other to improve the search performance of the algorithm. Simulation model enables MATS to evaluate the schedules by considering stochastic nature of services of the clinic and applying several resource and operational constraints. Finally, MATS presents a non-dominated set of solutions which contain most promising appointment schedules.

4.1. Mathematical programming model

In the proposed method, MP model provides the MATS with promising initial solutions. The MP model is an extension of the model proposed by Jula and Leachman (2010), which is enhanced and adapted for appointment scheduling in healthcare. The notation of the model is presented as follows:

Notation

t	discrete time index, $t = 1,\ldots,T$, where T is the time horizon and number of time grids in each day
j	stage index, $j = 1,\ldots,k, k+1,\ldots,M$, where M is the number of stages in the department. We assume that stage k is the doctor visit stage and the stage $(k+1)$ is dedicated to patients' revisit. Stage M is the discharge stage
p	patient type index, $p = 1,\ldots,P$; where P is the number of patient types
$[j]_{Bp}$	indicates the stage located before position of stage j, in the ordered set of B, for patient type p

Parameters

N_p	the number of patients of type p
$S_{j,p}$	service time of patient type p in stage j
$I_{j,p}$	the initial number of patients of type p in the line, waiting to be served at stage j
R_j	the number of available servers or operators in stage j at the beginning of the scheduling horizon
γ_p	the cost of waiting of a patient of type p for a single time period
H	an arbitrarily large number
B_p	the ordered set of stages that patient type p should follow
α	cost coefficient for waiting of a patient per time period
β	cost coefficient of operating the department per time period

Variables

$x_{j,t,p}$	the number of patients of type p at stage j to start being processed at time t
$Q_{j,t,p}$	the number of patients type p who are waiting to be served at stage j at time t
$X_{j,t,p}$	the cumulative number of patients of type p at stage j has been started being processed by time t
$r_{j,t}$	the number of available idle resources at stage j at time t; each stage has its dedicated resources
m	the last time block, in which all patients have been discharged (makespan of the schedule)
y_t	a binary (1 or 0) variable which indicates if there is any discharge at time t

The MP model is expressed as follows:

1. Objective functions:

$$\text{Minimizing} \quad \alpha \sum_j \sum_t \sum_p \gamma_p . Q_{j,t,p} + \beta m. \tag{1}$$

2. Queue balance constraints:

$$Q_{j,t,p} = I_{j,p} - X_{j,t,p} + X_{[j]_{Bp},t-s_{[j]_{Bp},p},p}, \quad \forall j \in B_p, t, p. \tag{2}$$

3. Cumulative variables:

$$X_{j,t,p} = \sum_{\tau=1}^{t} x_{j,\tau,p} \quad \forall j \in B_p, t, p. \tag{3}$$

4. Capacity constraints:

$$r_{j,t} = R_j - \sum_{(p|j\in B_p)} X_{j,t,p} + \sum_{(p|j\in B_p)} X_{j,t-s_{j,p},p} \quad \forall j \notin \{k, k+1\}, t, \tag{4}$$

$$r_{k,t} = R_k - \sum_p X_{k,t,p} + \sum_p X_{k,t-s_{k,p},p} - \sum_p X_{k+1,t,p} + \sum_p X_{k+1,t-s_{k+1,p},p} \quad \forall t. \tag{5}$$

5. Number of patients which have to be served:

$$X_{M,T-s_{M,p},p} = \sum_j I_{j,p} \quad \forall p. \tag{6}$$

6. The makespan indicator constraint:

$$y_t \cdot H \geqslant \sum_p \sum_j x_{j,t,p} \quad \forall t, \tag{7}$$

$$m \geqslant t.y_t, \quad \forall t. \tag{8}$$

$x_{j,t,p}, Q_{j,t,p}, X_{j,t,p}, r_{j,t} \geqslant 0 \ \forall j, t, p. \ X_{1,t,p} \ \forall t, p$ is an integer variable; y_t is a binary variable for t.

7. Initial conditions:
The values for $x_{j,t,p}, I_{j,p}, X_{j,t,p}, r_{j,t}$ should be pre-specified for $t < 0$ if applicable.

To the best of our knowledge, Pham and Klinkert (2008) is the only article that provided an MP model capable of addressing scheduling of patients with heterogeneous service sequence in a multi-stage surgery department. While they only considered the deterministic version of the problem, they focused on minimizing the makespan and scheduling of the patients as soon as possible. Their experiments includes only up to 26 patients served by six ORs. Our experiments suggest that this policy may cause significant patient waiting time.

Our proposed model is able to model the flow of patients in the stages of the facility as well as revisited stages. In addition, we consider minimization of bi-objectives of patients waiting time and completion time while considering the resources availability and compatibility. Our experiments show that our model can solve large problems in a reasonable amount of time.

4.2. Simulation model

We developed a discrete event simulation model of the described facility using the Arena™ 12 software. The model includes all stages shown in Fig. 1. The commonly used lognormal distribution is adopted to model the service time distribution for each stage (e.g., see Zhou & Dexter, 1998). The mean and variance of the service time is determined based on the specified values in each test problem, to be discussed in Section 5. In addition, the number of resources at each stage is determined according to the test problem parameters such as the number of patients, the

number of ORs, and so on. For OR stages, we assume that the number of surgeons is equal to the number of ORs, and the ORs are considered staffed. The allocation of patients to the servers is done by the algorithm with the goal of finding a better front (set of solutions). More specifically, the algorithm tries to find assignment of patients to the available servers and resources at each stage in a way that results in minimal completion time and patient waiting time. It is embedded in the logic of the simulation model where within the queues of each stage, the priority is given to the patients who arrive earlier. In other words, the first available resource is assigned based on first-come-first-served rule in the queues. Therefore, the server and resource allocation has direct correlation with the patients' appointments sequencing and timing.

The simulation model falls in the category of terminating simulation models (Banks, Carson, Nelson, & Nicol, 2005). That is, in our simulation model, a predetermined number of patients are served in the scheduling horizon of one day. The patients arrive according to the schedule, and we assume no tardiness in arrivals and no no-shows in the system. Patients proceed through the facility according to their service sequence.

We used the simulation model to estimate patients' waiting time and completion time for each proposed schedule. Also, the simulation model is used to as a platform to compare the performance of MATS with NSGA-II.

4.3. Multi agent tabu search (MATS)

Most multiobjective tabu search methods consider multiple solutions that are simultaneously improved towards the Pareto optimal front. Typically, these methods apply two approaches regarding the objective function handling. The first approach considers a weighted sum of the objectives as a new objective function. Therefore, any single objective optimization method can be used to solve the problem. The weights are usually pre-set and the result of this method is a single solution rather than a Pareto front. The second approach deals directly with finding the Pareto frontier solutions. This approach has been applied in different ways to enhance the single-objective method.

For instance, Hansen (1997) used a modification of weighted sum method in which multiple weighted sum objectives are improved simultaneously considering different sets of weights for objectives. Caballero, Gandibleux, and Molina (2004) developed a two-phase tabu search based algorithm: the first phase involved a tabu search method to generate a non-dominated solution set; the second phase consisted of an intensification method using path-relinking strategies. Jaeggi, Parks, Kipouros, and Clarkson (2008) developed a tabu search method using Hooke–Jeeves direct search methods. They suggested that NSGA-II might be a better approach for problems with a small number of variables. However, they noted that multiobjective tabu search is a better approach for large and highly constrained problems.

In this article, we propose a new multi-criteria multi-agent tabu search algorithm (MATS) to deliver the Pareto (near) optimal frontier for the appointment scheduling problem. MATS algorithm includes a number of agents that attempt to find members of the non-dominated solution set (also, called approximation set). Here, we incorporate a stochastic simulation model with the tabu search to solve a highly constrained problem (including time and resource constraints). The time and resource constraints implemented using simulation and MP models. For instance, availability of resources in each stage is managed by variables local to simulation model. However, this policy requires passing of many parameters to the model. Simulation model, for example, requires the initial values of resources that are provided by MATS algorithm. In order to reduce the number of function evaluations, we develop a deterministic scheduling module (DSM) which estimates the waiting time and completion time of a schedule based on the mean of service time, and selects a number of schedules with the largest fitness value to be evaluated by the simulation model.

Moreover, agents work in parallel and share information with other agents regularly to improve algorithm performance. Each agent functions as a standalone tabu search that seeks optimal solutions for the problem within each iteration. MATS updates the information among the agents after every iteration. Each iteration of MATS includes only a single iteration of an agent.

A solution consists an array of size n, where n is the number of patients. This array represents the time block at which each patient arrives at the first stage. Fig. 2 depicts the steps involved in MATS, and each step is described below.

4.3.1. Generating initial solutions using the MP model

In order to initialize the algorithm we need to provide each agent with an initial solution. A number of initial solutions are determined using MP. The rest of initial solutions are specified randomly by assigning an arbitrary time block to each patient.

4.3.2. Fitness computation and frontier points identification

In MATS, we determine the fitness value (G score) of a given set of solutions following the Schaumann, Balling, and Day (1998) approach.

$$G_i = \left[1 - \max_{1\leqslant j\leqslant J, j\neq i}\left(\min_{1\leqslant k\leqslant m}\left(f^i_{s1} - f^j_{s1}, \ldots, f^i_{sk} - f^j_{sk}, \ldots, f^i_{sm} - f^j_{sm}\right)\right)\right]^l, \tag{9}$$

where, G_i is the fitness value of solution i; J is the number of objectives and m is the number of solutions in the set; i and j are two solutions in a given set of solutions. f^i_{sk} is the scaled kth objective value of the ith solution. The max operator is over all solutions except solution i. The min operator includes all the objectives. The f^i_{sk} is scaled in range $[0,1]$, hence all objective values are in the same range and comparable. f^i_{sk} is scaled using Eq. (10).

$$f^i_{sk} = \frac{f^i_k - f^{\min}_k}{f^{\max}_k - f^{\min}_k}, \tag{10}$$

where, f^i_k is the original value of the kth objective and $f^{\max}_k$ and $f^{\min}_k$ are the maximum and minimum values among original values of the kth objective of all solutions, respectively. The non-dominated solution set is determined based on the G score of all solutions.

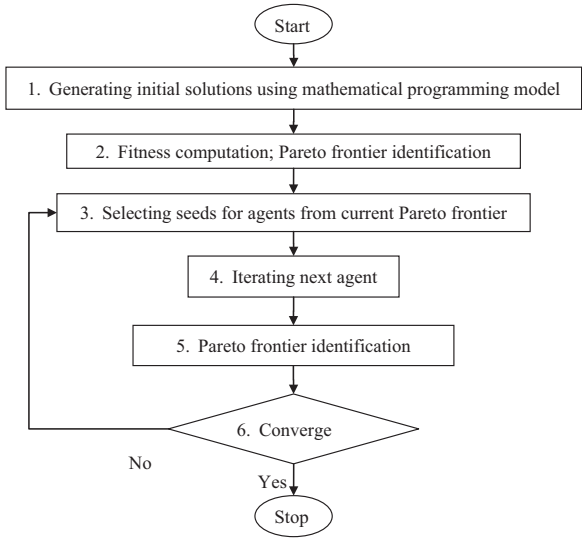

Fig. 2. Steps of MATS.

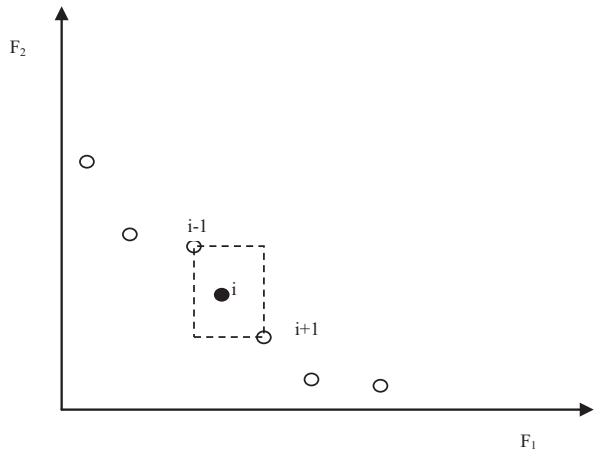

Fig. 3. The crowding distance of point *i* is the side length of the cuboid surrounding the point.

The solutions with the score greater than or equal to one are identified as non-dominated solutions.

For instance, consider five solutions with 2 objectives of f_1 and f_2: $a(0.5,2)$, $b(1,1)$, $c(4,0.5)$, $d(3,1.5)$, and $e(2,3)$. The scaled objectives of these points over the first objective are as follows: $f^1_{sa}=0$, $f^1_{sb}=0.1428$, $f^1_{sc}=1$, $f^1_{sd}=0.7142$, $f^1_{se}=0.4285$, $f^2_{sa}=0.6$, $f^2_{sb}=0.2$, $f^2_{sc}=0$, $f^2_{sd}=0.4$, $f^2_{se}=1$. For sake of brevity, only the calculation relevant to point a is presented. The term $\max^{J}_{j=1,j\neq i}\left(\min^{m}_{k=1}\left(f^i_{s1}-f^j_{s1},\ldots,f^i_{sk}-f^j_{sk},\ldots,f^i_{sm}-f^j_{sm}\right)\right)$ for $i=a$ is equal to following term by replacing the values: $\max\{\min(-0.1428,0.4),\min(-1,0.6),\min(-0.7142,0.2),\min(-0.4285,-0.4)\}$ which ultimately results in $\max(-0.1428,-1,-0.7142,-0.4285)$ and hence the G_a will be equal to 1.1428.

4.3.3. Selecting seeds for agents from the non-dominated solutions set

In order to achieve the best performance of the algorithm, the agents' seeds are updated every time the Pareto front changes. We expect that the better are the agents' seeds, the better results can be expected from the optimization. MATS considers the last iterated agent and updates the next agents' seeds with the solutions that have the highest fitness scores. This treatment may cause the search to trap in a local minimum. To remedy this issue we used the crowding distance density estimate by Deb, Agrawal, Pratap, and Meyarivan (2000) to improve the diversity of agents' seeds. In order to assign the agents' seed, we select the non-dominated solutions with the highest value of crowding distance estimator.

4.3.4. Iterating next agent

The next agent performs an iteration, which includes neighborhood generation, fitness evaluation and, selecting the next seed considering the tabu list. This process is described in detail in Section 4.4.

4.3.5. Pareto frontier identification

This step includes the identification of Pareto front based on the new G scores, which are obtained by including the selected promising solutions from the agents' iteration in the previous step. When an agent iterates, the G scores of the promising solutions within the neighborhood of the agent are evaluated and compared with the best solutions known in previous iterations. If any of the solutions within the neighborhood has a high fitness value it will be added to the long term memory. The solutions with the largest G score are identified as non-dominated solutions.

4.3.6. Convergence condition check

Several convergence conditions can be applied to the proposed method. MATS considers two criteria for algorithm termination, (a) the number of stalled iterations, and (b) the number of function evaluations. If the Pareto front does not improve from the previous iteration, the current iteration is considered as a stalled iteration. MATS converges if a specified number of stalled iterations happen consecutively. In addition, MATS terminates after performing a specified number of function evaluations (solutions evaluated by simulation).

In order to improve the performance, tabu search keeps record of local information by means of different types of memory structures. This local information may include parts of the solutions, or other attributes of the solutions. We use a short-term memory to address the tabu lists of local searches of the algorithm. After each move, the attribute of the move is recorded to avoid cycling in the algorithm. In the proposed method, each local search has a tabu list. We select the tabu tenure of 30 iterations. Based on the swap and insertion, we construct the tabu lists, which are shared by all agents. The lists contain the history of recent moves. For example, when swapping is applied, a list of moves is recorded. This list prevents any reverse moves.

The long term memory is used to record all best solutions achieved in all iterations. MATS records all the solutions with G score greater than $G_0 \in [0,1]$. Long term memory includes all non-dominated solutions at each iteration of the algorithm. In addition, the recorded solutions are used in the calculation of G score for future solutions. The G score of each solution is updated in every iteration. Long term memory is also used to achieve intensification and diversification in the algorithm.

As an agent performs the search, a number of best solutions are evaluated using simulation model. These solutions are then evaluated and ranked based on their G scores. At the end of each iteration, the seeds of agents are updated and replaced with the best solutions recorded in the long term memory. The selection procedure includes two steps: first, all the solutions with the greatest G score are selected. Then, the solutions with the largest crowding distance are selected as new seeds for the agents. This policy intensifies the search in the most promising regions. Contrary to the other multi objective tabu search methods which separately improve multiple solutions in parallel, our method uses all the information obtained collectively by all agents. Moreover, each agent uses the information which gathered by all agents.

Long term memory usually used in diversification policies of tabu search methods to lead the search into the unexplored regions of search space. In single objective optimization problems, a common strategy is penalizing objective function based on the frequency of occurrence of attributes. In this work, we use the crowding distance density estimator introduced by Deb et al. (2000) to maintain diversity of our search method. This method estimates the density of the solutions around a specific point by taking the average distance between this point and its two neighboring points on each side in the performance space. The distance quantity represents the size of the largest cuboid that encompasses the point excluding any other point from the Pareto front. Fig. 3 shows the cuboid which determines the crowding distance for point i in its front. F_1 and F_2 represent the first and second objective. In addition to guiding the search to unexplored regions, the crowding distance helps to increase the range of non-dominated solution set by assigning large values to extreme solutions for each objective function.

MATS applies a hash function in order to speed up the search and identify the solutions. Hash functions commonly refer to any algorithm or mathematical function that converts a large amount of data into an integer value. Hash functions are usually used to speed up table lookups or to find duplicated records.

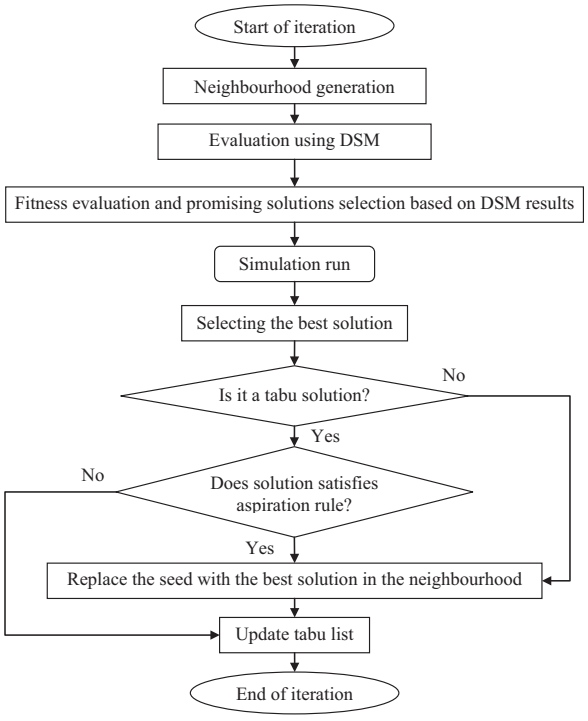

Fig. 4. Flowchart of an agent's iteration. This flowchart depicts the step 4 of MATS depicted in Fig. 2.

We use the hash function here to map an integer value to each solution which enables us to speed up search in the long term memory. Furthermore, it prevents the algorithm from recording duplicate solutions in the long term memory and agents' seed. In MATS, Cyclic Redundancy Check (CRC32) algorithm has been used as the hash function. Reader can refer to Stigge, Plotz, Muller, and Redlich (2006) for more information on its theory and implementation.

4.4. Tabu search agent

In this section, we describe the step 4 of the MATS algorithm. This step includes the steps of an agent's iteration. In the proposed algorithm, agents operate according to the tabu search algorithm. Generally, tabu search iteratively generates the next solution j from the current solution i through specified steps. A neighborhood is defined for each current solution, $N(i)$. The next solution is obtained by searching around $N(i)$ using neighborhood search methods.

Tabu search allows non-improving moves. That is, even if the best solution found in the current neighborhood is a non-improving one compared to the best-known solution, the non-improving solution will be used in the next iteration. We define a move as replacing the seed solution of the tabu search with a new solution. The components of the tabu search are as follows:

4.4.1. Local searches

Swapping: let i and j be two positions in a random sequence s. By performing a swap-move iteratively, a neighborhood of s is obtained by interchanging the patients in positions i and j. In the proposed method, swapping is applied to the second part of the solution, which concerns time blocks.

Insertion: let i and j be two positions in a random sequence s. A neighborhood of s is obtained by inserting the patients assigned to position i to position j, pushing the cells between these positions backward (forward), including the patients of position j, if j is greater (less) than i. This change of positions is performed on the first part of a solution (patients' time blocks). The policy of swap on the second part of solution and Insert on the first part reduces the chance of cycling and trapping in local minima.

4.4.2. Deterministic scheduling module (DSM)

In order to reduce the number of function evaluations (evaluating solutions by the simulation model), a deterministic heuristic has been developed. DSM calculates the average waiting time of patients and completion time of each schedule based on the mean of service times. This component enables us to screen solutions before being evaluated by the simulation model. DSM calculates the discharge time of each patient based on the patient service sequence and mean of service times. The completion time of the schedule is the maximum discharge time of all patients. The waiting time of each patient is determined under the assumption that the waiting time is the total time that a patient spends in the facility subtracted by the sum of service times at different stages. DSM ranks the evaluated solutions according the G score calculated based on deterministic evaluations. A number of solutions with the largest G score will be evaluated using the simulation model. Fig. 4 illustrates the flowchart of an agent's iteration which is a zoom-in of step 4 of MATS algorithm.

The neighborhood is built based on the current solution using the swap and insertion local search. A random number of solutions in each neighborhood are evaluated by using the DSM. The solutions are sorted and ranked based on the values assigned to them through the deterministic evaluation. The agent algorithm selects the promising solution based on the G score of all solutions calculated using the approximate average waiting time and completion time obtained from DSM. The selected promising solutions are then evaluated using the simulation model. The simulation model determines the average waiting time and completion time through multiple replications of the simulation. We used 30 replications of simulation run for each setting in our experiments.

The next step includes the fitness evaluation of the selected solutions with respect to all solutions with high fitness values obtained in previous iterations. In order to do so we have to add the set of new solutions which are going to be evaluated to the list of solutions with the highest fitness values (non-dominated solution set), and evaluate the G score of all of the solutions based on this combined set.

5. Performance study of MATS

5.1. Performance measures

To gauge the performance of the proposed MATS for multiobjective optimization, we compared the algorithm with the well-known NSGA-II method based on a number of performance measures.

Typically, to study the performance of a multiobjective optimization algorithm, two aspects are examined: effectiveness (the quality of outcome) and efficiency (the required amount of computational resources to generate such an outcome). For effectiveness measurement, various performance measures have been defined in the literature. Addressing the efficiency of an algorithm often includes considering a fixed number of function evaluations or a certain amount of computational time. Interested readers refer to Zitzler, Thiele, Laumanns, Fonseca, and Da Fonseca (2003) for a review of performance metrics. Here, we selected three criteria for evaluating performances: the hypervolume indicator, spacing indicator, and computation time. We use the hypervolume indicator to measure the closeness of the non-dominated solutions set to the Pareto-optimal front. The spacing indicator has also been

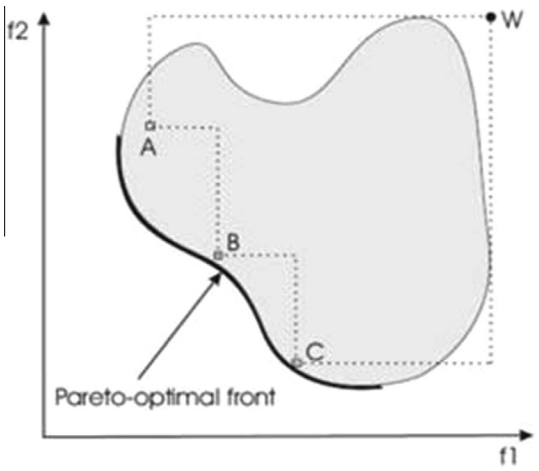

Fig. 5. Illustration of hypercube indicator, Durillo, Nebro, Luna, Dorronsoro, and Alba (2006).

utilized to examine the spread of non-dominated solutions. Furthermore, to compare the efficiency of the algorithm, we consider a limited number of function evaluation (500 function evaluations), and compare the computation time of algorithms.

5.1.1. Hypervolume indicator (HV)

One of the metrics that is employed by many researchers is *hypervolume*, first proposed by Zitzler and Thiele (1998). Hypervolume delivers a single scalar for the closeness of the non-dominated solutions to the Pareto optimal front. In addition, it can be used to compare algorithms in problems in which the Pareto optimal front is unknown.

For example, The HV indicator calculates the volume confined by the non-dominated points and a reference point as shown by the shaded area in Fig. 5. The reference point can be simply determined as a vector of worst possible objective values. The method that results in larger HV indicator values is more desirable. Fig. 5 presents the calculation of the HV indicator for a bi-objective problem where the objectives are to be minimized.

5.1.2. Spacing indicator

The spacing indicator proposed by Schott (1995) measures how evenly the points in non-dominated solutions set are distributed in the objective space. The indicator is the standard deviation of the distance of each point to its closest neighbor, Eq. (11).

$$\sigma = \sqrt{\frac{\sum_i (d_i - d)^2}{(n-1)}}, \tag{11}$$

where, $d_i = \min_j \left(\left| f_1^i - f_1^j \right| + \left| f_2^i - f_2^j \right| \right)$, $i,j = 1,\ldots,n$, and $\bar{d}$ is the mean of all d_i, and n is the number of solutions in the non-dominated solution set. The less value of the spacing indicator, the smoother the points are distributed within the objective space. The smooth spread of points in Pareto front is preferred because it suggests that the method is capable of finding non-dominated solutions in all areas of objective space.

5.1.3. Tests and results

This work uses GAMS™ to develop the mathematical programming model and Arena™ 12 for simulation. The CPLEX solver has been used to solve the MP model. The multiobjective tabu search has been coded in Visual C# 2005. We used the Arena object model component to establish connection between MATS and simulation model. We used a PC with 2.53 GHz Intel® Core 2 Duo CPU with 3 GB of RAM to run the experiments.

In order to evaluate the performance of the proposed method, we selected several important factors based on our preliminary analysis. These factors include the number of patients, number of patient types, and coefficient of variation of service times. We then designed a set of experiments to analyze the effect of selected factors on the performance of algorithms. Based on our case study and our observations in other facilities, we defined a set of test problems with three levels (10,20,40) for patients, and three levels (4,6,10) for patient types to represent different size facilities. Each patient type follows a different service sequence as depicted in Table 1.

Table 1 shows the specifications of patient types. For each patient type, the mean of service time at each stage is derived randomly from a uniform distribution between 15 and 120 min (i.e., Uniform [15,120]), as listed in Table 1. The values then are rounded to the closest multiplier of the length of a time block of 15 min. Moreover, we considered the lognormal distribution as well as two different levels of coefficient of variations (CV) of 0.1 and 0.4 for each service time. We used the lognormal distribution since it is commonly used in the literature to represent duration of services in the healthcare (e.g., see Zhou & Dexter, 1998). The mean of the distributions was assumed based on patient types.

In this paper, the algorithm chosen for studying the performance of the proposed method is the Non-Dominated Sorting Genetic Algorithm (NSGA-II), which is a widely used multiobjective evolutionary algorithm and publicly available (Deb et al., 2000). NSGA-II classifies the individuals into several layers by applying a non-dominated sorting method and a crowding distance operator. It incorporates elitism selection strategies. We used the implementation of NSGA-II in Matlab™ optimization toolbox. NSGA-II was run with a population size of 50 and 500 function evaluations. For the rest of parameters, we used default settings. The crossover rate of 0.8 (the single point crossover has been adopted), the

Table 1
Specification of patient types.

Patient type	Service sequence	Mean of service time of stages (min)	Description
1	1, 5, 9	15, 15, 105	Patients who need a doctor visit
2	1, 7	15, 45	Patients who need to do a lab test
3	1, 8	15, 105	Patients who need to do an MRI
4	1, 6	15, 120	Patients who need to do a X-ray/CT-scan
5	1, 2, 3, 4	15, 105, 60, 120	Patients with surgical procedures
6	1, 7, 2, 3, 4	15, 75, 15, 120, 45	Patients who need to do a lab before surgical procedures
7	1, 5, 9, 7	15, 75, 120, 90	Patients who need to do a lab after doctor visit
8	1, 5, 9, 8	15, 15, 30, 120	Patients who need to do an MRI after doctor visit
9	1, 5, 9, 6	15, 30, 75, 105	Patients who need to do a X-ray/CT-scan after doctor visit
10	1, 5, 9, 6, 10	15, 105, 45, 90, 105	Patients who need to do a X-ray/CT-scan after doctor visit and return for consulting the results with doctor

Table 2
Comparison of MTAS and NSGA-II in terms of quality and computational time.

Test problem	# Of patients	# Of patient types	CV	NSGA-II			MATS		
				HV	Spread	Time (s)	HV	Spread	Time (s)
P10T4C0.1	10 patients	4 types	0.10	79.49	0.25	353.23	82.20	0.09	196.05
P10T4C0.4			0.40	78.23	0.21	370.98	80.51	0.14	192.66
P10T6C0.1		6 types	0.10	80.39	0.16	507.21	89.48	0.14	182.50
P10T6C0.4			0.40	77.05	0.24	395.63	85.27	0.16	179.95
P10T10C0.1		10 types	0.10	67.83	0.33	366.36	74.65	0.26	224.30
P10T10C0.4			0.40	63.33	0.21	386.05	68.23	0.17	206.40
P20T4C0.1	20 patients	4 types	0.10	73.23	0.13	383.14	87.05	0.08	203.80
P20T4C0.4			0.40	71.04	0.15	357.47	84.43	0.13	204.22
P20T6C0.1		6 types	0.10	66.53	0.25	396.28	77.65	0.12	225.53
P20T6C0.4			0.40	61.88	0.19	398.58	70.92	0.13	209.39
P20T10C0.1		10 types	0.10	60.07	0.30	488.41	72.45	0.21	236.63
P20T10C0.4			0.40	54.92	0.18	401.63	62.84	0.12	236.50
P40T4C0.1	40 patients	4 types	0.10	64.27	0.11	397.23	78.32	0.07	232.30
P40T4C0.4			0.40	62.38	0.12	451.53	74.14	0.08	232.50
P40T6C0.1		6 types	0.10	58.49	0.11	429.84	71.57	0.07	230.81
P40T6C0.4			0.40	55.74	0.11	410.19	66.74	0.09	199.83
P40T10C0.1		10 types	0.10	52.58	0.15	396.30	66.71	0.13	240.30
P40T10C0.4			0.40	48.70	0.13	417.53	58.90	0.08	258.68

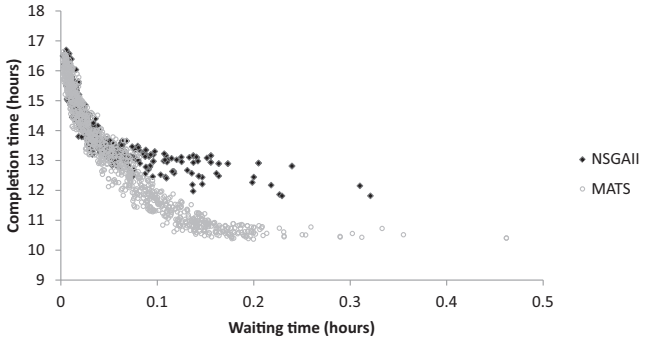

Fig. 6. Comparison of Pareto sets of MATS and NSGA-II for 30 runs.

mutation rate of 0.01 (order changing mutation has been adopted), and tournament selection have been used in experiments.

Table 2 presents the result of experiments on the test problems, and compares MATS with NSGA-II based on the three criteria of HV, spacing indicator, and computation time. The results are based on the average value of indicators over 30 runs for each method.

Studying the effectiveness of algorithms, the results in Table 2 show that MATS yields greater values of the HV indicator. It suggests that solutions offered by MATS are closer to the Pareto-optimal front than NSGA-II. The spacing indicator results suggest that MATS presents more evenly distributed non-dominated solutions than NSGA-II as well. Considering the efficiency of the algorithms, MATS needs less time to complete 500 function evaluations. In summary, our experiments show that MATS consistently presents better results than NSGA-II in terms of both efficiency and effectiveness.

Differences in the performance of the two methods can also be visually observed. The performance of the two algorithms significantly differs from each other when it comes to completion time as the waiting time increases. For example, Fig. 6 shows the non-dominated solution set of MATS and NSGA-II for the test problem P40T6C0.4 with 40 patients, six patient types, and CV of 0.4. The figure shows the non-dominated solution sets for 30 runs of MATS and NSGA-II. The figure suggests that MATS and NSGA-II can achieve comparable results in the solutions with greater completion time of the facility. However, in the region which addresses smaller completion time with greater waiting times of patients, there is a significant difference between the results of these methods. The results suggest that MATS outperforms the NSGA-II in terms completion time.

Table 3
Specification of test problems.

Test problem	# Of patients	# Of patient types	# Of surgeons
Day 1	39	30	10
Day 2	38	29	10
Day 3	36	29	9
Day 4	48	35	11
Day 5	32	30	10

6. Case study

This section applies our methodology to the data obtained from an OR department of a regional Canadian hospital with around 500 beds. This hospital is the only site in its health authority that offers services in specialties such as thoracic surgery. The OR department of this hospital includes 10 ORs and 20 post anesthesia care unit bays. Annually around 8000 surgeries are performed in the hospital; 20% of them are emergency cases and the rest are elective cases. However, not being a trauma center, most of the emergency cases in this hospital can be performed within 72 h. This time enables the surgical program to schedule the emergency and elective cases. Here we only address scheduling of surgeries including emergency and elective cases that can be scheduled in advance. That is, emergency cases that are not known by the beginning time of the scheduling horizon are out of scope of this study. In addition, patients in the understudy OR department include both inpatient and outpatient cases served in the department. Inpatient patients are treated as the same as the outpatient patients in terms of scheduling.

In this setting, the facility completion time refers to the time when the last served patient leaves the post anesthesia care unit (PACU). This measure reflects the total time that an OR department should be operational. Completion times longer than official operating hours results in overtime of the department. The average patients waiting time reflects the blocking at each stage. For instance, if a patient is waiting to receive a PACU bay, this patient is blocking an OR and delaying the future surgeries of that OR.

The number of patients and patient types are pre-determined in a higher-level planning step according to the available resources. We assume patients' punctual arrivals. Each patient is served by

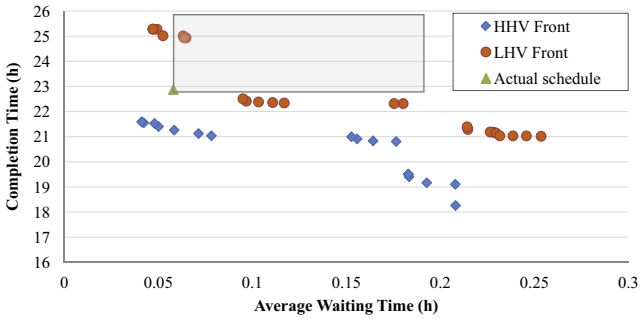

Fig. 7. Sample performance of proposed method versus actual schedule.

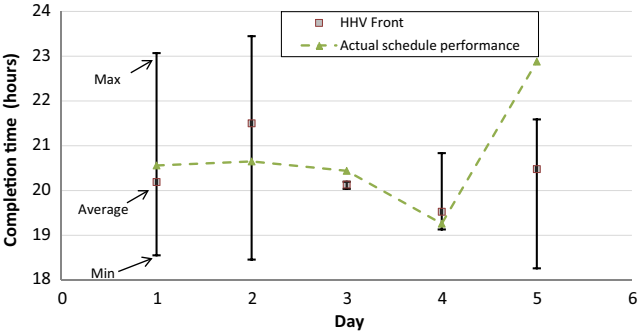

Fig. 8. Comparison of performance of proposed method versus actual schedule in terms of completion time.

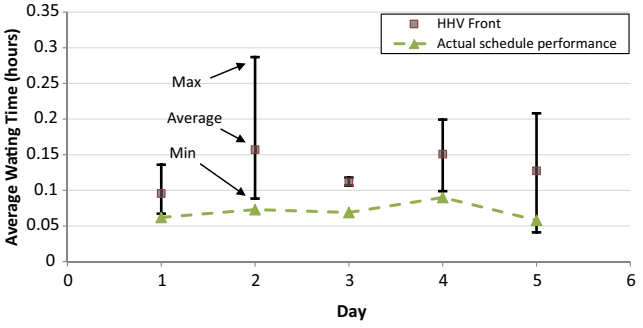

Fig. 9. Comparison of performance of proposed method versus an actual schedule in terms of waiting time.

a specific surgeon. Therefore, it is important that the compatibility of resources and patients be considered.

6.1. Experiments and results

The MP models have been developed using GAMS™, and CPLEX solver has been used to solve the models. The simulation model has been built in Arena™ 12 environment. Microsoft Visual C# was used to code the multi-agent tabu search algorithm. The connection between MATS and the simulation model has been established using Arena object model component. A PC with i7 2.8 GHz Intel® CPU with 8 GB of RAM has been used to run the experiments.

In order to evaluate the performance of the proposed method, a sample of five days of the OR department data are selected. These days have been suggested by the management of surgery unit, because the detailed data were readily available. The data for these five days are then used to construct five test problems. Each test problem is implemented in the simulation model and the parameters of MP models and MATS are set accordingly.

For each test problem, five runs of the methodology are considered. Each run provides a Pareto front. The hypervolume (HV) indicator for each run is then calculated. Based on the hypervolume values of these five runs, the HHV front (the front with the highest HV value) and the LHV front (the front with the lowest HV value) are chosen. For each day, the best and the worst performance of the algorithm are then compared with the actual schedule retrieved from actual data. Table 3 presents the specification of test problems.

6.2. Sample daily performance

Fig. 7 shows the result of the proposed method on day 5 of test problems. Results suggest that our method delivers a few non-dominated schedules (HHV front) compared with actual schedule. It is observed that our method offers schedules that significantly decrease the completion time while slightly raises patients' average waiting time.

Based on the coordinates of the actual schedule, a gray square has been drawn to represent the dominated area. The dominated solutions fall in the square. Most of the solutions from the front with lower HV value fall in the dominated area, which shows the better performance of actual schedule than most solutions in LHV. However, the LHV front still presents few non-dominated solutions.

Fig. 8 compares the performance of the proposed method with performance of actual schedule in terms of completion time. This figure presents the results of HHV front, and actual schedule for a week. For each day, maximum, minimum, and the average value of the set of solutions in the HHV front are presented. The figure suggests our method outperforms the actual schedules in terms of completion time, which is a very important measure in scheduling as it has direct impact on the overtime.

Fig. 9 compares the performance of the proposed method with performance of actual schedule in terms of average waiting time. This figure presents the results of HHV front, and actual schedule. For each set of solutions, maximum, minimum, and average value of HHV front are presented. The figure indicates that the proposed method can offer competitive results compared with actual schedules; however, it does not present significantly better solutions in terms of patients wait time.

Considering the test problems and the performance of the proposed method, it is noticed that the strength of the proposed method is more apparent in providing schedule with superior results in terms of completion time in comparison with an actual schedules, while providing results that are comparable (or slightly inferior) in terms of patient's wait time.

Overall, it is observed that our method offers quality schedules which provide the OR department management with more options to schedule patients and enables managers to select schedules which are more aligned with their priorities. For instance, if it is important for the management to decrease the facility completion time to reduce the overtime cost, they may choose the schedules with less completion time at an acceptable expense of patients' waiting time.

The experiments suggest that our method was capable of offering solutions that were not dominated by the performance of the actual schedule that had been retrieved from historical data. That is, the proposed method can deliver solutions that are at least as good as actual schedules employed by the practitioner.

7. Insights

Based on our study, and through the collaboration with the OR booking staff of the case study hospital and the health region's OR

booking coordinator, we obtained invaluable insights that can be of interest to both academic and practitioner readers. In this section, we provide our observations and some simple recommendations, which may be useful to improve performance of surgery scheduling in different facilities.

7.1. Overtime should be avoided whenever possible

Based on our interviews with the OR management and the OR booking personnel, aiming for less overtime has more priority in their scheduling than other factors such as minimizing postponements and OR idleness. The following reasons have been mentioned:

- The financial implication of having overtime suggests against it. The postponed cases can ultimately be done during the regular hours of future days if clinical safety measures allow.
- Clinical safety measures recommend avoiding long working days for surgeons. This not only leads to less complications due to surgeons' fatigue in OR departments, but also ensures that patients with major procedures are stabilized in recovery and arrive at surgical wards before evening shifts start.

7.2. It is recommended to schedule cases with longer durations or larger variability (e.g., inpatient cases) in the beginning or in the middle of the day

Cases with longer durations often have larger variability. This insight is mainly driven by the fact that scheduling cases with longer duration at the end of the day may result in more overtimes. In contrast, having longer cases at the beginning or middle of the day may result in more postponements. As mentioned above, managers are usually more inclined to see less overtime than having postponements when it comes to tradeoffs. Additionally, applying this recommendation gives managers more flexibility to compensate for delays due to long cases by taking reactive measures.

7.3. It is recommended to schedule cases of a surgeon in the same room

Scheduling all patients of a surgeon in a specific room will provide the opportunity of having shorter turnaround time as most equipment for a day of surgery may be placed in the same room and it leads to less transportation and setup time. In addition, it would avoid postponements and delays for other surgeons' cases if a case goes longer than expected. This is a desirable outcome as the surgeons' time are often the most expensive and scarce resource in the OR department. Applying this recommendation is subject to availability of enough number of cases for a surgeon to fully utilize an OR.

7.4. It is recommended to schedule similar cases in the same OR

This recommendation mainly aims reducing setup time by placing similar procedures (i.e., cases that require the same equipment) in the same room. It particularly pertains to procedures that require special equipment such as open-heart surgeries or orthopedic cases. Applying this recommendation is subject to the availability of enough number of similar cases to fully utilize ORs.

7.5. It is recommended to schedule cases with less flexibility first

Some cases can be assigned to only one type of resource, whereas others may be flexible and can be assigned to several alternative resources. This recommendation encourages scheduling of less flexible cases prior to more flexible cases to satisfy compatibility requirements. For example, scheduling of a case with an open-heart surgery (that can be performed only in a specific room) should be done in prior to the scheduling of a general surgery case (that can be performed in most rooms).

All these recommendations should be applied in conjunction with the clinical and operational constraints that exist in the OR departments. For instance, usually diabetic and pediatric cases are done in the early mornings to avoid long fasting of patients or usually surgeons' preference and clinical opinion have the final say in the sequencing of the surgeries.

8. Conclusions

In this article, we addressed the appointment scheduling of patients of different types with stochastic service times and heterogeneous service sequences in multi-stage outpatient facility settings. The study of literature in this domain reveals that there exists a gap in the methods that provide competitive optimization schemes while accommodating challenges presented in the appointment scheduling. We target this gap by introducing a multiobjective simulation-based tabu search method enhanced by MP (MATS), which takes advantage of the flexibility of simulation, and efficiently and effectively perform multiobjective optimization. The proposed method integrates the tabu search with mathematical programming (MP) to deliver (near) optimal Pareto fronts for appointment scheduling of patients in order to achieve bi-objectives of minimizing patients wait time and minimizing the completion time of the facility. We use the simulation model to address the stochastic nature of the problem and accommodate several constraints and parameters of the system. To study the performance of the proposed method, we developed many experiments over a range of scheduling factors – the number of patients, the number of patient types, and the coefficient of variation of service times.

We compared the proposed method with a well-known multiobjective evolutionary algorithm, NSGA-II, based on solution quality and computation time. The quality of solutions has been evaluated by two criteria: the closeness to the optimal Pareto front, and the smoothness of distribution of non-dominated solution in the objective function space.

Experiments suggest that MATS presents non-dominated solutions with closer to the Pareto optimal front than NSGA-II. Results also indicate that MATS yields frontiers, which cover a larger range of values in the objective space. In addition, we observed that the solutions presented by MATS are more evenly distributed than those of NSGA-II. Furthermore, MATS requires less computation time than NSGA-II. In conclusion, MATS shows its superior performance in terms of effectiveness and efficiency.

This article also includes a case study of applying proposed method on the scheduling of operating room department of a major Canadian hospital. The proposed method has been implemented and modified to fit the case study problem. Three years worth of actual data has been used to derive the statistical distribution of processing times and to construct the simulation model of the OR department. The results show the superiority of the proposed method in compare to actual schedule extracted from historical data.

The experiments suggest that the proposed method was capable of offering solutions that were not dominated by the performance of the actual schedule. That is, the proposed method can deliver solutions that are at least as good as actual schedules employed by the practitioner. Additionally, it enables the department management to select the schedules which are more aligned with their priorities.

This article also provides insights that interest both academic and practitioner readers. We provide some recommendations gained through collaboration with clinicians and further enhanced

by our modeling and analysis of results. These insights may be found useful to improve performance of surgery scheduling in different facilities.

Overall, the contribution of this paper to the area of expert systems are deemed as follows: first, this paper proposes a method for scheduling of patients with different pathways in healthcare industry in which multiple tools of scheduling are integrated. Second, the performance of proposed method is evaluated and confirmed using the actual data. Third, this research shows how the application of MP can improve the performance of simulation-based optimization methods, which historically only relied on the combination of simulation models and metaheuristic approaches such as NSGA-II.

Several directions for future research are apparent from this study. First, we considered appointment scheduling of patients with punctual arrivals. Future research may include extending the proposed method to address appointment scheduling in presence of no-shows or unpunctual arrivals. Second, this research could be adapted to explicitly address other scheduling challenges such as processes with sequence dependant setup times/cost, and resources with time windows constraints. Finally, alternative approaches (such as stochastic mathematical programming) could be developed and compared with the proposed method. The efficiency and effectiveness of these approaches should be further studied in different environments.

References

Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2005). *Discrete-event system simulation* (4th ed.). Upper Saddle River, New Jersey: Pearson Prentice Hall.

Begen, M. A., Levi, R., & Queyranne, M. (2012). Technical note—a sampling-based approach to appointment scheduling. *Operations Research, 60*(3), 675–681.

Blake, J. T., & Carter, M. W. (1997). Surgical process scheduling: A structured review. *Journal of the Society for Health Systems, 5*(3), 17–30.

Caballero, R., Gandibleux, X., & Molina, J. (2004). MOAMP – a generic multi-objective metaheuristic using an adaptive memory, Technical report, University of Valenciennes.

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

Cayirli, T., & Veral, E. (2003). Outpatient scheduling in health care: A review of literature. *Production and Operations Management, 12*(4), 519–549.

Chow, V., Puterman, M. L., Salehirad, N., Huang, W., & Atkins, D. (2011). Reducing surgical ward congestion through improved surgical scheduling and uncapacitated simulation. *Production and Operations Management, 20*(3), 418–430.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimisation: NSGA-II. *Lecture Notes in Computer Science, 1917*, 849–858.

Denton, B., & Gupta, D. (2003). A sequential bounding approach for optimal appointment scheduling. *IIE Transactions, 35*(11), 1003–1016.

Denton, B., Rahman, A., Nelson, H., & Bailey, A. (2006). Simulation of a multiple operating room surgical suite. In L. Perrone, F. Wieland, J. Liu, B. Lawson, D. M. Nicol, & R. Fujimoto (Eds.), *Proceedings of the 2006 winter simulation conference* (pp. 414–424). New Jersey: IEEE Piscataway.

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.

Durillo, J. J., Nebro, A. J., Luna, F., Dorronsoro B., Alba, E. (2006). {jMetal}: A Java framework for developing multi-objective optimization metaheuristics. *Departamento de Lenguajesy Ciencias dela Computaci'on*, University of Malaga, E.T.S.I. Informatica, Campus de Teatinos, ITI-2006-10. <http://mallba10.lcc.uma.es/wiki/index.php/Tools>.

Ewen, H., & Mönch, L. (2014). A simulation-based framework to schedule surgeries in an eye hospital. *IIE Transactions on Healthcare Systems Engineering, 4*(4), 191–208.

Fries, B. E., & Marathe, V. P. (1981). Determination of optimal variable-sized multiple-block appointment systems. *Operations Research, 29*(2), 324–345.

Gartner, D., & Kolisch, R. (2014). Scheduling the hospital-wide flow of elective patients. *European Journal of Operational Research, 233*(3), 689–699.

Guinet, A., & Chaabane, S. (2003). Operating theatre planning. *International Journal of Production Economics, 85*(1), 69–81.

Gul, S., Denton, B. T., Fowler, J. W., & Huschka, T. (2011). Bi-criteria scheduling of surgical services for an outpatient procedure center. *Production and Operations Management, 20*(3), 406–417.

Güler, G. (2013). A hierarchical goal programming model for scheduling the outpatient clinics. *Expert Systems with Applications, 40*(12), 4906–4914.

Gupta, D., & Denton, B. (2008). Appointment scheduling in health care: Challenges and opportunities. *IIE Transactions, 40*(9), 800–819.

Hansen, M. P. (1997). Tabu search for multiobjective optimization: MOTS. In MCDM, Cape Town, South Africa.

Hsu, V. N., de Matta, R., & Lee, C.-Y. (2003). Scheduling patients in an ambulatory surgical center. *Naval Research Logistics, 50*(3), 218–238.

Jaeggi, D. M., Parks, G. T., Kipouros, T., & Clarkson, P. J. (2008). The development of a multi-objective Tabu search algorithm for continuous optimisation problems. *European Journal of Operational Research, 185*(3), 1192–1212.

Jula, P., & Leachman, R. C. (2010). Coordinated multistage scheduling of parallel batch-processing machines under multiresource constraints. *Operations Research, 58*(4), 933–947.

Klassen, K. J., & Yoogalingam, R. (2009). Improving performance in outpatient appointment services with a simulation optimization approach. *Production and Operations Management, 18*(4), 447–458.

Lamiri, M., Grimaud, F., & Xie, X. L. (2009). Optimization methods for a stochastic surgery planning problem. *International Journal of Production Economic, 120*(2), 400–410.

Lee, S., & Yih, Y. (2014). Reducing patient-flow delays in surgical suites through determining start-times of surgical cases. *European Journal of Operational Research, 233*(2), 620–629.

Marcon, E., & Dexter, F. (2006). Impact of surgical sequencing on post anesthesia care unit staffing. *Health Care Management Science, 9*(1), 87–98.

M'Hallah, R., & Al-Roomi, A. H. (2014). The planning and scheduling of operating rooms: A simulation approach. *Computers & Industrial Engineering, 78*, 235–248.

Min, D., & Yih, Y. (2010). Scheduling elective surgery under uncertainty and downstream capacity constraints. *European Journal of Operational Research, 206*(3), 642–652.

Ozkarahan, I. (1995). Allocation of surgical procedures to operating rooms. *Journal of Medical Systems, 19*(4), 333–352.

Petrovic, D., Morshed, M., & Petrovic, S. (2011). Multi-objective genetic algorithms for scheduling of radiotherapy treatments for categorised cancer patients. *Expert Systems with Applications, 38*(6), 6994–7002.

Pham, D. N., & Klinkert, A. (2008). Surgical case scheduling as a generalized job shop scheduling problem. *European Journal of Operational Research, 185*(3), 1011–1025.

Robinson, L. W., & Chen, R. R. (2003). Scheduling doctors' appointments: Optimal and empirically-based heuristic policies. *IIE Transactions, 35*(3), 295–307.

Saremi, A., Jula, P., ElMekkawy, T., & Wang, P. (2013). Appointment scheduling of outpatient surgical services in a multistage operating room department. *International Journal of Production Economics, 141*(2), 646–658.

Schaumann, E. J., Balling, R. J., & Day, K. (1998). Genetic algorithms with multiple objectives. In *7th AIAA/USAF/NASA/ISSMO symposium on multidisciplinary analysis and optimization*, St. Louis, MO, AIAA. 3, September 2–4, 2114–2123, Paper No. AIAA-98-4974.

Schott, J. R. (1995). Fault tolerant design using single and multicriteria genetic algorithms optimization (Master's thesis). Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

Sier, D., Tobin, P., & Mcgurk, C. (1997). Scheduling surgical procedures. *Journal of the Operational Research Society, 48*, 884–891.

Stigge, M., Plotz, H., Muller, W., & Redlich, J. P. (2006). Reversing CRC-theory and practice, Technical report, SAR-PR-2006-05, Humboldt University, Berlin.

Tang, J., Yan, C., & Cao, P. (2014). Appointment scheduling algorithm considering routine and urgent patients. *Expert Systems with Applications, 41*(10), 4529–4541.

Testi, A., & Tànfani, E. (2009). Tactical and operational decisions for operating room planning: Efficiency and welfare implications. *Health Care Management Science, 12*(4), 363–373.

Tyler, D. C., Pasquariello, C. A., & Chen, C. H. (2003). Determining optimum operating room utilization. *Anesthesia & Analgesia, 96*(4), 1114–1121.

Wang, P. P. (1999). Sequencing and scheduling N customers for a stochastic server. *European Journal of Operational Research, 119*(3), 729–738.

Zhao, Z., & Li, X. (2014). Scheduling elective surgeries with sequence-dependent setup times to multiple operating rooms using constraint programming. *Operations Research for Health Care, 3*(3), 160–167.

Zhou, J., & Dexter, F. (1998). Method to assist in the scheduling of add-on surgical cases, upper prediction bounds for surgical case durations based on the log–normal distribution. *Anesthesiology, 89*(5), 1228–1232.

Zitzler, E., & Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms-a comparative case study. In *Proceeding of the fifth conference on parallel problem solving from nature (PPSN V)* (pp. 292–301). Berlin, Germany: Springer.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation, 7*(2), 117–132.