An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.14, No.3, pp.193-207 (2024)
https://doi.org/10.11121/ijocta.1466

RESEARCH ARTICLE

Artificial bee colony algorithm for operating room scheduling problem with dedicated/flexible resources and cooperative operations

Gulcin Bektur*, Hatice Kübra Aslan

Department of Industrial Engineering, Iskenderun Technical University, Hatay, Türkiye
gulcin.bektur@iste.edu.tr, haslan.mdbf18@iste.edu.tr

ARTICLE INFO

Article history:
Received: 10 October 2023
Accepted: 9 July 2024
Available Online: 12 July 2024

Keywords:
Operating room scheduling
Mixed integer linear programming model
Artificial bee colony algorithm
Multi- resources

AMS Classification 2010:
90B35, 68T20

ABSTRACT

In this study operating room scheduling (ORS) problem is addressed in multi-resource manner. In the addressed problem, besides operating rooms (ORs) and surgeons, the anesthesia team is also considered as an additional resource. The surgeon(s) who will perform the operation have already been assigned to the patients and is a dedicated resource. The assignment of the anesthesia team has been considered as a decision problem and a flexible resource. In this study, cooperative operations are also considered. A mixed integer linear programming (MILP) model is proposed for the problem. Since the problem is NP-hard, an artificial bee colony (ABC) algorithm is proposed for the problem. The solutions of the ABC are compared with the MILP model and random search.

1. Introduction

For many hospitals, operating rooms (ORs) are the costliest unit, but they are also the unit that makes the biggest contribution to the hospital's income. Therefore, the planning of ORs is important for hospitals [1]. Scheduling activities are important in the effective management of ORs. Patient assignment to ORs and determining the starting time of the operations becomes a complex problem due to additional resources [2]. In many hospitals, ORs are scheduled manually. As a result of the manual solutions of such a complex problem, ineffective schedules are created. By using optimization methods in the operating room scheduling (ORS) problem, it may be possible for the hospital management to serve more effectively to patients and managed the ORs efficiently [2].

ORS problems are an important problem that is studied frequently. Literature reviews on the ORS problem are reachable to related articles [3-8]. ORS problems can be classified according to various criteria. These criteria can be considered as the resources, resource types, scheduling period, objective functions, patient types, solution methods and additional features [9].

ORS problems are resource-constrained problems. The limited resources considered in ORS problems are surgeons, downstream beds [10], nurses, anesthesia team and equipment/tools. If the resources under consideration have been previously assigned to patients, they are classified as dedicated resources. If the assignment of resources is considered as a decision problem, it is classified as flexible resources [11].

According to the scheduling period, it is considered as a single/multi period. If scheduling is done for only one day, it is called a single period, if it is done for more than one day, it is called multi-period [12]. The scheduling of ORs is considered in two stages in hospitals. In the first stage, the patient's operation is assigned to a future date and it is long-term planning. The second stage is short-term planning, and it is the stage of determining the operation start times and assignment of ORs to patients on the relevant day. In short-term planning, only daily planning is done in hospitals [2].

Classification of the patients can be made as elective and emergency patients. In some studies, only elective patients are considered. Because in many hospitals, separate ORs are dedicated for emergency surgeries [2]. There are also studies that consider both elective and emergency patients [13]. In some studies, patients are prioritized according to the urgency of their surgery [14].

Many different objective functions are considered in ORS problems. There are multi-objective studies as well as studies that consider single objective function. Minimizing total cost, tardiness, overtime, idle time, waiting time, number of ORs, total completion time, maximum completion time (makespan), maximizing resource balancing [15], maximizing number of patients [16], service level are objective functions of the ORS problems [4].

*Corresponding author

Solution approaches of the ORS problems can be classified as exact and heuristic solutions. Since ORS problems are NP-hard problems, heuristic algorithms were proposed for solving large-sized problems [17]. Heuristic algorithms do not guarantee the best solution. Mathematical modeling [18], decomposition algorithms ([19] and [20]), branch and price, branch and cut [21], column generation [22] are exact solution methods that guarantee the best solution.

Real-life constraints should be taken into account as much as possible while defining the ORS problem. In other words, the problem should reflect the real-life problem as much as possible [23]. For this purpose, additional features are taken into account in many studies. In some studies, some parameters are considered fuzzy or stochastic [11]. Another feature that has been addressed is necessity of more than one surgeon in an operation [2]. Such operations are considered as cooperative operations. All employed surgeons must be available in order to perform the operation of the relevant patient. In some studies, making up of the team is considered [24]. In addition, the skill compatibility feature and the eligibilities on ORs and surgeons are considered. Not every patient can be assigned to every OR or surgeon with eligibility constraints [2].

The ORS literature was reviewed considering the classification of the problem. In most of the early studies on the subject, only surgeons and/or the ORs were considered as resources [25]. Fei et al. [25], proposed a column generation method for the solution of ORS problem. Fei et al. [26], proposed hybrid genetic algorithm (GA) for ORS problem. They considered multi- period feature. Vijayakumar et al. [27], considered nurses and equipment as additional resources. They proposed heuristic algorithms. Priorities of patients was taken into account. Agnetis et al. [19], proposed a decomposition algorithm for ORS problem. Fügener et al. [28], considered multiple downstream units for ORS problem. They proposed an exact solution method. Aringhieri et al. [29], proposed two-level heuristic algorithm for the ORS problem with downstream beds. Jebali et al. [30], used stochastic programming for ORS with downstream beds. They considered multi- period feature. Pariente et al. [31], proposed heuristic algorithm for ORS problem with objective function of maximizing service level. They considered priorities of patients. Wang et al. [32], considered nurses and anesthesiologist as additional resources for the solution of ORS problems. Constraint programming was used in the study. Heydari and Soudi [33], used stochastic programming for ORS problem. They considered downstream beds and elective/emergency patients. Vali- Siar et al. [12], considered nurses, anesthesiologist and downstream beds as additional resources. They proposed genetic algorithm (GA). Hamid et al. [24], considered downstream beds and equipment as additional resources for the ORS problem. NSGA II algorithm was proposed for the solution of the problem. Addis et al. [34], used robust optimization for multi- period ORS problem. Ahmed and Ali [35], used fuzzy TOPSIS and MILP model for the problem of ORS with objective functions of maximizing patient preferences and minimizing total cost. Coban [36], proposed a heuristic and an optimization model for the ORS problem with equipment. Khaniyev et al. [37], proposed heuristic algorithms for ORS problem. They considered uncertainty on parameters. Zhang et al. [11], used stochastic programming for the problem of ORS with downstream beds. Objective function of the problem is minimizing total cost. Britt et al. [38], considered multi- period ORS problem. Downstream beds and equipment were taken into account as additional resources. Roshanaei and Naderi [21], used benders decomposition algorithm for ORS problem. The objective function of the problem was maximization total scheduled surgical times. Park et al. [2], proposed a mathematical model for ORS problem with preferences and cooperative operations. Rachuba et al. [39], taken into account downstream beds for the problem of ORS. Simulation is used for the solution of the problem. Mazloumian et al. [18], proposed a robust multi- objective integer linear programming (MOILP) model for the solution of ORS problem with downstream beds. Azaiez et al. [40], proposed heuristic algorithm for ORS problem with makespan minimization. Makboul et al. [41], considered priorities of patients for ORS problem. Robust optimization was used for the problem. Oliveira et al. [42], considered anesthesiologist as an additional resource for ORS problem with multi- period. Integer linear programming (ILP) model was proposed for the problem. Lotfi and Behnamian [1], proposed multi-objective variable neighborhood search algorithm for multi- period ORS problem.

Heuristic algorithms have been proposed in very few of the studies in which additional resources such as nurses, downstream beds, and anesthesia team are taken into account. In many studies, only surgeons are taken as additional resources. In addition, there are studies that consider downstream beds as additional resources. There are few studies that consider the anesthesia team ([32],[12],[42]). Among these studies, Vali-Siar et al. [12] proposed a GA. Other studies used optimization or simulation methods.

In ORS problems, setup times have been neglected in many studies. However, in real life, the ORs are being prepared for the next operation when an operation is completed. Different equipment and tools are used in different operations. Some tools and equipment are mobile. After an operation is completed, setup must begin for the next operation immediately. During the setup phase, the cleaning of the OR, the transportation of the necessary tools, the sterilization of the used resources, the preparation of the surgeons, nurses and the anesthesia team are carried out [43]. Setup of an operation varies depending on the operation scheduled before it in the same OR. For example, when two operations using the same mobile devices are scheduled sequentially, the setup time may be shortened according to the sequential scheduling of operations using different mobile devices. In other words, setup times are sequence dependent [44]. There are few studies that consider sequence-dependent setup times in ORS problems. It was observed that additional resources considered in ORS problems such as the surgeons, beds and anesthesia team were neglected in many studies about ORS problem with setups [44]. Arnaout and Kulbashian [45], considered sequence dependent setup times in the ORS problem. Additional resources were not considered in the problem. The objective function was makespan minimization. Simulation was used for the problem. Arnout [46], proposed a heuristic algorithm for the solution of the ORS problem with sequence dependent setup times. Additional resources were not taken into account. Hamid et al. [43], used simulation for the ORS problem with sequence dependent setup times. Intensive care unit (ICU) beds were taken into account as an additional resource. The objective function is makespan minimization. Zhao and Li [47], considered sequence dependent setup times in the ORS problem. The use of additional resources was not taken into account in the study. They minimized the total cost. A nonlinear programming model and constraint programming used to solve the problem.

In this study, the problem is defined by considering a state hospital. Anesthesia teams are taken into account in the study. Anesthesia teams consist of specialist doctors, nurses and anesthesia technicians. An anesthesia team accompanies the patient during the operation. Assigning an anesthesia team to patients is an decision problem. In other words, the anesthesia team is a flexible resource. The relevant anesthesia team can serve only one patient at a time. Since there are limited number of anesthesia teams in hospitals, patient waiting occur if there is no team available. In addition, the case of more than one surgeon involvement in some operations is considered. Surgeons can only perform one operation at a time. The patient's operation may be start as long as the employed surgeon or surgeons are idle. Since the assignment of surgeon(s) to operations are predetermined, surgeons are considered as a dedicated resource. In addition, the setup time of the OR for the relevant patient varies depending on the previous operation in the same OR. In other words, operation setup times are sequence dependent. By solving the problem, the anesthesia team and OR are assigned to the patients and the order of the operation is determined. A MILP model and ABC algorithm are proposed for the problem. The proposed algorithm is compared with the MILP and random search.

According to the literature review, it was seen that sequence-dependent setup times were not addressed in many studies [48]. In addition, heuristic algorithm has not been proposed for the ORS problems, which took into account the sequence-dependent setup times and additional flexible/ dedicated resources. Literature is given in Appendix Table A1.

In this study an ORS problem is addressed that is not considered in the literature. Sequence dependent setup times, both flexible and dedicated resources are taken into account and a very complex operating room scheduling problem is addressed. In many studies that is proposed heuristic algorithm to similar problems, mathematical models are used to calculate objective function value of the solutions, due to complexity of the obtaining a feasible solution considering all resources. Collaboration with optimization model may be time consuming. In this study a heuristic algorithm is proposed to solve this complex problem. The unique value of the ABC algorithm is the decoding algorithm, calculation of objective function of the solutions, considering all flexible/dedicated resources.

With this study, a heuristic algorithm is proposed to a problem that is not considered before. The success of the proposed algorithm is demonstrated comparing the results of heuristic with MILP model results through small size problems. Only small size test problems are solvable in reasonable time (3600 seconds). For large size test problems, the ABC algorithm is compared with random search.

In the second section of the study, the problem definition and mathematical model are given. In the section third, heuristic algorithm is given. In the fourth section, test problems are derived and parameters of heuristic algorithms are determined. In addition, the success of the heuristic is demonstrated. The last section is the conclusion section.

2. Optimization model

The addressed problem is described in detail in this section. A MILP model has been proposed. The proposed model is applied to an example problem.

2.1. Problem definition

A state hospital was taken into account in defining the problem under consideration. In the study, operational (short time) scheduling activity was addressed. The assignment of OR to patients, the order of the operations, assignment of anesthesia team to operations are achieved by the solution of the addressed problem. In order to perform the operation of n number of patients, the patient must be assigned to an OR among m ORs. An operation of a patient may begin as long as the surgeon or surgeons who will perform the operation are available and an anesthesia team must be assigned to the operation of the patient. Each surgeon and anesthesia team can only operate on one patient at a time. Some operations may require more than one surgeon. If the surgeon or at least one of the surgeons who will perform the operation is in the operation of another patient or if there is no idle anesthesia team, patient waiting occur. Since both surgeons and the anesthesia team are taken into account, a multi-resource problem is defined. Since the surgeon(s) who will perform the operation of the patient is determined before the operation day, surgeons are a dedicated resource. The anesthesia team to be assigned to the patient is considered as a decision problem and is a flexible resource. Before starting the operation, OR must be prepared for the operation. Setup is done in the same OR immediately after the operation of the previous patient is completed. In the setup phase, the cleaning of the OR, sterilization and positioning of the necessary equipment and devices are conducted. The setup of the operations can be done simultaneously in different ORs. Setup times are sequence dependent.

Characteristics of the model:

- Two different type of resource is considered as flexible and dedicated resource. Surgeon(s) that perform each operation is predetermined and is a dedicated resource. The assignment of anesthesia team to operations is conducted by the MILP model and is a decision problem. The anesthesia teams are a flexible resource.
- Appropriate constraints have been added to the model so that each resource can only perform one operation at a time.
- Before the operation, setup of the operation is conducted.
- More than one surgeon may be involved in an operation.
- If at least one surgeon that will involve in an operation is in another operation at a time, there will be a waiting times of patients.
- If an anesthesia team is needed for different operations at the same time and there is no anesthesia team available, waiting times will be occurred.

Assumptions:

- The operation times and setup times are deterministic.
- The surgeon(s) that perform each operation are predetermined.
- The setup of an operation is conducted after the completion of the previous operation.
- Patients do not have anesthesia team preference.
- All patients have equal priority.
- The resource responsible for the setup is ignored.

2.2. MILP model

Sets and Indices

p, l and k show patient indices and $N=\{p,l,k| p=l=k=1,\ldots,n\}$

o shows OR index and $M=\{o| o=1,\ldots,m\}$

r shows position index and $N=\{r| r=1,\ldots,n\}$

d shows surgeon index and $U=\{d| d=1,\ldots,u\}$

g shows anesthesia team and $A=\{g| g=1,\ldots,a\}$

Parameters

t_p: Operation time of the patient p

SQ_p: Setup of OR for patient p that is scheduled on the first position

$ST_{p,l}$: Setup time of OR for patient l that is scheduled after patient p

B: Very big number

$H_{p,d}$: $\begin{cases} 1, & If\ f\ surgeon\ d\ operates\ the\ patient\ p \\ 0, & Otherwise \end{cases}$

Decision Variables

$y_{p,r,o}$: $\begin{cases} 1, & If\ patient\ p\ is\ assigned\ to\ room\ o\ on\ position\ r \\ 0, & Otherwise \end{cases}$

$x_{p,g}$: $\begin{cases} 1, & If\ patient\ p\ is\ assigned\ to\ anesthesia\ team\ g \\ 0, & Otherwise \end{cases}$

$f_{p,l}$:

$$\begin{cases} 1, If\ operation\ completion\ time\ of\ \ p\ is\ less\ than \\ \quad the\ operation\ start\ time\ of\ patient\ l \\ 0, If\ operation\ completion\ time\ of\ \ l\ is\ less\ than \\ \quad the\ operation\ start\ time\ of\ patient\ p \end{cases}$$

C_p: Operation completion time of patient p

W_p: Operation starting time of patient p

I_p: Waiting time of patient p

T_l: Setup completion time of patient l

C_{max}: Maximum completion time

Model

$$Min\ Z_1 = C_{max} \quad (1)$$

$$T_l + B(1 - y_{l,r,o}) \geq SQ_l \quad \forall\ l, r, o\ and\ r=1 \quad (2)$$

$$T_l - B(1 - y_{l,r,o}) \leq SQ_l \quad \forall\ l, r, o\ and\ r=1 \quad (3)$$

$$T_l + B(2 - y_{l,r,o} - y_{k,r-1,o}) \geq C_k + ST_{k,l}$$
$$\forall\ k, l, r, o, l \neq k,\ r > 1 \quad (4)$$

$$T_l - B(2 - y_{l,r,o} - y_{k,r-1,o}) \leq C_k + ST_{k,l}$$
$$\forall\ k, l, r, o, l \neq k,\ r > 1 \quad (5)$$

$$C_l = T_l + t_l + I_l \ \ \forall\ l \quad (6)$$

$$W_p = T_p + I_p \quad \forall p \quad (7)$$

$$C_l \leq W_p + Bf_{p,l} + B(2 - x_{l,g} - x_{p,g})$$
$$\forall p, l, g\ and\ p < l \quad (8)$$

$$C_p \leq W_l + B(1 - f_{p,l}) + B(2 - x_{l,g} - x_{p,g})$$
$$\forall p, l, g\ and\ p < l \quad (9)$$

$$C_l \leq W_p + Bf_{p,l} + B(2 - H_{p,d} - H_{l,d})$$
$$\forall p, l, d\ and\ p < l \quad (10)$$

$$C_p \leq W_l + B(1 - f_{p,l}) + B(2 - H_{p,d} - H_{l,d})$$
$$\forall p, l, d\ and\ p < l \quad (11)$$

$$\textstyle\sum_g x_{p,g} = 1 \quad \forall p \quad (12)$$

$$\textstyle\sum_p y_{p,r,o} \leq 1 \quad \forall\ r, o \quad (13)$$

$$\sum_r \sum_o y_{p,r,o} = 1 \quad \forall\, p \tag{14}$$

$$\sum_p y_{p,r,o} - \sum_l y_{l,r-1,o} \leq 0 \quad \forall\, r, o \text{ and } r > 1 \tag{15}$$

$$C_{max} \geq C_p \qquad \forall\, p \tag{16}$$

$$y_{p,r,o}, x_{p,g}, f_{p,l} \in \{0,1\} \text{ and}$$

$$C_p, T_l, V_p, I_p, C_{max} \geq 0 \tag{17}$$

Constraint (1) minimizes makespan. Constraints (2-3) calculate the setup completion time of the patients that is scheduled on the first position of each OR. Constraints (4-5) calculate the setup completion time of the patients that is scheduled except for the first position of each OR. Constraint (6) calculates the operation completion time of the patients. Constraint (7) calculates the operation starting time of the patients. Constraints (8-9) prevent simultaneous operations on patients assigned to the same anesthesia team. Constraints (10-11) prevent simultaneous operations on patients assigned to the same surgeon(s). Constraint (12) ensures that an anesthesia team is assigned to each patient. Constraint (13) satisfied that maximum one patient can be assigned to a position of an OR. Constraint (14) provides that assignment of each patient to an OR. Constraint (15) allows patients to be assigned in sequence. Constraint (16) calculates C_{max}. Constraints (17) are sign constraints.

An example is given in Figure 1. Parameters of the problem is given in Appendix Table B1. Accordingly, patients 1,8, and 2 were assigned to OR 1, patients 4, 7 and 3 were assigned to OR 2, and patients 9, 5 and 6 were assigned to OR 3. First Anesthesia team was assigned to the 1st patient, and the 2nd Anesthesia team was assigned to the 4th patient. The anesthesia team assigned to patients is indicated in parentheses next to the patient number in the Figure 1. The anesthesia team assigned to other patients is given in the Figure 1. (Dx) denotes the required surgeon(s) for operation of the relevant patient. For example, for patient 6 the second surgeon (D2) employed for the operation. If the Figure 1 is examined, it is seen that the anesthesia teams and surgeon(s) are performed only one operation at the same time. The setups of operations can be done at the same time. The setup of the operations starts as soon as the previous operation is completed in the same OR. The objective function of the optimal solution is 814.

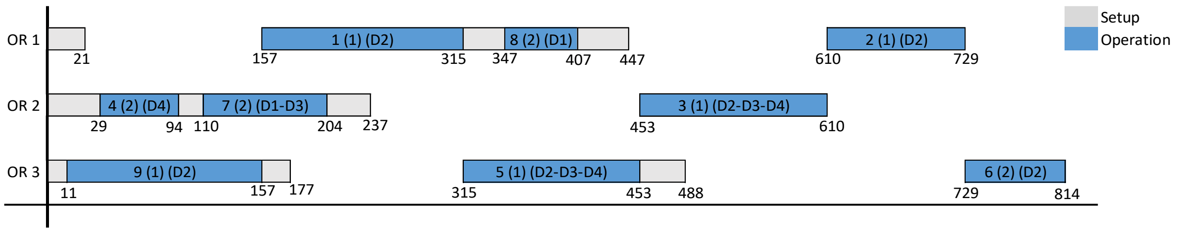

Figure 1. Gantt Chart of the optimal schedule

3. ABC algorithm

3.1. Steps of the algorithm

ABC algorithm was proposed in 2005 by Karaboğa [49]. ABC algorithm was designed by modeling the foraging behavior of bees. ABC algorithm is an algorithm based on swarm intelligence. The algorithm has 3 stages: employed bee stage, onlooker bee stage and scout bee stage. The algorithm makes intensification at the employed and onlooker bee stages. It makes diversification at the scout bee stage. At the end of the employed bee stage, the probability value of the solutions is calculated. Accordingly, the probability values of high-quality solutions are also high. Probability values are taken into account when choosing a solution at the onlooker bee stage. High quality solutions are more likely to be selected [50]. New solution is generated for selected solution by one of the insertion or swap methods. If the new solution produced is a better solution, the existing resource is replaced with the new solution, otherwise the I_i value of the relevant resource is increased by one. In the algorithm, bees are in a position to turn to higher quality resources. For resources whose I_i value is equal to the limit value, the scout bee stage is run and the related solution is replaced with a randomly derived solution. The steps continue until the predetermined number of iterations is achieved [51]. The ABC algorithm is given below [49,51].

```
Procedure: ABC algorithm
Input: Problem parameters, Iteration
number (T), Limit value, Population
size (2N)
Output: Optimal or near optimal
solution
Construct initial population with
size N randomly and calculate the
fitness (f(i)) of the each solution;
t←0;
While (t<T)
      Assume trial value of each
      resource 0;
      //Employed bee phase
      For i=1:N
            Match resource i with a
            resource randomly and
            generate a new resource
            by two point crossover
            and calculate fitness
            value of the new
            resource;
```

```
            If new resource better
            than resource i
                  Replace resource i
                  with the new
                  resource;
                  trial(i)←0;
            Else
                trial(i)←trial(i)+1;
            End
      End
      Determine the maximum fitness
      value as F;
      Calculate probability value of
      resources;
      Probability(i)←0.9(f(i)/F)+0.1;
      //Onlooker bee phase
      Assign each onlooker bee to a
      resource considering
      Probability values;
      For i=1:N
            Match resource i with a
            resource randomly and
            generate a new resource
            by two point crossover
            and calculate fitness
            value of the new
            resource;
            If new resource better
            than resource i
                  Replace resource i
                  with the new
                  resource;
                  trial(i)←0;
            Else
               trial(i)← trial(i)+1;
            End
      End
      Record the resource with best
      fitness value;
      //Scout bee phase
      Find the resource with maximum
      trial number as i*;
      If (trial(i*)>limit)
            Replace resource i with
            a random solution;
            trial(i*)←0;
      End
      t←t+1;
End
```

Fitness values of the solutions are calculated by Equation 18. Z(i) is the objective function value of the resource i.

$$f(i)=\begin{cases} \frac{1}{1+Z(i)} & if\ Z \geq 0 \\ 1+|Z(i)| & if\ Z < 0 \end{cases} \tag{18}$$

3.2. Representation of the solutions and decoding algorithm

The matrix of V_p^{Pop} is used to represent the solutions. Pop denotes the number of individuals in the population. The V_p^{Pop} matrix consists of the number of Pop rows and the number of n (number of patients) columns. The number of columns is equal to the number of patients and the number of rows is equal to the population size. $V_p^i \in$[1,n] and $V_p^i \neq V_l^i$. Each row constitutes of permutation representation of the patients. In other words, patients are ranked randomly in each row of the V_p^{Pop} matrix. The representation of the solutions is given in Figure 2. The assigned OR and the anesthesia team of patients are determined by the decoding algorithm. Therefore, this information is not included in the representation of the solutions.

1st Ind.	V_1^1	V_2^1	...	V_n^1	
2nd Ind.	V_1^2	V_2^2	...	V_n^2	$V_p^i \in$[1,n] and
⋮	⋮	⋮	⋮	⋮	$V_p^i \neq V_l^i\ \forall p,i,l$
popth Ind.	V_1^{pop}	V_2^{pop}	...	V_n^{pop}	

Figure 2. Representation of the solutions

Objective function of the number of Pop solutions are calculated by decoding algorithm. With the decoding algorithm, the patients are assigned to the ORs and the anesthesia team and order of the operations are determined. In addition, anesthesia teams and surgeon(s) operate only one operation at the same time. Some of the abbreviations used in the algorithm are given in the description of the MILP model. Newly defined abbreviations are given below.

OT_o : Operation completion time of the last patient that is assigned to OR o

o^*: The OR that the next patient will be assigned

g_p: The anesthesia team that is assigned to patient p

PA_g: Operation completion time of the patient that is assigned to anesthesia team g

p'_o: The patient that is last assigned to OR o

k_o: The number of patients that is assigned to OR o

seq_k^o: The patient that is scheduled the order of k in OR o

The operation times of the patient V_p^1 that is assigned to OR o' and the patient V_l^1 that is assigned to OR o'' is not overlap as long as the one of the following conditions is met.

Case 1: The operation completion time of patient V_p^1 is smaller than operation starting time of patient V_l^1. This situation is represented by Equation 19. Case 1 is shown in Figure 3.

$$C_{V_p^1} \leq W_{V_l^1} \tag{19}$$

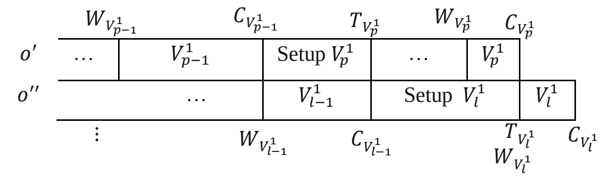

Figure 3. Case 1

In Figure 3, the operation time of V_p^1 and V_l^1 do not overlap. Because the operation completion time of patient V_p^1 is equal to the operation start time of patient V_l^1.

Case 2: The operation start time of patient V_p^1 is greater than the operation completion time of patient V_l^1. This situation is represented by Equation 20. Case 2 is shown in Figure 4.

$$W_{V_p^1} \geq C_{V_l^1} \tag{20}$$

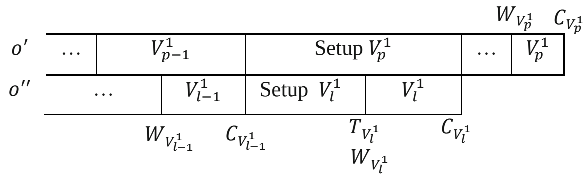

Figure 4. Case 2

In Figure 4, the operation time of V_p^1 and V_l^1 do not overlap. Because the operation starting time of patient V_p^1 is greater than the operation completion time of patient V_l^1.

By using the decoding algorithm, feasible solutions are obtained from each solution representation and the objective functions are calculated.

In the decoding algorithm, first of all, for each solution, the OT_o values, which shows the operation completion time of the patient who was last assigned to the OR o, are taken as 0. The first patient in each row is assigned to the first OR. In the first solution, the first patient is shown as patient V_1^1 and the OR to which it will be assigned is o^*. The setup of the first patient V_1^1 begins at time 0. The setup completion time of the patient V_1^1 ($T_{V_1^1}$) is calculated. Since patient V_1^1 is in the first order of the OR it is calculated as $T_{V_1^1} = SQ_{V_1^1}$. After the setup is completed, the operation starts and the operation start time is shown as $W_{V_1^1}$. The operation completion time ($C_{V_1^1}$) is calculated as $W_{V_1^1}$+$t_{V_1^1}$. The time that OR is used is recorded as an interval ($W_{V_1^1}$−$C_{V_1^1}$). The patient is randomly assigned to the $g_{V_1^1}^*$ anesthesia team. The operation time of the anesthesia team is taken as the interval ($W_{V_1^1} - PA_{V_1^1}$) and the value of $PA_{V_1^1}$ is equal to the value of $C_{V_1^1}$. Patient V_1^1 who was last assigned to OR o^* is recorded as p'_{o^*}. The next patient V_2^1 is assigned to the OR o^* that is the smallest setup completion time ($o^* \leftarrow arg \min_o (OT_O + ST_{p'_o,V_2^1})$). $T_{V_2^1}$ value is calculated as ($OT_O + ST_{p'_o,V_2^1}$) or if no patient has been assigned to the relevant OR yet is calculated as ($OT_o + SQ_{V_2^1}$). First, after determining the o^* OR to which the V_2^1 patient will be assigned, the $T_{V_2^1}$ is calculated. The patient's operation completion time is calculated as $W_{V_2^1} + t_{V_2^1}$. If this value coincides with the operation times of other ORs, the surgeon(s) in the conflicting ORs and the surgeon(s) employed in the operation of patient V_2^1 are checked. If the same surgeon(s) is employed, the operation start time of the V_2^1 is postponed. If different surgeon(s) are employed, the patient V_2^1 is assigned a different anesthesia team than the patients with the overlap. If there is no free anesthesia team, the earliest completed anesthesia team is assigned to the patient. These steps are repeated for all patients. The decoding algorithm is given below.

Procedure: Decoding algorithm
Input: A solution (V_p^1), problem parameters
Output: Objective function of the solution
$OT_o \leftarrow 0$; $k_o \leftarrow 0$;
//The first patient V_1^1 is assigned to first OR and first //anesthesia team;
$o^* \leftarrow 1; T_{V_1^1} \leftarrow SQ_{V_1^1}; C_{V_1^1} \leftarrow SQ_{V_1^1} + t_{V_1^1}$;
$g_{V_1^1}^* \leftarrow 1; PA_{g_{V_1^1}^*} \leftarrow C_{V_1^1}; OT_{o^*} \leftarrow C_{V_1^1}; p'_{o^*} \leftarrow V_1^1; k_{o^*} \leftarrow k_{o^*} + 1$; $seq_k^{o^*} \leftarrow V_1^1$; $W_{V_1^1} \leftarrow T_{V_1^1}$;
For i=2:n
    $o^* \leftarrow \arg\min_o (OT_o + ST_{p'_o,V_i^1})$;
    $T_{V_i^1} \leftarrow OT_{o^*} + ST_{p'_{o^*},V_i^1}$; $p'_{o^*} \leftarrow V_i^1$;
    $seq_k^{o^*} \leftarrow V_i^1$; $W_{V_i^1} \leftarrow T_{V_i^1}$;
    $x \leftarrow T_{V_i^1} + t_{V_i^1}$; z←0;
    //The operation starting time of patient V_i^1 is determined considering //the surgeons;
    While (j<=m)
        Δ←1;
        For l=1:k_j
            U←seq_l^j;
            If ($x \leq W_U$)or($W_{V_i^1} \geq C_U$)
                //No overlap
            Else
                z←z+1;
                Overlap(z)=U;
                If($H_{U,d} == H_{V_i^1,d}$)
                    $W_{V_i^1} \leftarrow C_U$;
                    $x \leftarrow W_{V_i^1} + t_{V_i^1}$;
                    j←1; Δ←0;
                End
            End
            If (Δ==0)
                Break
            End
        End
        If (Δ==1)
            j←j+1;
        End
    End
    z←0;
    //Assignment of anesthesia team and updating of operation starting //time considering anesthesia teams;
    For j=1:m

```
            For l=1:k_j
                U←seq_l^j;
                If(x≤W_U)or(W_{V_i^1} ≥ C_U)
                    //No overlap
                Else
                    z←z+1;
                Overlap(z)=U;
                G\{g_U^*};
                End
            End
        End
        If (G=={})
            g*_{V_i^1} ← arg min_g PA_{g*};
            W_{V_i^1} ← max(PA_{g*_{V_i^1}}, W_{V_i^1});
        Else
            g*_{V_i^1} ← arg max_{g∈{G}} PA_{g*};
            W_{V_i^1} ← max(PA_{g*_{V_i^1}}, W_{V_i^1});
        End
        C_{V_i^1} ← W_{V_i^1} + t_{V_i^1}; PA_{g*_{V_i^1}} ← C_{V_i^1};
        OT_{o*} ← C_{V_i^1};  k_{o*} ← k_{o*} + 1;
End
```

4. Computational results

4.1. Parameters of the heuristic

Although in most of the studies on heuristic algorithms parameter levels are determined without an analytic method, in this study Taguchi experimental design (TED) method is used to determine the levels of the ABC algorithm parameters. The parameters of the ABC algorithm are N, T and limit value. Firstly, alternative parameter levels are determined through preliminary experiments and given in Table 1. L27 orthogonal array is chosen due to there are 3 parameters and 3 levels for each parameter. In TED method, signal-to-noise ratio (S/N) is used as a measure to determine the characteristics of engineering problems. To optimize the ABC algorithm parameters "the smaller, the better" performance criterion is used in TED method due to the addressed problem has a minimization objective function. The calculation of S/N is given in Equation 21. In Equation 21, n is the number of observations in each experiment and Y_i is the objective function of ABC algorithm with the related parameters. The optimal parameters are selected considering the highest S/N values. Minitab 16 for Windows (Minitab Inc.) is used to apply TED method to problem.

$$\frac{S}{N} = -10 \times \log\left(\frac{1}{n}\sum_{i=1}^{n} Y_i^2\right) \tag{21}$$

For the test problem with 7 ORs algorithm was run at the relevant parameter levels. The main effects plot for S/N ratios for the algorithm is given in Figure 5. In ABC algorithm, N level sets to 1000, T level is 100 and limit is 10.

Table 1. Parameter levels of the ABC algorithm

Parameters	Levels
N	500/750/1000
T	50/75/100
limit	5/7/10

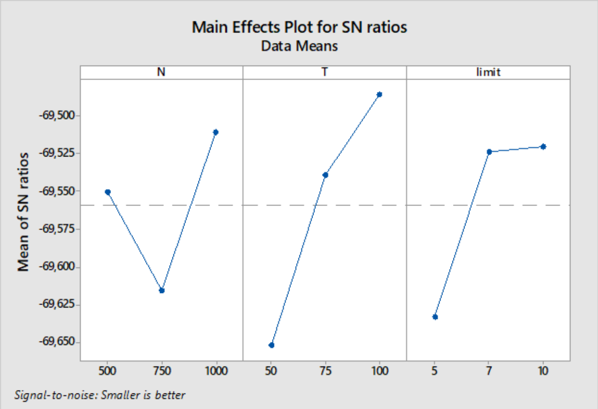

Figure 5. S/ N ratios of the algorithm

4.2. Comparisons

Properties of test problems are given in this section. The number of ORs (m) set to 3, 5, 7 or 10. The number of patients was taken as 3*m, 5*m, 7*m and 10*m. The number of surgeons was taken as 4, 7, 10 and 14, and the number of anesthesia team as 2, 3, 5 and 7. The parameter t_p were derived according to a uniform distribution in the U(40,170) range. Sequence-dependent setup times are derived in accordance with the uniform distribution in the range of U(20,50), U(10,40) or U(30,85). The $H_{p,d}$ parameter is derived so that 60% of the patients receive service from only one surgeon, 25% of the patients receive service from two surgeons and 15% from 3 surgeons. For each problem type two test problems are derived.

Test problems are run with the MILP model, ABC algorithm and random search. The results of random search also is an upper bound for the related test problem since for all test problems random search gave worse solution than ABC algorithm. In random search, random solutions are generated and the objective function of these solutions are calculated using the proposed decoding algorithm. The random search is run the same duration of ABC algorithm for the related test problem.

The time limit of the MILP model is 3600 seconds. The results are given in Table C1-C4 in Appendix section. Objective function values, CPU values and Error values obtained by using the relevant algorithm are given in the tables. Error value is calculated with Equation 22.

$$Error = \frac{(Solution\ of\ the\ algorithm - The\ obtained\ best\ solution)}{The\ obtained\ best\ solution} \tag{22}$$

The results with 3 OR are given in Table C1. Model gave optimal solutions for 7 test problems with number of 9 or 15 patients. Also, the heuristic algorithm found optimal solutions to these problems. ABC algorithm gave better results except six test problems. MILP model found no feasible solutions to number of 5 test problems with the number of 21 or 30 patients within 3600 seconds. For other test problems feasible solutions were found by MILP model. Accordingly, the ABC algorithm gave the better results for all test problems. The results with 5 ORs are given in Table C2. MILP model found feasible solutions to test problems with the number of 15 or 25 patients within time limit. MILP model could not find a solution to test problems with the number of 35 or 50 patients within time limits. Accordingly, the ABC algorithm also found better solutions for test problems with 5 ORs. The results of 7 ORs are given in Table C3. MILP model found feasible solutions to test problems with number of 21 patients. No feasible solutions were found other test problems by MILP model for the number of 7 ORs. The results of 10 ORs are given in Table C4. According to Table C4, MILP model found feasible solution to only one test problem. The ABC algorithm found better solutions than MILP model and random search.

5. Conclusions

ORs are one of the most important resources of hospitals. Therefore, effective scheduling of ORs has an important role in the effective management of the hospital. ORS problems are multi-resource problems. In this study, the ORS problem was defined by considering the anesthesia team as well as the surgeons. While surgeons are a dedicated resource, the anesthesia team is a flexible resource. In the ORS problem, sequence dependent setup times are taken into account. Although the ORS problem is an important problem, there are few studies that take into account the sequence-dependent setup times. A MILP model is proposed. ABC algorithm has been developed for large scale test problems. A heuristic algorithm is proposed for the first time to solve the ORS problem with multi-resource, sequence-dependent setup times. An algorithm has been developed to calculate the objective functions of the solutions. The proposed ABC algorithm is compared with MILP model. As a result, the ABC algorithm gave more successful results than MILP model. In future studies, the problem can be handled with multi- objective functions. Objective functions such as tardiness minimization, maximization of resource utilization may be considered besides makespan minimization. In multi- objective optimization problems, pareto optimal solutions are found. In multi- objective optimization, all obtained solutions are compared with each other to select non-dominated solutions in solution space that is increased the complexity of the problem. Different methods may be used such as Augmented ε- constraint method to obtain pareto optimal solutions to multi- objective optimization problems. Extracting Pareto optimal solutions from the solution space can significantly increase the running time of the heuristic algorithm. In this study, surgeons and anesthesia teams are considered as resources. In future studies, the resource conducts the setup and other resources such as machines used in operations may be taken into account. In this study all patients have same priority. In future studies patients may be prioritized. In this study, operations and setup times are considered deterministic. Stochastic parameters can be taken into account. Different heuristic algorithms may be proposed to solve the problem or exact solution methods may be used to solve the problem.

References

[1] Lotfi, M., & Behnamian, J. (2022). Collaborative scheduling of operating room in hospital network: Multi- objective learning variable neighborhood search. *Applied Soft Computing*, 116, 108233.

[2] Park, J., Kim, B., Eom, M., & Choi, B. K. (2021). Operating room scheduling considering surgeons' preferences and cooperative operations. *Computers and Industrial Engineering*, 157, 107306.

[3] Riet, C. V., & Demeulemeester, E. (2015). Trade-offs in operating room planning for electives and emergencies: A review. *Operations Research for Health Care*, 7, 52- 69.

[4] Cardeon, B., Demeulemeester, E., & Belien, J. (2010). Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, 2010, 201, 921- 932.

[5] Rahimi, I., & Gandomi, A. H. (2021). A comprehensive review and analysis of operating room and surgery scheduling. *Archives of Computational Methods in Engineering*, 28, 1667-1688.

[6] Zhu, S., Fan, W., Yang, S., Pei, J., & Pardolos, P. M. (2019). Operating room planning and surgical case scheduling: A review of literature. *Journal of Combinatorial Optimization*, 37, 757- 805.

[7] Harris, S., & Claudio, D. (2022). Current trends in operating room scheduling 2015 to 2020: A literature review. *Operations Research Forum*, 3, 21- 63.

[8] Ferrand, Y. B., Magazine, M. J., & Rao, U. S. (2014). Managing operating room efficiency and respensiveness for emergency and elective surgeries- A literaure survey. *IIE Transactions on Healthcare Systems Engineering*, 4 (1), 49- 64.

[9] Riise, A., Mannino, C., & Burke, E. K. (2016). Modelling and solving generalised operational surgery scheduling problems. *Computers and Operations Research*, 66, 1- 11.

[10] Augusto, V., Xie, X., & Perdomo, V. (2010). Operating theatre scheduling with patient recovery in both operating rooms and recovery beds. *Computers and Industrial Engineering*, 2010, 58, 231- 238.

[11] Zhang, J., Dridi, M., & Moudni, A. E. (2021). A two-phase optimization model combining Markov decision process and stochastic programming for advance surgery scheduling. *Computers and Industrial Engineering*, 160, 107548.

[12] Vali- Siar, M. M., Gholami, S., & Ramezanian, R. (2018). Multi- period and multi- resource operating room scheduling under uncertainty: A case study. *Computers and Industrial Engineering*, 126, 549-568.

[13] Rachuba, S., & Werners, B. (2014). A robust approach for scheduling in hospitals using multiple objectives. *Journal of Operational Research Society*, 65, 546- 556.

[14] Cardoen, B., Demeulemeester, E., & Belien, J. (2009). Optimizing a multiple objective surgical case sequencing problem. *International Journal of Production Economics*, 119, 354- 366.

[15] Cappanera, P., Visintin, F., & Banditori, C. (2014). Comparing resource balancing criteria in master surgical scheduling: A combined optimisation-simulation approach. *International Journal of Production Economics,* 2014, 158, 179- 196.

[16] Azar, M., Carrasco, R. A., & Mondschein, S. (2022). Dealing with uncertain surgery times in operating room scheduling. *European Journal of Operational Research*, 2022, 299, 377- 394.

[17] Landa, P., Aringhieri, R., Soriano, P., Tanfani, E., & Testi, A. (2016). A hybrid optimization algorithm for surgeries scheduling. *Operations Research for Health Care*, 8, 103- 114.

[18] Mazloumian, M., Baki, M. F., & Ahmadi, M. (2022). A robust multiobjective integrated master surgery Schedule and surgical case assignment model at a publicly funded hospital. *Computers and Industrial Engineering*, 163, 107826.

[19] Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C., & Pranzo, M. (2014). A decomposition approach for the combined master surgical schedule and surgical case assignment problems. *Health Care Management Science*, 2014, 17, 49- 59.

[20] Roshanaei, V., Luong, C., Aleman, D. M., & Urbach, D. R. (2020). Reformulation, linearization, and decomposition techniques for balanced distributed operating room scheduling. *Omega*, 93, 102043.

[21] Roshanaei, V., & Naderi, B. (2021). Solving integrated operating room planning and scheduling: Logic- based Benders decomposition versus Branch-price and cut. *European Journal of Operational Research*, 293, 65- 78.

[22] Range, T. M., Lusby, R. M., & Larsen, J. (2014). A column generation approach for solving the patient admission scheduling problem. *European Journal of Operational Research*, 235, 252- 264.

[23] Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C., & Pranzo, M. (2012). Long term evaluation of operating theater planning policies. *Operations Research for Health Care*, 2012, 1, 95-104.

[24] Hamid, M., Nasiri, M. M., Werner, F., Sheikhahmadi, F., & Zhalechian, M. (2019a) Operating room scheduling by considering the decision- making styles of surgical team members: A comprehensive approach. *Computers and Operations Research*, 108, 166- 181.

[25] Fei, H., Chu, C., & Meskens, N. (2009). Solving a tactical operating room planning problem by a column- generation- based heuristic procedure with four criteria. *Annals of Operations Research*, 166, 91- 108.

[26] Fei, H., Meskens, N., & Chu, C. (2010). A planning and scheduling problem for an operating theatre using on open scheduling strategy. *Computers and Industrial Engineering*, 58, 221- 230.

[27] Vijayakumar, B., Parikh, P. J., Scott, R., Barnes, A., & Gallimore, J. (2013). A dual bin packing approach to scheduling surgical cases at a publicly- funded hospital. *European Journal of Operational Research*, 224, 583- 591.

[28] Fügener, A., Hans, E. W., Kolisch, R., Kortbeek, N., & Vanberkel, P. T. (2014). Master surgery scheduling with consideration of multiple downstream units. *European Journal of Operational Research*, 239, 227- 236.

[29] Aringhieri, R., Landa, P., Soriano, P., Tanfani, E., & Testi, A. (2015). A two level metaheuristic for the operating room scheduling and assignment problem. *Computers and Operations Research*, 2015, 54, 21-34.

[30] Jebali, A., & Diabat, A. (2015). A stochastic model for operating room planning under capacity constraints. *Journal of Production Research*, 53, 24, 7252- 7270.

[31] Pariente, J. M., Hans, E. W., Framinan, J. M., & Gomez- Cia, T. (2015). New heurisitcs for planning operating rooms. *Computers and Industrial Engineering*, 90, 429- 443.

[32] Wang T., Meskens, N., & Duvivier, D. (2015). Scheduling operating theatres: Mixed integer programming vs. constraint programming. *European Journal of Operational Research*, 247, 401- 413.

[33] Heydari, M., & Soudi, A. (2016). Predictive/ Reactive planning and scheduling of a surgical süite with emergency patient arrival. *Journal of Medical Systems*, 40, 30.

[34] Addis, B., Carello, G., Grosso, A., & Tanfani, E. (2016). Operating room scheduling and rescheduling: A Rolling horizon approach. *Flexible Services and Manufacturing Journal*, 2016, 28, 206-232.

[35] Ahmed, A., & Ali, H. (2020). Modeling patient preference in an operating room scheduling problem.

Operations Research for Health Care, 2020, 25, 100257.

[36] Coban, E. (2020). The effect of multiple operating room scheduling on the sterilization schedule of reusable medical devices. *Computers and Industrial Engineering*, 147, 106618.

[37] Khaniyev, T., Kayış, E., & Güllü, R. (2020). Next-day operating room scheduling with uncertain surgery durations: Exact analysis and heurisitcs. *European Journal of Operational Research*, 286, 49- 62.

[38] Britt, J., Baki, M. F., Azab, A., Chaouch, A., & Li, X. (2021). A stochastic hierarchical approach for the master surgical scheduling problem. *Computers and Industrial Engineering*, 2021, 158, 107385.

[39] Rachuba, S., Imhoff, L., & Werners, B. (2022). Tactical blueprints for surgical weeks- An integrated approach for operating rooms and intensive care units. *European Journal of Operational Research*, 298, 243- 260.

[40] Azaiez, M., Gharbi, A., Kacem, I., Makhlouf, Y., & Masmoudi, M. (2022). Two- stage no- wait hybrid flow shop with inter- stage flexibility for operating room scheduling. *Computers and Industrial Engineering*, 2022, 168, 108040.

[41] Makboul, S., Kharraja, S., Abbassi, A., & Alaoui, A. (2022). A two- stage robust optimization approach for the master surgical schedule problem under uncertainty considering downstream resources. *Health Care Management Science*, 25, 63- 88.

[42] Oliveira, M., Visintin, F., Santos, D., & Marques, I. (2023). Flexible master surgery scheduling: Combining optimization and simulation in a Rolling horizon approach. *Flexible Services and Manufacturing Journal*.

[43] Hamid, M., Hamid, M., Musavi, M., & Azadeh, A. (2019b) Scheduling elective patients based on sequence- dependent setup times in an open- heart surgical department using an optimization and simulation approach. *Simulation: Transactions of the Society for Modelling and Simulation International*, 95 (12), 1141- 1164.

[44] Ciavotta, M., Dellino, G., Meloni, C., & Pranzo, M. (2010). A rollout algorithmic approach for complex parallel machine scheduling in healthcare operations. Operations Research for Patient: Centered health care delivery: *Proceeding of the XXXVI International ORAHS Conference*.

[45] Arnaout, J. M., & Kulbashian, S. (2008). Maximizing the utilization of operating rooms with stochastic times using simulation. *Proceedings of the 2008 Winter Simulation Conference.*

[46] Arnaout, J. (2010). Heuristics for the maximization of operating rooms utilization using simulation. *Simulation*, 2010, 86, 8-9, 573- 583.

[47] 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, 160- 167.

[48] Karakas, E., & Ozpalamutcu, H. (2019). A genetic algorithm for fuzzy order acceptance and scheduling problem. *An International Journal of Optimization and Control: Theories & Applications*, 9 (2), 186-196.

[49] Karaboğa, D. (2005). An idea based on honey bee swarm for numarical optimization: Technical report. *Erciyes University*.

[50] Lei, D., & He, S. (2022). An adaptive artificial bee colony for unrelated parallel machine scheduling with additional resource and maintenance. *Expert Systems with Applications*, 205, 117577.

[51] Xu, Y., & Wang, X. (2021). An artificial bee colony algorithm for scheduling call centres with weekend-off fairness. *Applied Soft Computing*, 109, 107542.

Gulcin Bektur *received her PhD degree from Eskisehir Osmangazi University, department of Industrial Engineering. She is an Assistant Professor at Iskenderun Technical University, department of Industrial Engineering. Her research areas are scheduling, vehicle routing, mathematical modelling and heuristic search.*
http://orcid.org/0000-0003-4313-7093

Hatice Kübra Aslan *received her Bachelor degree from Iskenderun Technical University, department of Industrial Engineering. She works as an industrial engineer in a production company. Her research areas are workforce scheduling and mathematical modelling.*
http://orcid.org/0000-0001-5020-3920

Appendix

A. Related studies

Table A1. Related studies

Article	Add. Res.	Res. Type	Patients	Obj. Funct.	Method	Add Pro.
Zhang, 2021	d	a	a	Minimizing total cost	Column Gen. Based Heuristic, stoch. Prog.	a,b
Rachuba, 2022	d	b	a	Maximizing total number of patients	Chance const. Opt., Simulation	a,b
Park, 2021	a	a	a	Minimizing number of ORs and overtime	Mathematical model	c,e
Mazloumian, 2022	d	a	a	Minimizing total waiting Minimizing postponed Minimizing loss incurred	Robust MOILP Model	a,b
Lotfi, 2022	a	b	c	Minimizing total comp. time Minimizing makespan	Multi- obj. variable neigh. Search	b
Khaniyev, 2020	-	-	a	Minimizing waiting, idle and overtime	Heuristic algorithms	a
Hamid, 2019 a	a,d,e	a	a	Minimizing total cost Maximizing service level Maximizing consistecy score	NSGA II	f
Britt, 2021	a,d,e	b	a	Minimizing total cost Minimizing number of ORs	Hybrid heuristic alg.	a,b,c
Azaiez, 2022	d	b	a	Minimizing max. compl. time	MILP, Heuristics	b
Aringhieri, 2015	a,d	a	a	Minimizing total cost of waiting time	Two level heuristic, ILP model	
Addis, 2016	-	-	a	Minimizing total waiting time and tardiness	Robust optimization	a,b
Roshanaei, 2021	a	a	a	Maximizing total scheduled surgical times	Benders decomposition, MILP	
Wang, 2015	a,b,c, d	b	a	Minimizing makespan	MILP, Constraint prog.	f
Makboul, 2022	a,d	c	a	Maximizing score of surgeries	Robust opt.	a,b,f
Fei, 2010	a	a	a	Minimizing cost	Hybrid GA	b
Coban, 2020	e	a	a	Minimizing total cost	Heuristic, MILP model	
Pariente, 2015	a	b	a	Maximizing service level	Heuristics	f
Oliveira, 2022	a,c	c	a	Minimizing deviations	ILP model, simulation	b
Ahmed, 2020	a	b	a	Max. Patient preference Minimizing total cost	Fuzzy TOPSIS, MILP model	f
Agnetis, 2014	a	b	a	Maximizing total score	Decomposition	b
Jebali, 2015	d	b	a	Minimizing costs	Stoch. Prog.	a,b
Vijayakumar, 2013	a,b,e	a	a	Maximizing number of patients	Heuristic	b,f
Heydari, 2016	d	b	c	Minimizing makespan and overtime	Stochastic prog.	a,b
This study	**a,c**	**c**	**a**	**Minimizing max. compl. time**	**Heuristic,MILP**	**d,e**

Add. Resources: a: Surgeon, b: Nurse, c: Anesthesiologist, d: Downstream beds, e: Equipment/Tools

Additional Resource Type: a: Dedicated, b: Flexible, c: Hybrid

Patients: a: Elective, b: Emergency, c: Hybrid

Add. properties: a: Uncertainty on parameters, b: Multi- period, c: Preferences, d: Setup times, e: Cooperative operations, f: Priorities of patients

B. Parameters of example problem

The proposed MILP model was coded in the GAMS 24.0.2 program. Solved with CPLEX solver. For the first test problem, the MILP model was run. In Table A1 parameters of the problem are given. There are 9 patients, 3 ORs, 4 surgeons and 2 anesthesia teams.

Table B1. Parameters of SQ_p, $ST_{p,l}$ $H_{p,d}$ and t_p

p	t_p	SQ_p	$ST_{p,l}$									$H_{p,d}$			
			1	2	3	4	5	6	7	8	9	1	2	3	4
1	158	21	29	32	34	12	39	25	33	32	35	0	1	0	0
2	119	26	15	24	29	38	35	37	27	27	36	0	1	0	0
3	157	20	11	37	22	11	32	15	14	28	18	0	1	1	1
4	65	29	20	22	22	22	28	15	16	13	20	0	0	0	1
5	138	34	33	17	32	31	35	35	19	19	26	0	1	1	1
6	85	32	20	35	34	27	18	30	17	24	22	0	1	0	0
7	94	14	26	40	33	39	17	26	12	33	28	1	0	1	0
8	60	35	36	40	38	22	10	26	16	17	20	1	0	0	0
9	146	11	13	32	32	26	20	35	27	39	37	0	1	0	0

C. Solutions of test problems

Table C1. Solution of test problems with 3 ORs

n	ST	MILP Model			ABC			Random Search	
		Z	CPU	Error	Z	CPU	Error	Z	Error
9	U(10,40)	**814***	1882	0	**814**	6.59	0	**814**	0
9	U(10,40)	**630***	2804	0	**630**	5.61	0	632	0.003
9	U(20,50)	**759***	1620	0	**759**	4.91	0	**759**	0
9	U(20,50)	**742***	1874	0	**742**	4.64	0	758	0.022
9	U(30,85)	530	3600	0.017	**521**	6.13	0	**521**	0
9	U(30,85)	514	3600	0.024	**502**	5.45	0	**502**	0
15	U(10,40)	**980***	1235	0	**980**	9.39	0	995	0.015
15	U(10,40)	**1065***	2152	0	**1065**	7.78	0	1087	0.021
15	U(20,50)	1061	3600	0.002	**1059**	7.96	0	**1059**	0
15	U(20,50)	**878***	2252	0	**878**	9.28	0	**878**	0
15	U(30,85)	998	3600	0.034	**965**	7.74	0	989	0.025
15	U(30,85)	983	3600	0.005	**978**	7.59	0	1001	0.023
21	U(10,40)	1482	3600	0.086	**1365**	11.86	0	1385	0.015
21	U(10,40)	1281	3600	0.063	**1205**	14.79	0	1227	0.018
21	U(20,50)	1434	3600	0.075	**1334**	12.49	0	1337	0.002
21	U(20,50)	-	3600	-	**1166**	12.57	0	1200	0.029
21	U(30,85)	1698	3600	0.103	**1540**	12.71	0	1621	0.052
21	U(30,85)	-	3600	-	**1338**	13.65	0	1452	0.085
30	U(10,40)	1908	3600	0.181	**1616**	23.86	0	1663	0.029
30	U(10,40)	-	3600	-	**1500**	20.52	0	1559	0.039
30	U(20,50)	2277	3600	0.368	**1665**	16.98	0	1738	0.044
30	U(20,50)	-	3600	-	**1644**	17.58	0	1658	0.009
30	U(30,85)	2172	3600	0.225	**1773**	22.45	0	1868	0.054
30	U(30,85)	-	3600	-	**1916**	16	0	1927	0.006
* optimal solution		Average		0.062			0		0.02

Table C2. Solution of test problems with 5 ORs

n	ST(p,l)	MILP Model			ABC			Random Search	
		Z	CPU	Error	Z	CPU	Error	Z	Error
15	U(10,40)	680	3600	0.012	**672**	10.75	0	684	0.018
15	U(10,40)	579	3600	0.032	**561**	12.44	0	564	0.005
15	U(20,50)	580	3600	0.133	**512**	10.04	0	513	0.002
15	U(20,50)	**566**	3600	0	**566**	11.8	0	577	0.019
15	U(30,85)	793	3600	0.025	**774**	12.89	0	798	0.031
15	U(30,85)	731	3600	0.046	**699**	10.53	0	778	0.11
25	U(10,40)	1276	3600	0.44	**886**	23.33	0	920	0.038
25	U(10,40)	-	3600	-	**915**	17.49	0	923	0.009
25	U(20,50)	1359	3600	0.162	**1170**	24.35	0	1189	0.016
25	U(20,50)	1391	3600	0.218	**1142**	19.97	0	1184	0.037
25	U(30,85)	1930	3600	0.885	**1024**	21.69	0	1198	0.169
25	U(30,85)	-	3600	-	**1012**	23.56	0	1032	0.02
35	U(10,40)	-	3600	-	**1191**	23.3	0	1230	0.033
35	U(10,40)	-	3600	-	**1317**	33.58	0	1378	0.046
35	U(20,50)	-	3600	-	**1514**	27.9	0	1698	0.121
35	U(20,50)	-	3600	-	**1494**	29.1	0	1495	0.001
35	U(30,85)	-	3600	-	**1277**	31.86	0	1319	0.033
35	U(30,85)	-	3600	-	**1293**	30.96	0	1387	0.073
50	U(10,40)	-	3600	-	**2022**	40.71	0	2023	0
50	U(10,40)	-	3600	-	**1892**	35.37	0	2077	0.098
50	U(20,50)	-	3600	-	**2180**	41.71	0	2223	0.02
50	U(20,50)	-	3600	-	**1967**	39.75	0	2094	0.065
50	U(30,85)	-	3600	-	**1872**	38.53	0	1999	0.068
50	U(30,85)	-	3600	-	**1960**	41.46	0	2111	0.077
		Average		0.19			0		0.046

Table C3. Solution of test problems with 7 ORs

n	ST(p,l)	MILP Model			ABC			Random Search	
		Z	CPU	Error	Z	CPU	Error	Z	Error
21	U(10,40)	542	3600	0.146	**473**	17.27	0	497	0.051
21	U(10,40)	641	3600	0.009	**635**	14.56	0	645	0.016
21	U(20,50)	747	3600	0.201	**622**	16.83	0	629	0.011
21	U(20,50)	929	3600	0.078	**862**	16.33	0	927	0.075
21	U(30,85)	801	3600	0.004	**798**	19.92	0	801	0.004
21	U(30,85)	712	3600	0.029	**692**	14.11	0	785	0.134
35	U(10,40)	-	3600	-	**876**	31.26	0	985	0.124
35	U(10,40)	-	3600	-	**1022**	36.01	0	1108	0.084
35	U(20,50)	-	3600	-	**1163**	32.53	0	1189	0.022
35	U(20,50)	-	3600	-	**1064**	26.07	0	1089	0.023
35	U(30,85)	-	3600	-	**1193**	32.8	0	1198	0.004
35	U(30,85)	-	3600	-	**927**	28.63	0	1010	0.09
49	U(10,40)	-	3600	-	**1198**	46.04	0	1267	0.058
49	U(10,40)	-	3600	-	**1275**	41.35	0	1355	0.063
49	U(20,50)	-	3600	-	**1198**	38.9	0	1299	0.084
49	U(20,50)	-	3600	-	**1340**	42.65	0	1374	0.025

49	U(30,85)	-	3600	-	**1540**	39.15	0	1643	0.067
49	U(30,85)	-	3600	-	**1602**	40.86	0	1620	0.011
70	U(10,40)	-	3600	-	**1998**	65.94	0	2104	0.053
70	U(10,40)	-	3600	-	**1619**	68.37	0	1779	0.099
70	U(20,50)	-	3600	-	**1868**	66.32	0	1963	0.051
70	U(20,50)	-	3600	-	**1765**	63.92	0	1910	0.082
70	U(30,85)	-	3600	-	**1766**	68.58	0	1856	0.051
70	U(30,85)	-	3600	-	**1925**	65.78	0	2034	0.057
			Average	0.077			0		0.055

Table C4. Solution of test problems with 10 ORs

n	ST(p,l)	MILP Model			ABC			Random Search	
		Z	CPU	Error	Z	CPU	Error	Z	Error
30	U(10,40)	1101	3600	0.59	**691**	28.35	0	741	0.072
30	U(10,40)	-	3600	-	**780**	29.31	0	803	0.029
30	U(20,50)	-	3600	-	**735**	24.28	0	743	0.011
30	U(20,50)	-	3600	-	**733**	25.44	0	764	0.042
30	U(30,85)	-	3600	-	**639**	25.09	0	658	0.03
30	U(30,85)	-	3600	-	**826**	24.54	0	995	0.205
50	U(10,40)	-	3600	-	**1154**	50.48	0	1163	0.008
50	U(10,40)	-	3600	-	**985**	46.8	0	1051	0.067
50	U(20,50)	-	3600	-	**943**	51.19	0	1012	0.073
50	U(20,50)	-	3600	-	**1082**	53.68	0	1083	0.001
50	U(30,85)	-	3600	-	**1075**	51.1	0	1159	0.078
50	U(30,85)	-	3600	-	**1119**	46.44	0	1252	0.119
70	U(10,40)	-	3600	-	**1686**	75.05	0	1721	0.021
70	U(10,40)	-	3600	-	**1623**	75.81	0	1641	0.011
70	U(20,50)	-	3600	-	**1706**	73.99	0	1714	0.005
70	U(20,50)	-	3600	-	**1262**	80.06	0	1405	0.113
70	U(30,85)	-	3600	-	**1469**	81.6	0	1614	0.099
70	U(30,85)	-	3600	-	**1563**	73.56	0	1593	0.019
100	U(10,40)	-	3600	-	**2332**	112.7	0	2383	0.022
100	U(10,40)	-	3600	-	**2373**	106.3	0	2393	0.008
100	U(20,50)	-	3600	-	**2086**	135.8	0	2254	0.081
100	U(20,50)	-	3600	-	**1927**	137.1	0	2121	0.101
100	U(30,85)	-	3600	-	**2034**	146.9	0	2287	0.124
100	U(30,85)	-	3600	-	**2300**	155.2	0	2406	0.046
			Average	0.59			0		0.057