Menemui Matematik (Discovering Mathematics)
Vol. 41, No. 2: 46 - 56 (2019)

Local Search Heuristic For Elective Surgery Scheduling Considering Patient Urgency

N. S. A. Rashid [1,a], N. A. A. Rahmin [*,1,b], W. J. Leong [1,c] and M. A. Nazihah [1,d]
[1] *Mathematics Department, Faculty of Science, University of Putra Malaysia, 43400, Serdang, Selangor.*
[a]nurshafiqahbteabrashid@gmail.com, [b]aliza@upm.edu.my, [c]leongwj@upm.edu.my, [d]nazihah@upm.edu.my
*Corresponding author

ABSTRACT

This paper study a surgery scheduling and surgeon assignment problem in operating rooms focusing on elective patients with different urgency. A long waiting time can increase the patient's urgency and lead to health complication. Our aim is to maximize the sum of the urgency values assigned to each surgery. An integer linear programming model is referred to solve the problem. As the model consume a high computational time to solve for a large-scale instance, we developed a local search algorithm based on a simple heuristic to deal with the problem. To test the efficiency of our proposed heuristics, we compare the solutions of integer linear programming model with the heuristics. The results show that solutions obtained by the local search algorithm are good quality and has significantly reduce the computational time even when considering more surgeries in the waiting list.

Keywords: Surgery Scheduling, Surgeon Assignment, Heuristic, Local Search, Integer Linear Programming.

INTRODUCTION

In health care sector, the demand in providing an efficient health services are getting higher each day. The operating theatre contributed to most of the hospital's profits despite its high costs. Therefore, the planning and scheduling of operating theatres must be efficient enough to prevent from inefficiency of the resources. This paper considered the elective surgery scheduling with the priority based on the urgency level given. A waiting list consists of number of patients who need to undergo their treatment will be scheduled. Each patient in the waiting list will be categorized into different urgency level depending on their health condition. Patients are scheduled based on their urgency level into any available days and rooms. Higher urgency patient need to be scheduled first because a long waiting time can cause a complication to their health condition and gives dissatisfaction to the patients. After the surgeries were scheduled, two or more surgeons need to be assigned to these surgeries in order to make sure all patients are treated according to the schedule. This paper focused on surgery scheduling and surgeon assignment problem with the aim of maximizing the sum of the urgency value assigned to each patient. An integer linear programming (ILP) model (Pradenas et al., 2012) is referred to develop an improvement heuristic to solve the surgery scheduling problem. A simple heuristic (SH) will be as the basic for the improvement heuristic. Based on the result, we want to compare the computational time and the quality of the solution between the model and the proposed heuristics. A backtracking heuristic (BH) is used to solve the surgeon assignment problem.

Operating theater scheduling can be divided into two categories which are elective surgery and emergency surgery. Elective surgery can be scheduled in advance while emergency surgery arrives unexpectedly on the day of the surgery and must be scheduled as soon as possible. In our case, we will only focus on scheduling elective surgeries only. Cardoean et al. (2010) provides a review on the recent operation research on operating room planning and scheduling. They classified the problem into various performance measures such as waiting time of patients, utilization of operating rooms, cost savings and make span.

There is a lack of literature considered patient priority in the surgery scheduling problem. Daiki and Yuehwern (2010) addressed a scheduling problem of elective surgery with different patient priorities with the aim of minimizing the cost. Patients are classified into urgent, semi-urgent and non-urgent according to their health condition. They want to study the effect of patient priority on the surgery scheduling problem. Tanfani and Testi (2010) dealt with the same problem. They want to solve a Master Surgical Schedule Problem (MSSP). A set of patients with priority score (depends on the waiting time and urgency status of each patient) are selected to be operated on in the operating rooms. They need to assign the patients into different wards with given number of operating rooms during the planning horizon. Ergin (2013) study the scheduling and rescheduling of the elective patients in the operating room. The reason why there is rescheduling of the elective patient is because the sudden arrival of the emergency patient.

Another significant study in the surgery scheduling problem is involving the operating room utilization and patient satisfaction. Yu et al. (2010) divide the patients into different priorities according to their condition. The aim of this study is to maximize the patient satisfaction by considering their priorities. An available date and room is assigned so that the patients are scheduled as soon as possible. A weekly surgery scheduling problem is addressed by Pradenas et al. (2012). Each of the patients is associated with urgency level to indicate their health status. The aim of this study is to maximize the summation of the urgency level in order to satisfy the patient needs. They also solve the surgeon assignment problem where surgeons are assigned to the surgery scheduled before. Guillermo et al. (2017) study the surgery scheduling problem with the aim of maximizing the operating room utilization. A prioritize list is used to select the order of patients to be operated on. The author used the approach from Testi et al. (2006) where the prioritization is measure based on two factors which are patient biomedical category and the number of waiting days before treatment. Joonyup et al. (2018) optimize the assignment of surgeries by considering the patient health condition. They assume that increase in waiting time can affect the patient health condition. Therefore they assigned sicker patients to have their surgery first without increase the risk of other patients.

Based on the literature discussed above, Pradenas et al. (2012) tackle both of the surgery scheduling and surgeon assignment problems. In this paper, we refer to their model and solve the same problem with another methodologies to improve the computational time and quality of the solution.

In order to discuss a methodology to solve the surgery scheduling problem presented in this paper, we introduced the Local Search (LS). LS is commonly used in the scheduling problem to reach a better solution and close to an optimal solution. There are few studies which used LS in the surgery scheduling problem. Vancroonenburg et al. (2015) study a multi-day operating theater (OT) scheduling problem with the aim of maximizing the efficiency. A mixed integer programming (MIP) model is developed to maximize the number of surgeries scheduled and minimize the number of operating rooms (OR) need to be opened for surgical cases. The LS algorithm is developed to deal with the inefficiency of the MIP model for large-scale instances. Mateus et al. (2018) develop a LS to deal with elective surgery scheduling problem since the mathematical model is difficult to solve and too time consuming. This work is based on the problem studied by Marques and Captivo (2017) where they proposed a mixed integer linear programming (MILP) model to solve the problem with the aim of minimizing the waiting time and maximize the number of surgery scheduled. Moosavi and Ebrahimnejad (2018) propose a multi-objectives model and heuristic algorithm to solve the elective surgeries scheduling with upstream and downstream units. A mixed integer programming based LS neighborhood (MIP-based LNS) algorithm is applied to deal with the robustness and large-scale instances.

According to Mateus et al. (2018), the advantage of using mathematical models to solve a real case problem is that we possibly can obtain an optimal solution for the problem. However, the downside of this method is that it can be too hard to solve or taking too much time. Heuristics are a good way to find good solutions quickly. In order to solve the surgery scheduling problem, we propose a LS heuristic to deal with the inefficiency of the ILP model. A simple heuristic is used as the basic for the LS heuristic. This LS will be as our starting point in developing a metaheuristic algorithm in the future for more complex problem. Several set of generated data is used to test the performance of the proposed heuristic.

The remainder of this paper is as follows. Section 2 formulated the studied problem as ILP and explained the proposed algorithms. Section 3 deals with the computational experiments and finally, Section 4 concludes the paper.

METHODOLOGIES

This section will discuss the methodologies for the surgery scheduling and surgeon assignment problem. An ILP model is used for the surgery scheduling problem. A simple heuristic (SH) algorithm is developed as the basic for the LS. We used LS to improve the schedule. A backtracking heuristic is applied to solve the surgeon assignment problem.

SURGERY SCHEDULING PROBLEM

Based on the information from Pradenas et al. (2012), there are few procedures for the operation room system.The first one is the daily availability of each operation room is limited to 8 hours usage per day from 8.00 AM until 4.00 PM. The duration of each surgery is estimated from the historical data obtained from the hospital. An urgency indicator value from 1 to 4 is given to the patients where 4 indicates greater urgency.

MATHEMATICAL MODEL

The table below are the parameters used in the ILP model:

Parameters	
i	Index for surgery, where $i \in \mathbb{Z}^+$
d	Index for days
r	Index for operating rooms
esp	Surgery speciality
cir	Number of surgeons available for each surgery speciality, esp
Urg_i	Indicator of the urgency of surgery, i.
Dur_i	Estimated duration of surgery, i.
Gr_i	Number of operation room group where i is performed.

Day_{gr_t}	Number of days to program for operation room group, t.
OR_{gr_t}	Number of rooms available in operation room group, t.
S_{gr_t}	Number of surgeries to be programmed in operation room group, t.
Dur_{max}	Maximum time available for each room in single day.
$S_{min\_r\_d}$	Minimum number of surgeries to be programmed in each room, r on each day, d.
$Beds_{d_t}$	Number of beds available for operated patients in group, t on each day, d.
esp_{cir}	Surgery speciality with a cir number of surgeons.
$n_{esp_{cir}}$	Number of surgeries programmed that belong to esp_{cir}.
$t_{esp_{cir}}(i)$	Set that indicates the surgeries, *i* selected that pertain to surgery speciality esp_{cir}.

The ILP model involving the parameters in the above table are formulated as follows:

$$\max \sum_{d=1}^{Day_{gr_t}} \sum_{r=1}^{OR_{gr_t}} \sum_{i=1}^{S_{gr_t}} Urg_i \times x_{ird} \tag{1}$$

$$s.t{:} \sum_{i=1}^{S_{gr_t}} Dur_i \times x_{ird} \leq Dur_{max} \quad \forall r = 1, \dots, OR_{gr_t}, \forall d = 1, \dots, Day_{gr_t} \tag{2}$$

$$\sum_{d=1}^{Day_{gr_t}} \sum_{r=1}^{OR_{gr_t}} x_{ird} \leq 1, \qquad \forall i = 1, \dots, S_{gr_t} \tag{3}$$

$$\sum_{i=1}^{S_{gr_t}} x_{ird} \geq S_{min\_r\_d} \qquad \forall r = 1, \dots, OR_{gr_t}, \forall d = 1, \dots, Day_{gr_t} \tag{4}$$

$$\sum_{r=1}^{OR_{gr_t}} \sum_{i=1}^{S_{gr_t}} x_{ird} \leq Beds_{d_t}, \qquad \forall d = 1, \dots, Day_{gr_t} \tag{5}$$

$$\sum_{i=1}^{n_{esp_{cir}}} x_{t_{esp_{cir}}(i)rd} = 0, \qquad \forall d = 1, \dots, Day_{gr_t},\ \forall r \neq 1,\ for\ t_{es\ \ cir} \leq 4 \tag{6}$$

$$x_{ird} = \{0,1\}, \qquad i = 1, \dots, S_{gr_t}, r = 1, \dots, OR_{gr_t}, d = 1, \dots, Day_{gr_t} \tag{7}$$

Objective function (1) maximizes the sum product of the decision variable and the urgency value of each surgery scheduled. Constraint (2) is the time constraint where all surgeries need to be done within the time frame 8.00 AM to 4.00 PM which is 480 minutes per operation room, r in day, d. Constraint (3) ensures that each surgery is assigned only once at a given time per operating room. Constraint (4) represent the minimum number of surgeries that can be performed in each operation room, r. Constraint (5) is the capacity constraint where all patients will have recovery bed after the surgery. Constraint (6) is the surgeon constraint where for every surgery scheduled, there must be four or less surgeons ($cir \leq 4$) and surgeon that performed only in room 1, giving value of 0 for any possible assignment in other rooms. For our research, we only considered to assign at least two surgeons for each surgery which is lead surgeon and assistant surgeon. Lastly, constraint (7) is the decision variable used in the model where,

$$x_{ird} = \begin{cases} 1; & \text{if surgery, } i \text{ is performed in room, } r \text{ on day, } d \\ 0; & \text{otherwise} \end{cases}$$

SIMPLE HEURISTIC

As the number of patients increases, ILP becomes time consuming. Therefore, we proposed a heuristic to solve the problem. A simple heuristic (SH) is developed to schedule the surgeries based on their urgency value. We want to make sure patients with critical condition will get their treatment as soon as possible. An urgency value will determine the assignment of the surgeries to the operating rooms on each days. The SH start by sequencing the waiting list. We sequence the patient with high urgency value over patient with lower urgency value. If the urgency value is same, we will sequence the patient based on their duration in increasing order. Finally, we will scheduled the patients based on the list.

Algorithm 1 explains the SH procedure for surgery scheduling problem. In Step 1, we will choose two surgeries, i and j. If the urgency value of surgery i is lower than surgery j, we will swap the position of the surgeries (Step 2). But if the urgency value for both surgery i and j are same, we will check the duration of surgery (Step 3). If the duration of surgery i is higher than surgery j, we will swap the position of the surgeries. This algorithm will stopped when all the surgeries in the list are considered. After we have obtained the initial list, we will proceed to Step 4 where all the surgeries will be scheduled to room, r on day, d. In Step i), we declare all surgery, i with System(i)=0 to indicate the surgery not yet be scheduled. After that we will choose surgery, i to be scheduled in room, r on day, d (Step 4(ii),4(iii),4(iv)). In Step 4(v), for every surgery, i with System(i)=0, we will calculate the summation of OR time until it is less than or equal to Dur_{max}. If it satisfy the condition, we will add the surgeries to room, r on day, d and change the status of System(i). It will continue until all surgeries are considered and all rooms and days are occupied.

Notations used in the Simple Heuristic Algorithm	
i	Index for surgery, where $i \in \mathbb{Z}^+$
d	Index for days
r	Index for operating rooms
N $_{Day}$	Total days to schedule the surgeries
N $_{Room}$	Total number of operating rooms
$Urg(i)$	Urgency for surgery i
$Dur(i)$	Estimated duration for surgery i
Dur_{max}	Maximum OR time for each room, r in day, d
$totalDay(d)Room(r)time$	Sum of the OR time for day, d in each room, r.
System(i)	If System(i)=0, surgery i is not selected to be scheduled, otherwise surgery, i is seleced to be scheduled.

Algorithm 1: Simple Heuristic	
Step 1	Set surgery $i = 1$ and $j = i + 1$.

Step 2	If $Urg(i)\quad Urg(j)$
	then swap the surgeries;
	else
	go to Step 3.
	end if

Step 3	If $Urg(i)$==$Urg(j)$ and $Dur(i) > Dur(j)$

	then swap the surgeries;
	else
	go to Step 1 with $i = i + 1$.
	end if
	Stop when all surgeries i in the list are considered.

Step 4	Assign surgery i to room r. i) Declare all surgery i with System(i)==0. ii) Set $d = 1$. iii) Set $r = 1$. iv) Set $i = 1$. v) For surgery i with System(i)==0, Do Calculate $totalDay(d)Room(r)time$ If $totalDay(d)Room(r)time \leq Dur_{max}$ Assign surgery to $d = 1$, $r = 1$ and System(i)=1. go to step (iv) with $i = i + 1$ if i >N $_{Surgeries}$ go to Step (iii) with $r = r + 1$. if r >N $_{Room}$, go to step (ii) with $d = d + 1$. Stop when d $>$ N $_{Day}$

LOCAL SEARCH

After executing the SH, an initial schedule is obtained. We shall improve the schedule in term of the effeciency (computational time and solution's quality) through LS. This heuristic is developed to assist the hospital by giving priority to the urgent patient based on their condition. LS consider the searching of solutions in the neighborhood and it will move from one neigborhood to another to find a better solution. The iteration will continue until no better solution can be found. If the best solution is found, it will replaced the current solution with the best solution.

In the Algorithm 2, we begin by adding the surgery from unscheduled list if it satisfy the condition (Step 1). In Step 2 and Step 3, we will consider the schedule for room, r on day, d. In Step 4, we do not add surgery if it exceeds the time constraint. We will check the sum of OR time before we add the surgery into the schedule . After that we will calculated the new sum of OR time if we added the surgery. The new sum of OR time must be less than or equal to Dur_{max}. If this condition is satisfied, the surgery will be added. After the surgery is added, we proceed to Step 5 where we check the total urgency value and compare it with the current urgency value. If the new urgency value is greater than current urgency value, we will accept the surgery into the schedule. Then we will consider another surgery until all surgeries in the unscheduled list have been considered (Step 6). Finally we will come out with an improve schedule.

Notations used in the Local Search Algorithm	
row	Index for unscheduled surgery
d	Index for days
r	Index for operating rooms
N $_{Day}$	Index for total days
N $_{Room}$	Total number of operating rooms

Dur_{max}	Maximum OR time for each room r in day d
$totalDay(d)Room(r)time$	Sum of duration time

	Algorithm 2: Local Search
Step 1	Set $row = 1$ Addsurgery = Unscheduled((row);:)
Step 2	Set $d = 1$
Step 3	Set $r = 1$
Step 4	if $totalDay(d)Room(r)time < Dur_{max}$ Calculate $newtotalDay(d)Room(r)time$ if $newtotalDay(d)Room(r)time \leq Dur_{max}$ go to Step 5
Step 5	Calculate $TotalNewUrgencyValue$ if $TotalNewUrgencyValue > TotalCurrentUrgencyValue$ Addsurgery to the table else go to Step 6

Step 6	Set $r = r + 1$,
	if r >N $_{Room}$, then
	Set $d = d + 1$
	if d >N $_{Day}$, then
	go to Step 1, $row = row + 1$

SURGEON ASSIGNMENT PROBLEM

In this section, we will discuss the surgeon assignment problem after the surgeries are scheduled. Based on the information from Pradenas et al. (2012), the surgeon will be assigned based on their type and speciality. There are few conditions we need to follow, first, for each surgery scheduled there need to be at least two surgeons in the operation room which is the lead surgeon and assistant surgeon. Lead surgeon is responsible for any decision making during the operation while assistant surgeon will assist lead surgeon during the operation. Second is the surgeon rotation which is the lead surgeon will be free from participating in the next surgery while assistant surgeon will be the lead surgeon on next surgery if any.

A backtracking heuristic (BH) is used to determine if there is another surgery of the same type has been selected in another room simultaneously. This is to prevent from the surgeon being assigned to different room at the same time. We will used this heuristic to solve the surgeon assignment problem.

RESULT & DISCUSSIONS

GENERATED DATA

To test the efficiency of the proposed method, we generate different size of data based on the information from Pradenas et al. (2012). We have to create the data since they only give the general information for the surgery data. Furthermore it is hard to obtain a real hospital data due to a lot of process and it is confidential. We generate three sets of data by using a Uniform Distribution with the notation $U \sim [a,b]$ where the data is generated randomly from integer value of uniform distribution defined on interval $[a,b]$.

We varies the number of surgeries, i that need to be scheduled for each set, N_i for 30,50 and 100 surgeries. The number of ORs available for scheduling, N_{Room} and number of days to schedule the surgeries, N_{Day} are random to test our proposed method in limited amount of capacity. Urgency value for each surgery i, Urg_i is randomly generated from a uniform distribution of [1,4] as an integer value where 4 indicates a greater urgency. The surgery type, $Type_i$ is randomly generated from uniform distribution of [1,2] as an integer value. We only consider two types of surgery since it is hard to obtain information on all types of surgeries from the hospital. **Table 1** show the details of surgeon lists for three set of generated data. For each sets of data we varies the total number of surgeons for both surgery type, $N_{Surgeons}$ for 20,20 and 30 surgeons with each surgery type having the same number of surgeons ($N_{SurgeonsType1}$ and $N_{SurgeonsType2}$).

The surgery duration is assumed between 30 to 120 minutes based on the information that we received from the hospital. The assumption of the surgery duration is obtained from Angiology and Vascular Department, Hospital Tengku Ampuan Afzan (HTAA) located in the Kuantan city. Thus, we randomly generated the surgery duration, Dur_i from uniform distribution of [30,120]. **Table 2** present the details of the surgery lists for three sets of generated data.

Table 1: Details of the surgeon lists for three sets of generated data.

Set	$N_{Surgeons}$	$N_{SurgeonsType1}$	$N_{SurgeonsType2}$
1	20	10	10
2	20	10	10
3	30	15	15

Table 2: Details of the surgery lists for three sets of generated data.

Set	N_i	N_{Room}	N_{Day}	Urg_i	$Type_i$	Dur_i(min)
1	30	2	2			
2	50	2	3	[1,4]	[1,2]	[30,120]
3	100	3	5			

COMPUTATIONAL RESULTS

To analyze the quality of the solutions and computational time for our proposed heuristics (SH_{BH} and LS_{BH}), we need to obtain the optimum values for each size of the problem. Thus, the ILP model are solved using Excel Solver. The proposed heuristics are implemented in software MATLAB R2016a and BH are coded in C language programming software. Tests were performed on a computer with an intel Core i5-2410M processor of 3.3 GHz and 4 GB of RAM. The performance of the results obtained are measured by the percentage of urgency value for scheduled and unscheduled surgeries and also the computational time.

We run the heuristics for ten times to get the average of computational time for each set of data. The average computational time for the surgery scheduling ($Time_{Sc}$ in seconds) and surgeon assignment ($Time_{A}$ in seconds) problem is presented in **Table 3(a)** and **Table 3(b)**. By referring to **Table 3(a)** and **Table 3(b)**, the total computational time ($Time_{Tot}$ in seconds) to obtain the final solution using ILP_{BH} is higher compared to SH_{BH} and LS_{BH}. In **Table 3(a)**, we cannot obtain the computational time for Set 3 since Solver (ILP) cannot solve the model.

We present the results for total urgency value for scheduled (Urg_{S}) and unscheduled (Urg_{US}) surgeries obtained by the model and proposed heuristics in **Table 4**. To further analyze the quality of the solutions by the proposed heuristics, we compare the solutions for heuristics with the solutions of the ILP_{BH}. The solutions by the heuristics for the three sets of data are reasonably good since the gap of scheduled surgeries, Gap_{S}[1] for both heuristics is less than 6%. The Gap_{S} obtained by SH_{BH} is reduced when we used LS_{BH} to improve the schedule. Even though Solver (ILP) cannot find the optimal solution for Set 3, the proposed heuristics are practically as good as the ILP since the Urg_{S} is higher than 90%.

Figure 1(a) and **Figure 1(b)** present the percentages of urgency value for scheduled (Urg_{S}) and unscheduled (Urg_{US}) surgeries for ILP_{BH} and the heuristics (SH_{BH} and LS_{BH}). If we observe **Figure 1(a)**, there is not much difference between the solutions for ILP_{BH} and the proposed heuristics. For Set 1 and Set 2, the Urg_{S} difference between ILP_{BH} and SH_{BH} is less than 5% while the Urg_{S} difference between ILP_{BH} and LS_{BH} is only less than 2%. In **Figure 1(b)**, the Urg_{US} for SH_{BH} is slightly higher than LS_{BH}. Therefore, by referring to the **Figure 1(a)** and **Figure 1(b)**, we shall implement the LS_{BH} to solve the surgery scheduling problem as we get a near to optimal solutions in a reasonable time (less than a minute).

Table 3(a): Average computational time for surgery scheduling and surgeon assignment problem using ILP_{BH}.

Set	ILP_{BH}		
	$Time_{Sc}$(s)	$Time_{A}$(s)	$Time_{Tot}$(s)
1	1800	0.012	1800.012
2	2400	0.011	2400.011
3	-	-	-

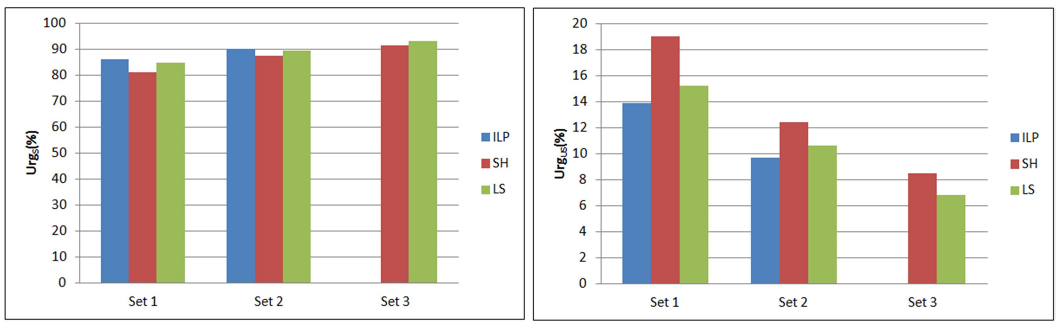
Table 3(b): Average computational time for surgery scheduling and surgeon assignment problem using SH_{BH} and LS_{BH}.

Set	SH_{BH}			LS_{BH}		
	$Time_{Sc}$(s)	$Time_{A}$(s)	$Time_{Tot}$(s)	$Time_{Sc}$(s)	$Time_{A}$(s)	$Time_{Tot}$(s)
1	0.106	0.014	0.120	0.215	0.011	0.226
2	0.120	0.021	0.141	0.248	0.021	0.269
3	0.165	0.135	0.300	0.300	0.135	0.610

[1] Gap_{S} (%) = ((|value of solution-optimal value|) / optimal value) x 100

Table 4: Total urgency value for scheduled and unscheduled surgeries obtained by ILP_{BH} and proposed heuristics.

Set	ILP_{BH}		SH_{BH}			LS_{BH}		
	Urg_S(%)	Urg_{US}(%)	Urg_S(%)	Urg_{US}(%)	Gap_S(%)	Urg_S(%)	Urg_{US}(%)	Gap_S(%)
1	86.1	13.9	81.0	19.0	5.9	84.8	15.2	1.5
2	90.3	9.7	87.6	12.4	3.0	89.4	10.6	1.0
3	-	-	91.5	8.5	-	93.2	6.8	-



(a) **(b)**

Figure 1: Percentages of Urgency value, *Urg*
(a) Scheduled surgeries, Urg_S(%), **(b) Unscheduled** surgeries, Urg_{US}(%).

CONCLUSION

In this paper, we developed two heuristics to solve the elective surgery scheduling problem considering patients urgency. As the number of surgeries increase in the waiting list, ILP model failed to solve the problem and/or consume high computational time. Therefore, we used a Simple Heuristic to obtain the feasible solutions. We improved the solutions by developing a Local Search algorithm based on the Simple Heuristic. To prove the efficiency of our proposed heuristics, we compared the solutions from the mathematical model with the solutions obtained from the heuristics.

Based on the results, both heuristics are very good in reducing the large running time of the ILP model. The Local Search is better since it significantly reduce the computational time while giving a good solution which is as close as the optimal solution from the model. Our main objective in maximizing the summation of the urgency value is also achieved. Since the proposed heuristic is efficient enough, this is a very good procedure to be implemented in solving the operating room scheduling problem considering the patients urgency.

In this work, we estimated the duration of the surgery based on the information from the hospital. It is important that the prediction of the surgery duration used for the heuristics should be as accurate as possible. Since the duration of the surgery is stochastic, it is difficult to obtain a good estimation. The failure to do so might affect the scheduling process in real life situation. Therefore, a detailed study for the surgery durations should be make in the future work. Another future work that can be done is improving the model (Pradenas et al., 2009) into a multi-objectives problem and use metaheuristic for the surgery scheduling and different heuristic procedure for surgeon assignment problem.

ACKNOWLEDGEMENT

The authors would like to thank Hospital Tengku Ampuan Afzan (HTAA), especially the doctor for giving time in providing the relevant information and data. This research is supported by Putra Grant (VOT: 9567600).

REFERENCES

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

Daiki, M. and Yuehwern, Y. (2010), An elective surgery scheduling problem considering patient priority, *Computers & Operations Research*, **37**: 1091 – 1099.

Ergin, E. (2013), *An optimization model for scheduling and rescheduling elective surgery patients under the constraint of downstream units*. Ph.D. thesis, North Dakota State University of Agriculture and Applied Science.

Guillermo, D., Pablo, A. R. and Patricio, W. (2017), Solving the operating room scheduling problem with prioritized lists of patients, *Ann Oper Res*, **258**: 395 – 414.

Joonyup, E., Sang-phil, K., Yuehwern, Y. and Vikram, T. (2018), Scheduling elective surgery patients considering time-dependant health urgency: Modelling and solution approaches, *Omega*, 1 – 17.

Marques, I. and Captivo, M. E. (2017), Different stakeholder's perspectives for a surgical case assignment problem: Deterministic and robust approaches, *European J. Oper. Res.*, **261(1)**: 260 – 278.

Mateus, C., Marques, I. and Captivo, M. E. (2018), Local search heuristics for a surgical case assignment problem, *Operations Research for Health Care*, **17**: 71 – 81.

Moosavi, A. and Ebrahimnejad, S. (2018), Scheduling of elective patients considering upstream and downstream units and emergency demand using robust optimization, *Computers and Indutrial Engineering*, **120**: 216 – 233.

Pradenas, L., Vidal, F. and Melgarejo, E. (2012), An algorithm to surgery scheduling and surgeons assignment in a public hospital, *Revista del Instituto Chileno de Investigacion Operativa*, **2**: 20 – 29.

Rezaeiahari, M. and Khasawneh, M. T. (2017), An optimization model for scheduling patients in destination medical centers, *Operations Research for Health Care*, **15**: 68 – 81.

Tanfani, E. and Testi, A. (2010), An algorithm to surgery scheduling and surgeons assignment in a public hospital, *Ann Oper Res*, **178**: 105 – 119.

Testi, A., Tanfani, E., Valente, R., Ansaldo, L. and Torre, C. (2006), Prioritizing surgical waiting lists, *Journal of Evaluation in Clinical Practice*, **14**: 59 – 64.

Vancroonenburg, W., Smet, P. and Berghe, G. V. (2015), A two-phase heuristic approach to multi-day surgical caes scheduling considering generalized resource constraints, *Operations Research for Health Care*, **7**: 27 – 39.

Yu, W., Jiafu, T. and Gang, Q. (2010), A genetic algorithm for solving patient-priority-based elective surgery scheduling problem, *LSMS 2010,ICSEE 2010*, **6329(2)**: 297 – 304.