Software Tool for Operating Room Scheduling in a Spanish Hospital Department

Daniel Clavel*, Diana Botez†, Cristian Mahulea*, Jorge Albareda‡
*Aragón Institute of Engineering Research (I3A), University of Zaragoza, Maria de Luna 1, 50018 Zaragoza
{clavel,cmahulea}@unizar.es.
†Computer Engineering Department, Gheorghe Asachi Technical Univ. of Iasi, Romania
dia.botez@gmail.com
‡Orthopedic Surgery Department, University Hospital "Lozano Blesa", Zaragoza, Spain
jcalbareda@salud.aragon.es

***Abstract*—Medical doctors working in hospitals should spend their working time taking care of the patients. However, in the Spanish hospitals, they also play the role of decision makers being involved, for example, in scheduling surgeries in operation rooms. Although some doctors might be doing this for a long time, which give them experience in such decisions, the efficiency of the operation room can be improved with the help of some technical solutions that are the result of advance research in planning and scheduling surgeries. Such technical solutions can also decrease the overtime hours and reduce the under/over utilization of some operating rooms. This paper's focus is to create such a solution that will help medical doctors to spend less time in managing tasks and have a better occupancy rate within the available capacity of the operation room. It will also decrease the stress level and allow the medical doctors to offer more of their time to healthcare. The contributions of this paper are: (i) a new heuristic algorithm for scheduling operations that do not require the use of any optimization tool, (ii) an in-built managing system for patients and operation rooms in the Operating Theater, and (iii) simulation results using real data in the proposed software tool. The created interface is intuitive and developed with direct feedback from doctors allowing different access level, for a better security of the medical data.**

***Index Terms*—Software solution, Operation room scheduling, Heuristic method**

I. Introduction

Spanish hospitals are an attractive case study because population of Spain is one of the oldest in Europe: 17.95% of the Spanish population is older than 65 years and the median age is 42.7 years [1]. In Spain, the average life expectancy is 81.8 years and the country also has one of the highest female life expectancy in Europe: 84.9 years. This aging population implies that the demand for surgical services is increasing and consequently, more patients have to be included in a surgery waiting list. Moreover, the *Operation Room* (OR) is one of the most expensive resources of the hospitals. Approximately 60% of patients need a surgery at some point during their hospital stay [2]. Surgical costs typically account for approximately 40% of the hospital resource costs [3], while surgeries generate around 67% of hospital revenues [4]. It is more and more necessary to have good planning and scheduling methods to improve the efficiency of the ORs. In [5] modular petri net are proposed for modeling healthcare system. In this work, techniques for the planning and scheduling of patients will be analyzed and compared. In order to situate the planning and scheduling problems of ORs three classical levels are considered in bibliography [6]:

1) *case mix planning* is a long term strategic planning that involves the hospital's mission and its translation into hospital resource capacity planning on the basis of highly aggregated information.
2) *master surgery schedule* is a medium term tactical approach that determines how much operating room time is assigned to different surgeon groups on each weekday. These time allocations are commonly referred to as time block booking.
3) *scheduling of patients* is a short term operational approach to fix the patients that should be operated in the next time blocks.

In Spanish hospitals, each group of surgeon has its own waiting list of patients. Moreover, time blocks of the OR are previously booked to each surgeon group. So, the problem to solve is the scheduling of patients from a waiting list to the time blocks (level 3 stated before). This assignment should maximize the use of the OR respecting the patients' order in the waiting list as much as possible. Moreover, the surgery durations are not deterministic and this uncertainty could result in uncomfortable situations for the medical management staff. So a minimum confidence level of not exceeding the working time for each time block should be guaranteed.

In this paper we propose a software based solution with a friendly interface to help doctors in the scheduling task. This software uses a specific heuristic algorithm for the scheduling which is proposed in this paper. In addition, the proposed software solution can be available in a Java based interface, all data being saved in a database.

The software solution will be initially used in the Orthopedic Department of the "Lozano Blesa" Hospital in Zaragoza, Spain. So, the simulations have been performed using real data from that department.

Software, such as GUIDE [7] or HEAT [8] have been proposed for the development and analysis of clinical pathways in order to control the effectiveness and efficiency of the medical

This work has been partially supported by UZ2018-TEC-06

interventions. However these software do not provide decision about the scheduling of patients.

There are other commercial software available for general scheduling which could be adapted to be used in a hospital. However, these software, in addition of having a high price, may not consider aspects related with the management of medical team, waiting list or ORs. Our proposed solution had been developed in collaboration with doctors of the ”Lozano Blesa” hospital. So, the concerns and preferences of doctors had been considered not only at the level of surgical scheduling, but also at the user interface level.

This paper is organized as follows. Sec.II shows the organizational structure in the studied department and introduces the scheduling problem. In Sec.III the proposed heuristic method is displayed while in Sec.IV the main use cases of the software tool are described. Some simulations showing the use of the tool are included in Sec.V and some conclusions are given in Sec.VI.

II. Organization structure and notations

The studied department is composed of doctors (specialists and residents) divided into five medical teams. Each surgeon has his own waiting list of patients. The patients belonging to the waiting lists of the doctors belonging to a given team, compose the waiting list of the team. Moreover, each team has a coordinator (a more experienced doctor) who is responsible for planning the patients on his team.

Two ORs are available per day for non-urgent surgeries in the studied department. Each of these ORs have an active schedule from 8:30am to 3:00pm (6.5 hours). The Orthopedic Department is organized in such way that during a given day each OR is used by a unique team. The head of the Orthopedic Department (who is a doctor as well) is responsible for assigning teams to the ORs. The assignment is made with a time span of two months, that is, all teams knows in advance the days in which they can use the ORs.

Currently, no automatic method exists for operation scheduling, so each team coordinator is guided by his own intuition and experience to schedule the surgeries. Normally, each team uses the ORs 3 times per week and the scheduling is made every 2 weeks, so the coordinator of the team should schedule the patients from the waiting list to the next 6 ORs (2 weeks). The main objective of this work is to provide a solution for surgery department's management, particularly for the patient scheduling task. It is important that the utilization of the OR does not exceed 3pm because in this case medical staff could lengthen their working day and the starting time of the next surgery session (15:00) could be delayed.

Let $\mathcal{S} = \{s_1, s_2, \dots, s_{|\mathcal{S}|}\}$ be the set of surgery types that can be performed in the considered hospital department and let $d : \mathcal{S} \to \mathbb{R}_{>0}$ be the *duration function*: $d(s_i)$ of the surgery s_i.

The total duration Td of an OR working day can be computed as the sum of: a) the delay respect to the starting time denoted as ds, b) the sum of all durations of the scheduled surgeries $d(s_i)$ and c) the duration of cleaning times between consecutive surgeries Ct. All these durations are considered random variables with normal probability density function (pdf). Their average and standard deviation values have been computed by using historical data from the hospital (for our study case, we use data of the last two years):

- Delay respect to the started time: $ds = N(10, 12)$ has an average of 10 minutes and a standard deviation of 12 minutes.
- Duration of surgery s_i: $d(s_i) = N(\mu_{d(s_i)}, \sigma_{d(s_i)})$ has an average of $\mu_{d(s_i)}$ and a standard deviation of $\sigma_{d(s_i)}$.
- Cleaning time: $Ct = N(20, 10)$ has an average of 20 minutes and a standard deviation of 10 minutes.

The probability of not exceeding working time is defined by the probability that the variable representing the total duration Td is less or equal than the duration of the OR available time denoted by X:

$$P(Td \leq X) \quad (1)$$

notice that Td is a random variable with normal pdf obtained by the sum of other normal variables above explained. A minimum confidence level Cl of not exceeding working time should be guaranteed by imposing:

$$P(Td \leq X) > Cl \quad (2)$$

The expected occupation rate of day j of an OR denoted as Or_j is computed considering the durations of the scheduled surgeries. Let us assume that $O_j \subseteq S$ is the set of surgeries scheduled in the OR working day j, then:

$$Or_j = \frac{\sum_{s_i \in O_j} \mu_{d(s_i)}}{X} \cdot 100 \quad (3)$$

III. Exact and heuristic solution methods

In this section different solution methods for the scheduling patients problem are discussed. First, some existing meta-heuristics solutions based on solving sequentially mathematical programing problems are mentioned and then, the new proposed heuristic algorithm that has been implemented in the tool will be described.

A. Exact solution methods based on mathematical programming

Two exact solution methods had been previously proposed for the scheduling problem in the studied department:

1) A mixed integer linear programing (MILP) problem was proposed in [9]. This model tries to obtain a given occupation rate Or of the ORs respecting as much as possible the order of the patients in the waiting list. However, it does not impose a minimum confidence level of not exceeding the available time.
2) A New Mixed Integer Quadratic Constraint Programing (N-MIQCP) problem was proposed in [10]. This model improves MILP by considering the maximization of the Or at the same time with guaranteeing a minimum confidence level of not exceeding the working time.

MILP and N-MIQCP problems have a high computational complexity, so in order to schedule a large number of OR working days it is necessary to do this sequentially, programming each time the next 3 OR working days. Moreover, an efficient commercial solver like CPLEX [11] is necessary to obtain the scheduling in a reasonable time.

B. Heuristic solution method

To avoid the use of an expensive commercial solver in the developed software solution, we propose an heuristic algorithm for the scheduling problem. Moreover, we observe by simulations that the obtained solutions are very similar to the ones obtained using N-MIQCP problem, but with a lower computational time.

The proposed algorithm schedules patients from the waiting list to the next m OR working days in a sequential way. The heuristic algorithm is inspired from list scheduling techniques [12], [13] and [14] , where the items to schedule are ordered according to a given priority. In our case, the items are the patients, and they are ordered according the inclusion date. The algorithm is based on the idea that the first patients on the waiting list should be scheduled in the first days. In order to reduce the possible combinations of surgeries, we classify the surgeries of the department depending on their average duration. According to this classification, the combinations to evaluate for scheduling are generated. The algorithm's input data is:

- The set of surgery types performed in the department, i.e., $\mathcal{S}$, and their average occurrence percentages.
- The number of OR working days to schedule, i.e., m.
- The duration of the OR working day, i.e., X.
- The minimum confidence level of not having overtime denoted as Cl.
- An ordered waiting list of patients $W = \{w_1, ..., w_n\}$ where, for each patient w_i, we know the preference order $po(w_i)$, the average duration of its surgery $\mu(w_i)$ [1] and the standard deviation of its surgery $\sigma(w_i)$
- t and β are two input parameters of the algorithm. t indicating the number of subsets in which the set of surgeries S is going to be divided while β is a parameter in the fitness function.

The algorithm is composed by 7 steps which can be divided in two parts: a) a previous data analysis and b) the scheduling of OR working days.

A previous data analysis: composed from first three steps running once at the beginning of the scheduling.

Step 1: Classify the surgeries of $\mathcal{S}$ in t disjoint subsets such that $S = \bigcup_{i=1}^{t} S_i$ and the average duration of all surgeries in S_i are less than the average duration of surgeries in S_j if $i \leq j$, i.e., $\mu(s_a \in S_1) \leq \mu(s_b \in S_2)$. Furthermore, the partition such that the expected number of surgeries belonging to each subset S_i in a real waiting list is same.

Step 2: Obtain the set of possible scheduling types, each type being defined by the subset to which the surgeries belong. For example, a possible scheduling type (st_i) could be $\{k, k, j\}$ and it is composed by two surgeries belonging to the subset S_k and another one belonging to the subset S_j. The set of all st combinations is composing the set SPS, where all elements of the set SPS satisfies the chance constraint of not having overtime. The chance constraints are computed by using the surgery with lower average duration from the sets S_i.

Step 3: Classify the patients on the waiting list. To each patient $w_i \in W$ we assign a certain type through $ty(w_i) = j_{\in\{1,2,...,t\}}$ according with the subset $(S_1, S_2, ..., S_t)$ to which the surgery belongs.

Scheduling of OR working day: composed by last four steps (4, 5, 6 and 7) are running sequentially m times, one for each OR to schedule.

Step 4: For each scheduling type $st_i \in SPS$, obtain a set SRS_i of real scheduling. Given a scheduling type, a real scheduling is composed by patients from the waiting list who have the same types of surgeries. For example, considering the scheduling type $st_i = \{k, k, j\}$ a real scheduling belonging to SRS_i would be composed by two patients w' and w" with $ty(w') = ty(w'') = k$ and other patient w"' with $ty(w''') = j$). As first patients in the waiting list should be scheduled first, the first real scheduling of type st_i founded in the waiting list is added to their corresponding SRS_i. Moreover, if there are other real scheduling of type st_i ending with the same patient preference order than the first real scheduling, they are also added in SRS_i.

Step 5: For each SRS_i evaluate the real scheduling and select the best one. First, the real scheduling that do not fulfill the chance constraint (2) are removed from SRS_i. After this, for each real scheduling in SRS_i, the expected occupation rate Or given by (3) and the average preference order Apo are computed. According with these values, each real scheduling of SRS_i is evaluated by the next fitness function (H):

$$H = (Apo - Min_{Apo}) * \beta + (Max_{Or} - Or), \qquad (4)$$

where Min_{Apo} is the minimum Apo value between all real scheduling in SRS_i and Max_{Or} is the maximum Or value between all real scheduling of SRS_i. The real scheduling with the lower fitnesses value of each SRS_i is selected.

Step 6: Between the previously selected scheduling of each SRS_i, the final scheduling decision is chosen. Using again the fitness function (4) the previously chosen scheduling (Step 5) are evaluated. The scheduling with lower fitness function is assigned to the next day.

Step 7: Remove scheduled patients. The patients scheduled in Step 6 are removed from the waiting list.

The parameter t in *Step 1* fix the number of surgery types. On the other hand, the fitness function (4) is composed from two terms. The first one is related with the objective of respecting the preference order of the patients while the second one is related with the objective of maximizing the occupation rate of the OR working days. These two terms are balanced by the value of the parameter β. After some simulations, has been observed that a value of $t = 3$ and $\beta = 2.6$ are appropriate in the hospital department. Note that we are assuming that

[1] To simplify the notation, the average duration and the standard deviation of the surgery corresponding to patient w_i are denoted as $\mu(w_i)$ and $\sigma(w_i)$ respectively.

all OR working days have the same daily working time X. In other case, step 2 should be executed one time for each different working time X_i. Consequently, in Steps 4 and 5 the SPS_i corresponding with the duration X_i of the currently OR working day $i < m$ should be used.

C. Comparison between exact heuristic solutions

In this subsection, the scheduling obtained by solving the N-MIQCP [10] problem is compared with the scheduling obtained by using the proposed heuristic method. One surgical team of the Orthopedic Department in the "Lozano Blesa" hospital is considered. Moreover, the length of the simulation is fixed to one year (52 weeks). So, assuming that the mentioned team uses 3 time blocks per week, 156 time blocks are scheduled in the simulation. The duration of each one of these time blocks is assumed to be 6.5 hours, from 8:30 a.m to 15:00 p.m. Due to the large size of the instances, the scheduling obtained with the N-MIQCP problem is performed by using receding horizon strategy [15], [16]). This means that problems with a shorter time horizon (1 week = 3 time blocks) are solved sequentially to compose the final scheduling.

The simulations have been performed by using the IBM ILOG CPLEX Optimization Studio which is often referred as CPLEX [11]. A computer with an Intel Core i3 and 4 GB of memory has been used.

The initial waiting list is composed of 100 patients and it is updated each week assuming the arrival of new patients. The type of the surgeries and the weekly arrival rate are generated considering data from the Orthopedic Department in the "Lozano Blesa" hospital in Zaragoza.

Considering different values of minimum confidence level (70%,75%,80%), 20 replications of one-year scheduling have been performed using the exact and heuristic methods. The average results of the scheduling have been compared from different points of view: utilization efficiency, the confidence of not exceeding the total time, order of the patients and computation efficiency.

Tab. I shows the results obtained. The first column imposes the minimum confidence level, this is an input parameter of the problem. The second column indicates the model used. The third and fourth columns are related to the utilization: occupation rate and the number of treated patients respectively. The fifth column shows the confidence level of not exceeding the working time. The sixth column is a parameter (Ω) [10] related to the order of the patients. The lower Ω is, the better the order of the patients in the scheduling is. Finally, the seventh column shows the computational time to schedule one year (156 time blocks).

The results show that similar utilization efficiency of the ORs is achieved with both, the N-MIQCP and the heuristic method. Maybe, a little better occupation rate is obtained and a few more patients are treated using the N-MIQCP problem. However, the confidence level is better using the heuristic approach and the scheduling obtained is respecting more the order of the list. Finally, it can be checked that the computational time is much lower using the heuristic approach than using the N-MIQCP problem. The same computation time is used for performing one-year scheduling using the heuristic method independently of the minimum confidence level (time $\sim$ 58 [s]). However, the computational time using the N-MIQCP problem increases (394, 685 and 1250 [s]) when the minimum confidence level increases (70, 75, 80 [%]). Notice, that the computational time using N-MIQCP problem increase exponentially with the time horizon. In this case an horizon of 3 time blocks has been chosen in the receding horizon strategy. However, with an horizon of 6 time blocks, some instances can not be solved using a computer with an Intel Core i3 and 4 GB of memory, after 6 hours of computation

So, the heuristic method allows obtaining similar scheduling with a much lower computational time and without the necessity of using a software solver for mathematical programming problems.

IV. SOFTWARE TOOL

In this section, the software tool will be presented, along with the advantages and features that it brings. The software tool's focus is to help doctors to spend less time in managing tasks and have a better occupancy rate within the available capacity. It will also decrease the stress level and allow the doctors to offer more of their time to patients.

This application is made to be used by the medical staff of a Surgery Department. Its interface is designed to help managing the medical teams and patients in a more clear, efficient and easy way. It also gives different access level to every user for providing a safe environment to work on and a better security of the medical data. The application will be installed in hospitals local network. This application uses a database and its interface is available in Spanish. English can also be provided for non-Spanish Hospitals. Every modification made will be saved and ready to be accessed and used in the current session or a future one. The software also allows concurrent users.

After the user has logged in, depending on the access level, the application can be used for:

1) Managing medical teams
   - Add/remove doctor from team or medical teams;
   - Move doctor to another medical team;
   - Change the medical leader of the team or the team name.
2) Managing patients
   - Add/remove/update patients or patient details;
   - Add/remove/change surgeries from patients medical history;
   - Manually schedule a patient to a certain medical team and doctor and to a certain day;
   - Unschedule a patient that cannot reach the hospital in the assigned day.
3) Managing operating rooms
   - Add/remove operating rooms from the operating theater;

TABLE I
SIMULATION RESULTS OF ONE-YEAR SCHEDULING SIMULATION. N-MIQCP VS HEURISTIC METHOD.

Minimum Confidence[%]	Model	Utilization		Confidence Level	Order of Patients	Computation
		Occu.[%]	Treated Patients	Avg. Confi.	Ω	Time [s]
70	**N-MIQCP**	78.3	439	77.3	406.1	394
	Heuristic	77.7	437	78.6	318	57
75	**N-MIQCP**	76.7	430	81.5	343.1	685
	Heuristic	76.1	428	82.7	336	57.4
80	**N-MIQCP**	74.5	415	86.3	313.2	1250
	Heuristic	73.9	414	86.9	353	58.2

- Schedule a medical team in a certain day for each operating room;
- Change data in the current available time table.

4) Planning and scheduling patients from a certain waiting list for a given number of working days.
5) Adding the information of the performed surgery for each patient to whom the scheduled surgery is completed.

Once a patient is unscheduled because of unavailability and if the patient is not re-scheduled manually, the priority of the patient on the next planning will be the highest.

The application had been built on 4 independent tracks:

1) The *back-end implementation* has different internal packages. One package for each user interface available, one for the common classes (e.g., Patient, Doctor, Surgery or Database Connection/Query), one for defining and computing the heuristic model and another one for the main-files. The last package contains the entry-point class of this application. At this step, the user can choose the language to use during the next session, in case of multiple languages available. If the application cannot connect to the database, an error message will be prompted. The session is initialized with a login form in the language chosen by the user (or the default one). If the authentication failed, an error message will be prompted. After a successful log-in, the application will start with the users access level.
2) *Database*: The application has a local relational database (within the hospitals intra-net). Along with he medical data, the database also contains information about all users allowed to connect, their access level and their hashed passwords. Access levels are described as follows:
   - *head of department*: can schedule teams for the available operating rooms and manage the medical staff;
   - *coordinator (team leader)*: can create schedules for his team's waiting list;
   - *medic*: can add/remove new/performed surgeries in the department;
   - *assistant (nurse)*: add/update patients in the database;
3) *User interface*: main menu can be seen in 2;
4) *Heuristic scheduling algorithm*: the heuristic scheduling algorithm and some simulation results has been presented in the previous section.

The user interface contains 6 main panels. They are split as follows:

- *Medical teams*: all the actions about medical staff can be performed in this panel by the head of department. All other users can only see the information.
- *Patients*: all users can see/add/update/remove a patient or patient's details.
- *Surgeries*: the set of elective surgeries that can be performed in the department are added in this panel. Existing surgeries can be updated or removed by any user with access level "Medic" or above. The average duration and the standard deviation of any surgery can be consulted here.
- *OR timetable*: all users can see the timetable for each operation room, but only the head of department can edit data from this panel.
- *Schedule*: all coordinators have access to this panel for creating schedules from team's waiting list.
- *User*: every user can change it's own password. The head of department can edit other user's credentials. This panel is also used for ending the current working session by logging out the user. The application will not close if the user does not logout.

The access level of each user in the tool is accordingly to the position they have in the medical department. The head of department has admin rights. This user can create other users for the coming doctors in the departments and also delete credentials for the doctors that leave the department. When a doctor changes the medical team or team coordinator is changed, the head of department has to update the database accordingly. The user can remove a doctor from the database and application only if the current doctor is not a team leader. For deleting a team leader, another doctor from the same team, must take the team leader position. The waiting list of a doctor that changes the team or position will not be affected by the change. Only when a doctor leaves the department and the username is removed from the database, all the patients from its waiting list will be transfered to coordinator's waiting list. Patients can be moved manually from a doctor's waiting list to another. According with the current structure of the above mentioned hospital, a coordinator can perform scheduling for his team's waiting list for a specified number of days while the other doctors from the team have only read access.

After a successful login, the first panel that every user sees, regardless the access level, is ”Medical Teams” panel. It allows read access to all users. Here, a user can see each medical team, its name and coordinator, as well as the next operating rooms assigned to the team. Also in this section, there are two separate sub-panels that allows the Head of Department to manage the medical staff as described above.

The first sub-panel in the ”Patients” section, offers a view to the waiting lists (either for team or doctor) along with few details about the selected patient and its history. When adding a new patients, some personal data are required. If there is already someone in the database that matches the details provided for the new patients, an error message will be prompted along with the existing patient’s ID. A patient can be searched in the database by name and birth date or by ID. When updating a patient’s data, there are 3 actions that can be done: (1) Update patient details (gender, birth date, remarks), (2) Add a new surgery for the patient. This surgery will be added to patient’s medical history and will be scheduled as soon as possible, depending on the size of the waiting list, (3) Update existing surgery for patient. Within the last action, a doctor/medical team can be changed or patient can be manually scheduled/unscheduled. If the patient chooses not have the surgery, it can be removed from its medical history. When selecting a patient from the list, more information (such as: surgery, doctor, admission date, remarks, etc) will be shown in the defined area in the bottom of the panel.

In the ”Surgeries” panel, action buttons are enabled for all users with access level grater or equal with ”MEDIC”. They offer the possibility to add new surgery, update the timings for existing ones or remove surgeries from database. Removing surgeries can only be done if there is no doctor assigned to it and all patient admitted for this surgery have status ”performed”.

”OR timetable” panel contains all the available operating rooms in the operation theater for non-elective patients. Each sub-panel contains daily bookings for medical teams. These panels are enabled for editing only for the Head of Department. The other users have read access to the bookings from any operating rooms. Each operating room has a default name that can be changed. A team can be booked in the selected operating room only if:

- there is no other team booked on the desired date;
- team is not booked on the desired date in another operating room.

Starting and ending availability time of the operating room can be chosen after selecting the team and date. After all the fields are filed correctly (including: ending time grater than starting time), a new booking in the operating room’s timetable can be added. There is no edit option for existing bookings. To do so, the wrong booking must be deleted, then create another one. To delete an existing booking, a pop-up window will request for the date of the booking that should be deleted. It also specifies the accepted date format. To avoid deleting the wrong booking, a confirmation message will be prompted. According wit the hospital’s current management, there are two types of operation rooms available: Morning and Afternoon. The difference between them is only about the time availability. Regardless the type, an operating room can be deleted only if there is no team booked from the current date forward and if it is not the only one left.

The ”Schedule” panel, has three sub-panels:

- first sub-panel is available only for coordinators. Here, coordinators can create the schedule from the waiting list according with the available number of days in the ORs; The fields are automatically pre-filled with the maximum value for the number of days to schedule7 and with an acceptable minimum confidence level. The maximum number of days to schedule is computed from the timetable of all ORs from the operation theater.
- second sub-panel is used for seeing the schedule of a certain OR/team for a certain day. This sub-panel also shows the scheduled patients, the total time available for the OR and the estimated occupancy time;
- third panel is used for setting surgeries as completed. This step is needed because in some cases, the patient does not show up for the surgery although he confirmed the presence before.

Fig 2 shows a summary of a performed schedule. Here, the result of each scheduled OR working day includes: the date, the ID of the scheduled patients, the specific OR, the starting/ending time, the expected occupation rate and the confidence level of not exceeding the total time. Table II shows the complete schedule obtained.

The main achievements for a medical department of using such an application are:

- optimize resource usage in an operation theater by maximizing the occupation rate of each operating room;
- establish a minimum confidence level of not overcoming the total day time in each OR;
- automatic update of the estimated duration of all available days for OR;
- prevent human mistakes (such as multiple allocation of a team, for the same day, in different operating rooms)
- more time for doctors to take care of patients;
- keep track of newer and older patients in the waiting lists as well as previous surgeries of known patients.

V. Simulation using the proposed software solution

In this section, the facilities and advantages of using the tool in the patient scheduling task are shown. To do this, using the software described in Sec. IV, some ORs which have been previously booked for a medical team are scheduled.

Fig. 1 shows the user interface of the panel “Patients” in which the subpanel “See patients list” is selected. Here the waiting list of patients of the medical team 1 can be consulted. This list is composed by 50 patients and by selecting a patient, more information (such as: surgery, doctor, admission date, remarks, etc) will be shown in the defined areas of the panel. Using the software tool the next 6 ORs working days previously booked for the medical team are scheduled.

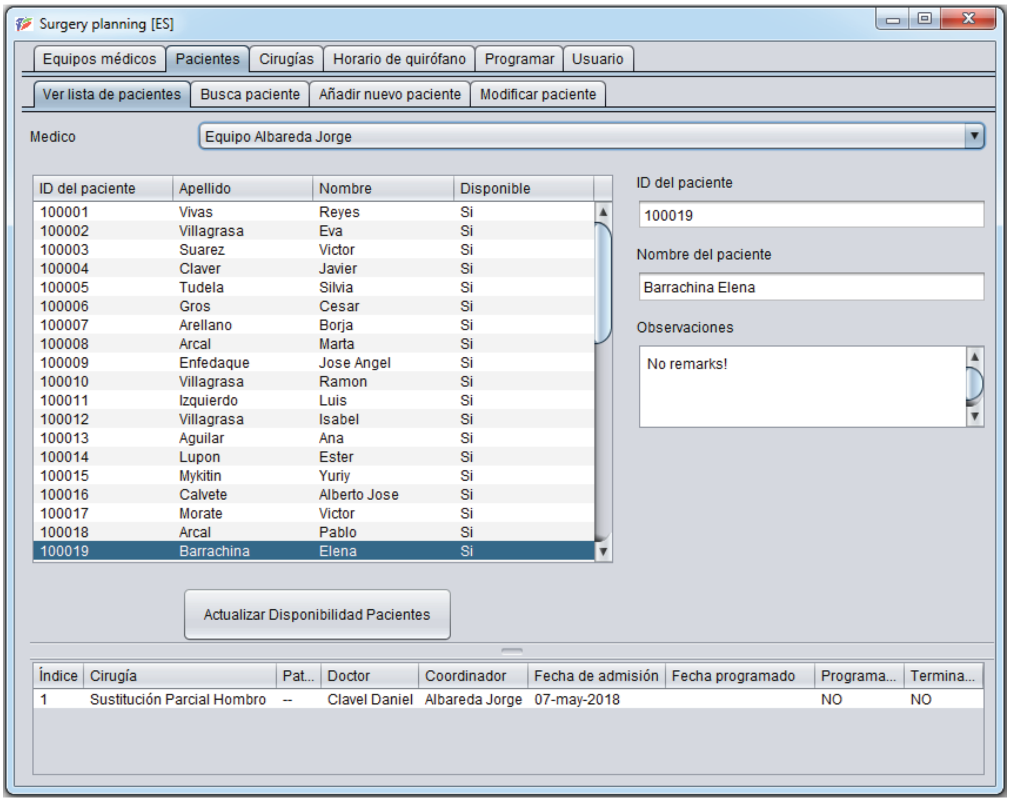

Fig. 1. User interface: panel of "Patients"

Fig. 2 shows the user interface of the panel "Schedule" in which the subpanel "Create a schedule" is selected. In the upper part, there are two pre-filled fields for the algorithm's input parameters - the number of days to schedule and the minimum confidence level. The team for which the user will create the schedule, can also be selected in the upper part. In this case 6 ORs are going to be scheduled for the Team 1 (Jorge Albareda) with a minimum confidence level of 70%. In the lower part, the schedule's output can be seen. The full scheduling of the selected team is showed in table II.

TABLE II
COMPLETE SCHEDULING OBTAINED BY USING THE TOOL FOR A TEAM (SEE FIG. 2)

Date [yyyy-mm-dd]	ID of patients	Total time [min]	Ocupation Rate [%]	Confidence level [%]
2018-05-14	1-2-5	390	78.46	74.04
2018-05-15	4-6-9	390	77.95	74.45
2018-05-17	3-8-15	390	79.23	74.21
2018-05-22	7-11-17	390	77.95	76.86
2018-05-23	10-12-14	390	74.62	88.5
2018-05-25	14-16-18	390	74.62	88.5

In the second subpanel ("See a scheduling") of the "Scheduling" panel more details about the patients scheduled in each OR working day can be consulted: complete names, kind of surgeries, their expected duration, the doctor who was assigned and the admission date.

Finally, in third subpanel ("Define a surgery as completed") of the "Schedule" panel, it is possible to set a patient as completed once their surgery has been performed. The real duration of the surgery is introduce in the tool and it will be used to update and customize the average duration and standard deviation.

VI. CONCLUSIONS

This paper is focused on creating an easy-to-use software solution for surgeons and auxiliary staff (e.g., assistants) that will be used for planning and scheduling of non-elective (non-urgent) patients. The planning and scheduling of patients is based on a heuristic algorithm that will schedule a given list of patients. The objectives of the algorithm are: (1) maximize the occupation rate of each OR without exceeding the total available time, (2) respect the order of patients as much as possible. The software solution adds other objectives as well: (3) reduce time spent for management tasks, (4) prevent human mistakes that can occur in scheduling patients or assigning medical teams in ORs, (5) possibility to use common database for each hospital to avoid multiple medical histories of one patient, (6) ease of medical staff management and the changes that might occur over the years, (7) secure data access given by different access levels in the proposed software solution. Moreover, the tool will be used in the Orthopedic surgery Department of the "Lozano Blesa" Hospital in Zaragoza and as result of this use, two news functionalities arises: (8) compute

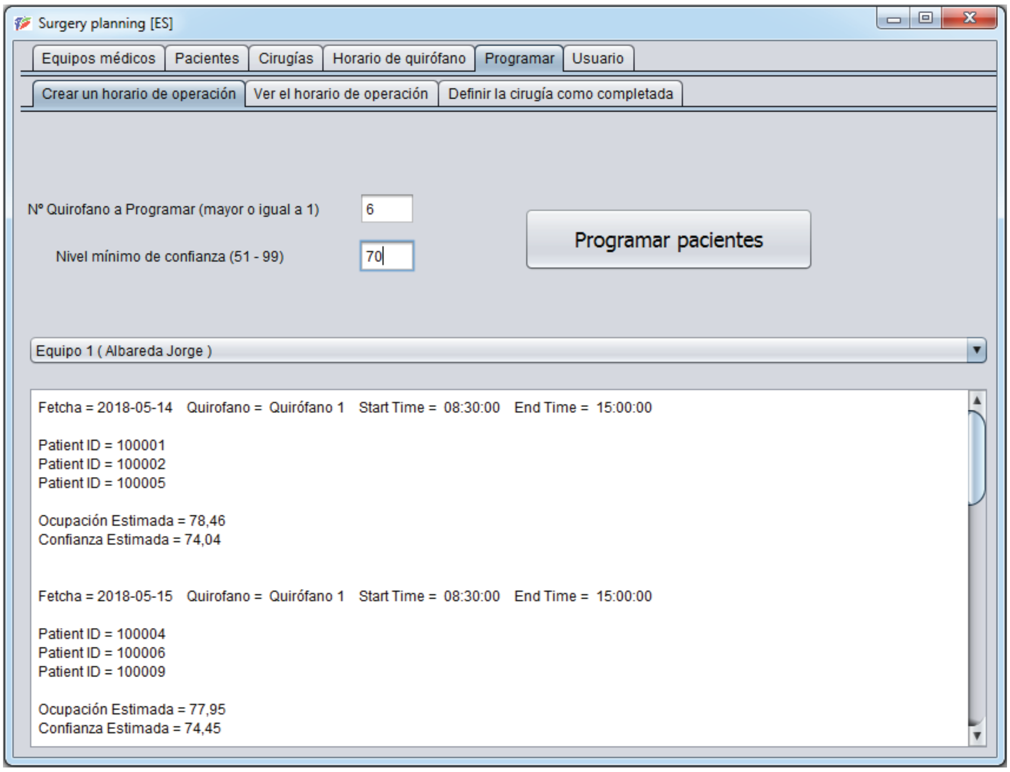

Fig. 2. User interface: panel of "Scheduling"

the number of ORs days necessaries to scheduled all patients in the waiting list. In this way, an approximate surgery date could be given to patients when they are included in the waiting lits. Moreover, using the tool in several surgical services of the hospital, a proper distribution of the surgical resources (ORs) to surgical services could be made depending on the needs. (9) A post analysis of the surgical activity in each OR. Including the starting and ending time of each performed surgery, a new panel comparing for each OR the real surgical activity with the expected one will be included. In this way, it will be easier to identify the cause when the working time will be exceeded.

References

[1] "The world factbook. [https://www.cia.gov/library/publications/the-world-factbook/]." [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook/

[2] *OECD, health data 2005 - statistics and indicators for 30 countries*, 2005.

[3] A. Macario, T. Vitez, B. Dunn, and T. McDonald, "Where are the costs in perioperative care?: Analysis of hospital costs and charges for inpatient surgical care," *Anesthesiology*, vol. 83, no. 6, pp. 1138–1144, 1995.

[4] R. Jackson, "The business of surgery. Managing the OR as a profit center requires more than just IT. It requires a profit-making mindset, too." *Health Management Technology*, vol. 23, no. 7, pp. 20–22, 2002.

[5] C. Mahulea, L. Mahulea, J. M. García Soriano, and J. M. Colom, "Modular petri net modeling of healthcare systems," *Flexible Services and Manufacturing Journal*, vol. 30, no. 1, pp. 329–357, Jun 2018. [Online]. Available: https://doi.org/10.1007/s10696-017-9283-9

[6] Z. Y. Abdelrasol, N. Harraz, and A. Eltawil, "A proposed solution framework for the operating room scheduling problems," in *Proceedings of the world congress on engineering and computer science*, vol. 2, 2013, pp. 23–25.

[7] S. Wanderer, D. Stavrou, G. Lypowy, C. Latter, and C. Caudle, "The guide architecture for implementation of clinical practice guidelines."

[8] L. Parrilla, J. Garca, J. Albareda, and C. Mahulea, "Heat: A tool to develop, analyze and monitor clinical pathways," in *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, May 2017, pp. 186–191.

[9] D. Clavel, C. Mahulea, M. Silva, and J. Albareda, "Operation Planning of Elective Patients in an Orthopedic Surgery Department," in *ETFA'2016: 21st IEEE International Conference on Emerging Technologies and Factory Automation*, Berlin, Germany, September 2016.

[10] D. Clavel, C.Mahulea, J. Albareda, and M. Silva, "Robust Scheduling of Elective Patients under Block Booking by Chance Constrained Approaches." *Technical Report RR-18-01, Universidad de Zaragoza*, 2018. [Online]. Available: http://webdiis.unizar.es/∼cmahulea/papers/rr_2018.pdf

[11] IBM. (2016) Ibm ilog cplex optimization studio. software. [Online]. Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/

[12] D. S. Johnson, "Near-optimal bin packing algorithms," Ph.D. dissertation, Massachusetts Institute of Technology, 1973.

[13] M. G. N. Klement, N. Grangeon, "Medical Imaging : Exams Planning and Resource Assignment: Hybridization of a Metaheuristic and a List Algorithm." in *10th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2017)*, no. 5, Porto, Germany, February 2017, pp. 260–267.

[14] H. Arabnejad and J. G. Barbosa, "List scheduling algorithm for heterogeneous systems by an optimistic cost table," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 682–694, March 2014.

[15] E. Camacho and C. Bordons, *Model Predictive Control*, ser. Advanced Textbooks in Control and Signal Processing. Springer London, 2004.

[16] I. Jurado, J. Maestre, P. Velarde, C. Ocampo-Martinez, I. Fernndez, B. I. Tejera, and J. del Prado, "Stock management in hospital pharmacy using chance-constrained model predictive control," *Computers in Biology and Medicine*, vol. 72, pp. 248 – 255, 2016.