Int. J. Res. Ind. Eng. Vol. 10, No. 1 (2021) 46–55
DOI: 10.22105/riej.2021.260145.1158

International Journal of Research in Industrial Engineering

www.riejournal.com

Research Paper

Applying Flexible Job Shop Scheduling in Patients Management to Optimize Processing Time in HospitalsInitial

N. MD. Sarfaraj*, Md. L. R. Lingkon, N. Zahan

Department of Industrial and Production Engineering, Mechanical Faculty, Rajshahi University of Engineering and Technology, Rajshahi, Bangladesh.
*sarfaraj.nagib@gmail.com

A B S T R A C T

The continuous growth of the population causes an increased demand for our healthcare services. Insufficient hospitals face challenges to serve the patient within a preferable duration. Long lines in front of counters increase the processing time of a patient. From the entry to the completion, plenty of time waste just for unscheduled hospital management system. Job shop scheduling is an optimization process in which jobs are assigned with maintain a particular sequence. In this paper, we proposed flexible job shop scheduling to solve this type of problem by considering patients as job and test counter as machine for the optimization of the processing time and increase the efficiency of a hospital or a clinic. Genetic Algorithm was used to analyze the processing time for multiple counter of a hospital for a stable and effective scheduling. The results showed that an optimized makespan was generated and patients could fulfill their needs much quickly after applying flexible job shop scheduling.

Keywords: Scheduling, Makespan, Generic algorithm.

Article history: *Received: 02 December 2020 Reviewed: 19 January 2021 Revised: 19 February 2021 Accepted: 10 March 2021*

Sarfaraj, N, MD., Lingkon, Md. L. R., & Zahan, N. (2021). Applying flexible job shop scheduling in patients management to optimize processing time in hospitals. *International journal of research in industrial engineering*, *10*(1), 46-55.

1. Introduction

For the decision making process in different service programs, scheduling is highly used on regular basis. This process connects the optimization and feasibility both of certain problems occurred for unscheduled management in daily life. Scheduling problems is one of the complex problems to deal with for optimizing a task. Job Shop Scheduling (JSP) is an NPhard problem [1] which deals with a set of machines that are assigned to do a set of tasks by maintaining a sequence. A Flexible Job Shop Scheduling Problem (FJSP) is an extension of classical job shop scheduling. The main difference between JSP and FJSP is that in FJSP each job can be processed

with any machine from a set of a candidate which is called free machine operation relationship by Wang et al. [2]. It increases the flexibility of the operation hence called flexible job shop scheduling. Generally, people all want to fulfill their certain task or need in a minimum duration period. Since the growth rate of population is increasing continuously, without maintaining the proper scheduling it will be difficult for us to fulfill this objective. Hospital is an important field where people have to face the rabble for their required services. Here the patients always want to take their care with minimum time and leave the hospital. The increasing population has a great impact on the health care system which has to deal with heavy patient flow. Hospitals provide different types of health care services, one of them is outpatient services which provide various diagnostics and examination of a patient [3]. Due to the high rate of patients overall processing time is increasing for a patient in hospitals. Then most of the people with minor cases visit the hospital on holidays. In this way, the main impact goes on the total processing time and ultimately the makespan will be increased. All the patients cannot overcome these types of barriers especially at the time of emergency cases. For the aged patients, it is more difficult to fulfill their requirements. A normal patient wastes many times by standing in the line. For completing the full service, a lot of time will be spent. But the 2 productive time spent by that patient is not huge in front of the hospital. As an over-populated country, Bangladesh is the worst sufferer of this problem. A Patient-friendly hospital management system is rare in our country. The maximum of the patient requires different tests on different counters. So, the patients have to move from one counter to another one for completing all of the required tests. Considering the transportation time, the total processing time is higher than expected. For these reasons sometimes the critical patients go back due to the long line. So, it will be very helpful for the patients if the total processing time of each patient can be minimized with the incensement of responsiveness. Implementation of flexible job shop scheduling in the hospital management system can solve all the difficulties faced by the patients. Job shop has various workstations used in performing different types of jobs. Here each machine will perform each operation within a given time. A flexible job shop will offer the flexibility to operate without maintaining that sequence. Teekeng and Thammano developed a method where they combined frog leaping and fuzzy logic [4]. Gao et al. [5] treated a flexible job shop scheduling problem with hybrid genetic and variable neighborhood descent algorithm. Li et al. [6] developed a Pareto-based discrete artificial bee colony algorithm for multi-objective flexible job-shop scheduling problems. Sajadi et al. [7] used a multi-objective genetic algorithm to optimize processing time for a stable FJSP with random machine breakdowns. Kumar and Bisoniya [8] studied the job shop scheduling using a genetic algorithm which is aimed at creating a mathematical model with the help of MATLAB. Xie et al. [9] reviewed many FJSP and studied the use of genetic algorithms and ant colony optimization for the shortest processing time. Pezzella et al. [10] proposed an algorithm based on a basic Genetic Algorithm (GA) to solve the Flexible Job-Shop Scheduling Problem and tested it on instances of 7 jobs and 7 machines. Pinedo [11] in his book developed a genetic algorithm equation for an optimal solution. However, for our case interaction between the patients and the counters with proper scheduled management of the hospital is the main concern. Marynissen and

Demeulemeester [12] studied job shop and flow shop to minimize the processing time of a hospital. Mahanta et al. [13] said patient waiting time decreases as the service rate increases. So the responsiveness of a hospital also depends on its processing time of a patient. Bekal et al. [14] developed a parallel patient treatment algorithm to predict treatment time and reduce processing time. Hartmann et al. [15] discussed a constraint satisfaction model and penalty minimization model for optimizing doctors scheduling for health care centers. From the works of previous researches, we can find that scheduling problem is a concerning subject for industry and beyond. Besides, unscheduled management hampers the services of patients at hospitals. Many researchers used various methods and developed different types of hybrid algorithms to solve FJSP.

However, none of the previous researches had used FJSP to reduce the processing time of a patient in a hospital. In this study, our objectives were minimizing the total processing time of a patient in the hospital and increasing the responsiveness of management by connecting the scheduling problem with flexible job shop scheduling and solve it with the help of a genetic algorithm using MATLAB.

2. Methodology

In this study, a schedule was proposed for a hospital management system to find the optimum processing time for each patient. Flexible job shop scheduling is a scheduling technique which is used to set up a system where different numbers of job can be processed in different numbers of machines at the same time. Unorganized hospital counters are connected with this type of schedule. As mentioned before, various researchers tried to solve different scheduling problems by using FJSP but hospital management systems were not mentioned. By using this technique, we could perform different operations in different counters for different patients at the same time. For conducting a solution, we used the application of a genetic algorithm which is a proven method for solving FJSP. A survey was conducted for finding the time required for the patients to complete their operations in different counters of a hospital. MATLAB (version: R2018a) was used to generate an optimized result based on the data we had collected from the survey. We preferred MATLAB over other software because of its precision in generating accurate value while working with a genetic algorithm as well as wide acceptance in both academic and professional sectors.

2.1. Data Collection

Collecting data and analyzing them gave us an insight into the practical situation of different hospitals in terms of long processing time. We collected our information from three private hospitals in Rajshahi district which is part of Rajshahi divisions, one of the eight divisions of Bangladesh. For precise information gathering, we made questionnaires and talked with many patients, and doctors of those hospitals. We discussed the issue with hospital management

including the officer in charge. We also had conversations with doctors outside of those hospitals. After analyzing those data, we selected five counters which had the most traffic for our research. Based on the information, we collected estimated processing time for each counter. Pie charts showing the processing time of five counter according to the patients are given in ***Figures 1-5***.

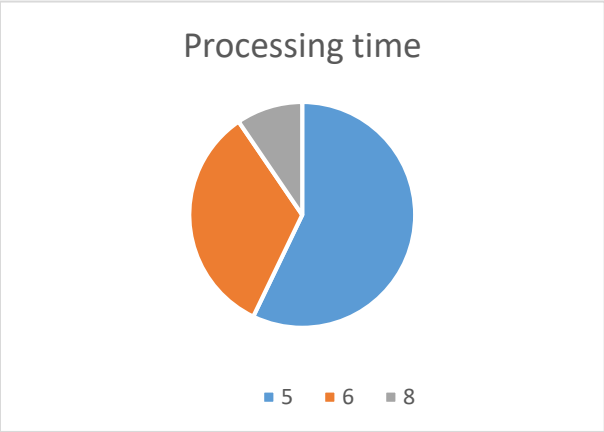

Figure 1. Estimated processing time in min for RBC.

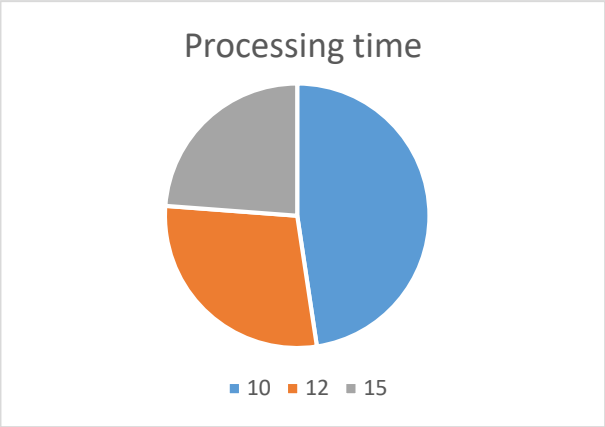

Figure 2. Estimated processing time in min for X-ray.

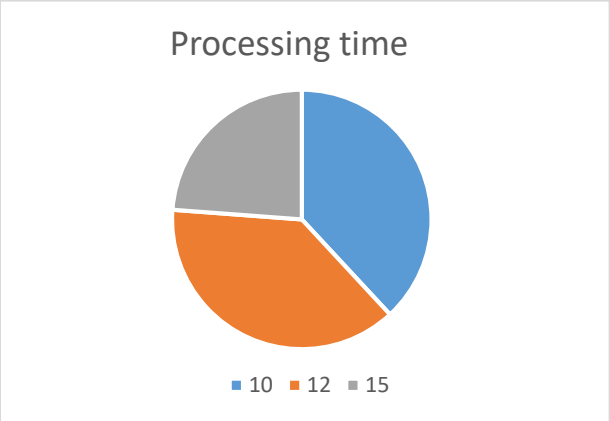

Figure 3. Estimated processing time in min for Pathology.

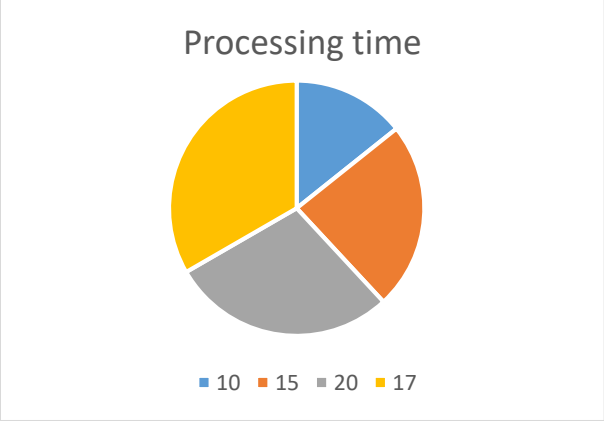

Figure 4. Estimated processing time in min for Ultrasound.

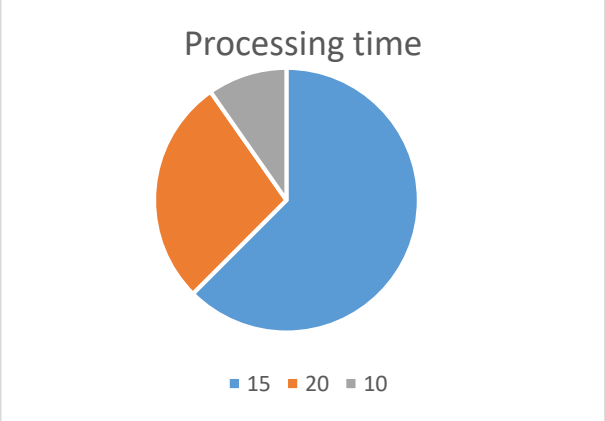

Figure 5. Estimated processing time in min for ECG.

In this paper, machine represented counters which were five in total and we set there processing time 5 min, 10 min, 11 min, 18 min and 15 min, respectively.

2.2. Mathematical Formulation

FJSP deals with M machines and N jobs. Furthermore, specific characteristics of mathematics of this problem of this area of mathematical field are interested for the researchers. Simple form of this problem is classical type that schedules n job from A1, A2…An in the set of B machines of B1, B2, . . . , Bm. Each patient has Cj operation that should be implemented sequentially. Here, j indicates Patients, subscript r indicates operation, and the value of i presents Counter. The Purpose of solving this scheduling problem is determining the sequence of the operations required for each patient in each counter, such that the total processing time for each patient can be minimized. Step 1: Each patient has single sequence having Pj,r operations; r _ 1, . . . , rj , where Pj,r represents hth operation for jth patient, and rj represents number of operations required for j th job. The set of counters is represented by B _ {B1, B2,…, Bm} where i represents counters and j represents patient and r is applied for operations [16]. Step 2: For completing each r operation on A counter represented as Pj,r_, a set of patients will be assigned, which have the ability for completing that operation.

Step 3: The set of patients is represented as Bj,r ⊂ B. Each counter would have individual processing time for finishing the operation. This individual processing time for finishing single operation is represented with Ri,j,r. Step 4: Bj,r will define as a set with variable ai,j,r with the numerical. If variable ai,j,h has value 1, it means that counter j is free for completing an operation. Here, we use different variable yi,j,r with numerical value. This variable is determined from the solution of the model. Flexible Job-shop scheduling can also be reviewed as expressing the ordering between all the operations that should be processed in the same counter of the hospital to fix precedence among all these operations in the counter of the hospital. Unorganized hospital counters are connected with this type of schedule. By using this technique, we can perform different operations in different counters for different patients at the same time. This can easily reduce the overall makespan for each patient. For this process, some assumptions should be made. Such as:

- All counters are independent.
- Each operation can be performed in one counter at a time.
- At time zero, all counters are available and all patients can be operated.
- Interruption will not occur at the time of operating the patients in a counter.
- Among the counters the transportation time will be considered.
- For a certain number of patients, reduction of make span will be calculated.

For our calculations, the processing time needed for each patient is required. From the survey, we collect data (processing time) for patient in different hospitals. Formula: Let, Pij = Processing

time for j in i machine. Where, j represents Patients i represents Counter And, Transportation time is included in processing time.

Here, transportation time is included. So, no transportation time is in the calculation. One counter can be required for multiple patients. That's why sometimes a patient has to wait till the end of service of the previous patient entered into that counter. Waiting time is also considered. Our collected data can be represented as such in ***Table 1***.

Table 1. Schedule information.

Patient	Counter	Processing time (min)
1	1,2	P_{11}=11, P_{21}=5
2	2,3,4	P_{22}=5, P_{32}=10, P_{42}=18
3	2,3	P_{23}=5, P_{33}=10
4	2,3,4	P_{24}=5, P_{34}=10, P_{44}=18
5	1,5	P_{15}=11, P_{55}=15

Our main objective is to minimize the make span for an individual patient. So, from here we can easily find our objective function.

Objective function:

$$\text{Minimize } C_{max} \text{ (Makespan)}$$

Based on some equations from the references we can use them for our collected data. Here we will use 4 basic equations such as:

$$Y_{kj} - Y_{ij} \geq P_{ij},$$

$$Y_{ij} - Y_{ij} \geq P_{ij},$$

$$C_{max} - Y_{ij} \geq P_{ij},$$

$$\text{All variable } \geq 0.$$

For the first equation, difference between starting time of processing for one job in one counter and starting time of processing for previous job in that counter will be greater than or equal to the processing time of the previous job in that counter. For example, difference between starting time of processing for job 2 in counter 2 and starting time of processing for job 1 in counter 2 will be greater than or equal to the processing time of job 1 in counter 2. Similarly, for the next

equation difference between starting time of processing for one job in one counter and starting time of processing for same job in another counter will be greater than or equal to the processing time for same job in same counter. For example, difference between starting time of processing for job 2 in counter 1 and starting time of processing for job 1 in counter 1 will be greater than or equal to the processing time for job 1 in counter 2. Then in the third equation total make span will replace the first term of that of the second equation. Then in the last equation all the variables will be greater than or equal zero.

3. Result

By using MATLAB we found out the result which was the minimized make span.

Table 2. Experimental result.

Patients	Operations	Counter 1 (min)	Counter 2 (min)	Counter 3 (min)	Counter 4 (min)	Counter 5 (min)	Distribution of time (min)	Total time (min)
1st	1	11 (1^{st})	-	-	-	-	11+5(no waiting time)	16
	2	-	5(2^{nd})	-	-	-		
2nd	1	-	5(3^{rd})	-	-	-	10+18+5(no waiting time)	33
	2	-	-	10(1^{s})	-	-		
	3	-	-	-	18 (2^{nd})	-		
3rd	1	-	5 (1^{st})	-	-	-	5+5(waiting)+10	20
	2	-	-	10 (2^{nd})	-	-		
4th	1	-	5 (3^{rd})	-	-	-	18+2(waiting)+10+3(waiting)+5	38
	2	-	-	10 (2^{nd})	-	-		
	3	-	-	-	18 (1^{st})	-		
5th	1	11 (2^{nd})	-	-	-	-	15+11 (no waiting)	26
	2	-	-	-	-	15 (1^{st})		

In MATLAB, all the equations were categorized as genetic algorithm. All the equations were linear. So, by using the function of linear programming in this software we found the value of reduced make span. From the first equation using the data we found seven consecutive equation. Another seven was found for the second one. Similarly, eleven equations were found for the third basic equation. Value of all the 13 variables were greater than or equal zero. So, another 13 equations were found for the last basic equation. Overall, 43 constraints were found using these basic equations with our data. Here, in the calculation firstly all the values of data in the right side of the constraints were categorized as ‘b’. Then, the values of coefficient of the constraints were expressed by ‘A’. As our objectives function was connected only with one variable $Cmax$,

so for the main function the values of all variable except $Cmax$ would be zero. So, the result found for minimized make span was led by the constraints with the function of MATLAB.

Table 3. Comparison between survey and experimental result.

Patient No.	Total processing time (Unscheduled and waiting time included) (min)	Total processing time (Scheduled and no waiting time) (min)
1	20	16
2	40	33
3	23	20
4	40	38
5	30	26

From our objectives, we know that the total processing time should be decreased using our method. Then our result showed the total processing time for each patient was decreased compared to the normal process in daily life. From the table 2, we could see the comparison of the normal process and our scheduled process. For the first patient, the total processing time found from the survey was 20 minutes and 16 minutes from our process. Similarly for the second patient, 40 minutes from the survey and 33 minutes from our process. For the third patient, 23 minutes from the survey and 20 minutes from our process. For the fourth patient, 40 minutes from the survey and 38 minutes from our process. For the fifth patient, 30 minutes from the survey and 26 minutes from our process. From the survey, all the patient had at least 1 minute of waiting time. But from our scheduled process there was no waiting time for the first, second, and fifth patients.

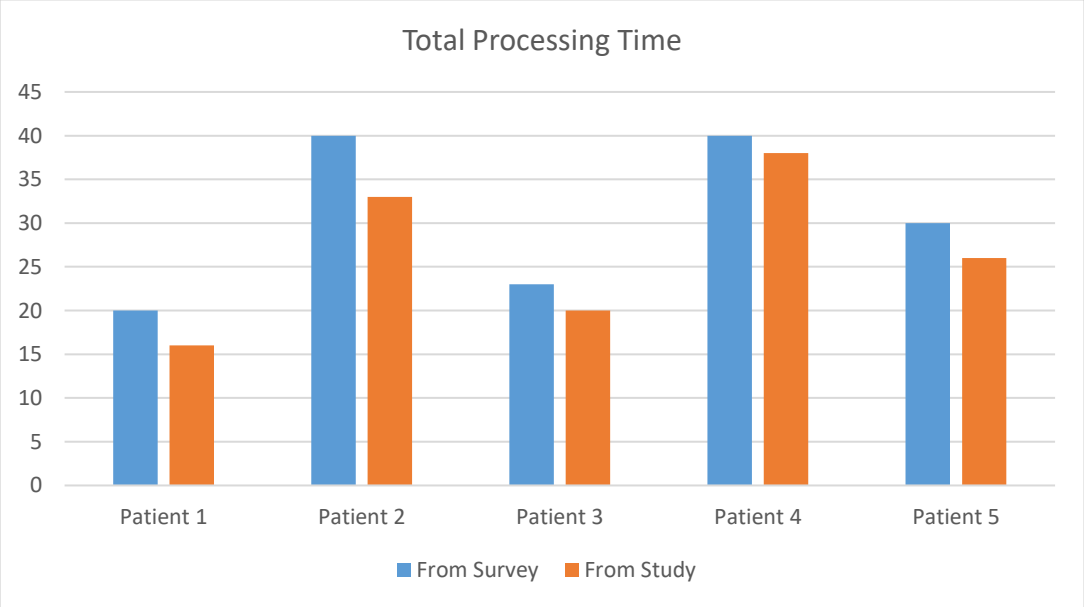

Figure 6. Graphical representation of the reduction of total processing time.

4. Conclusion

The obtained result showed that the processing time of the patients was reduced by applying flexible job-shop scheduling with the help of a genetic algorithm. As we know, the total processing time in a hospital for a patient has a connection with the waiting time and transportation time. Here the total processing time meant the total time required for completing all of the operations. In a general hospital management system, the scheduling system is not designed properly and take more transportation time because people won't even know where they should go after completing one test. When they use the proper scheduling system, they will be informed about their destination after completing one test. For the known destination, the transportation time required will be less compared to an unknown one. So, the transportation time will be decreased and the waiting time required will be less than 0 5 10 15 20 25 30 35 40 45 Patient 1 Patient 2 Patient 3 Patient 4 Patient 5 Total Processing Time From Survey From Study 9 before. Reduced processing time makes patients happy. Hence, hospital management can deal with their patients more effectively which increases their responsiveness. Our study can be implemented in hospitals, clinics, and diagnostic centers. In our process, we try to decrease the total servicing time with minimum transportation and waiting time. But we could not eliminate the waiting time and transportation time completely by using this process. We considered five counters in our paper, which can be considered as the limitation of the study. So, other researchers can consider more counters. Other scheduling methods can also be applied to hospital patient management problems. Further study can be conducted by applying flexible job-shop on hospital bed management or operation theater scheduling problems.

References

[1] Garey, M. R., Johnson, D. S., & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of operations research*, *1*(2), 117-129. https://doi.org/10.1287/moor.1.2.117

[2] Wang, L., Zhou, G., Xu, Y., & Liu, M. (2012). An enhanced Pareto-based artificial bee colony algorithm for the multi-objective flexible job-shop scheduling. *The international journal of advanced manufacturing technology*, *60*(9), 1111-1123. https://doi.org/10.1007/s00170-011-3665-z

[3] Saraswati, T. G., & Saputri, M. E. (2019, May). Queuing management and evaluation of standard operating procedures for hospital mental health polyclinics. *1st international conference on economics, business, entrepreneurship, and finance (ICEBEF 2018)* (pp. 779-782). Atlantis Press. https://doi.org/10.2991/icebef-18.2019.163

[4] Teekeng, W., & Thammano, A. (2011). A combination of shuffled frog leaping and fuzzy logic for flexible job-shop scheduling problems. *Procedia computer science*, *6*, 69-75. https://doi.org/10.1016/j.procs.2011.08.015

[5] Gao, J., Sun, L., & Gen, M. (2008). A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. *Computers and operations research*, *35*(9), 2892-2907. https://doi.org/10.1016/j.cor.2007.01.001

[6] Li, J. Q., Pan, Q. K., & Gao, K. Z. (2011). Pareto-based discrete artificial bee colony algorithm for multi-objective flexible job shop scheduling problems. *The international journal of advanced manufacturing technology*, *55*(9), 1159-1169. https://doi.org/10.1007/s00170-010-3140-2

[7] Sajadi, S. M., Alizadeh, A., Zandieh, M., & Tavan, F. (2019). Robust and stable flexible job shop scheduling with random machine breakdowns: multi-objectives genetic algorithm

approach. *International journal of mathematics in operational research*, *14*(2), 268-289. https://doi.org/10.1504/IJMOR.2019.097759

[8] Kumar, G., & Bisoniya, T. S. (2005). Flexible job shop scheduling operation using genetic algorithm. *Journal of innovations in engineering and technology*, *5*, 1-5.

[9] Xie, J., Gao, L., Peng, K., Li, X., & Li, H. (2019). Review on flexible job shop scheduling. *IET collaborative intelligent manufacturing*, *1*(3), 67-77. DOI: 10.1049/iet-cim.2018.0009

[10] Pezzella, F., Morganti, G., & Ciaschetti, G. (2008). A genetic algorithm for the flexible job-shop scheduling problem. *Computers & operations research*, *35*(10), 3202-3212.

[11] Pinedo, M. (2012). Scheduling (Vol. 29). New York: Springer.

[12] Marynissen, J., & Demeulemeester, E. (2016). Literature review on integrated hospital scheduling problems. *KU Leuven, faculty of economics and business, KBI_1627*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2873413

[13] Mahanta, L. B. (2019). Estimation of the waiting time of patients in a hospital with simple Markovian model using order statistics. *Hacettepe journal of mathematics and statistics*, *48*(1), 274-289.

[14] Bekal, M., Manikandan, A., Tewari, A., & GB, P. (2019). Parallel patient treatment algorithm and it's application in hospital queuing recommendation. *International journal of advance research, ideas and innovations in technology*, *5*(3), 578-582.

[15] Chawasemerwa, T., Taifa, I. W., & Hartmann, D. (2018). Development of a doctor scheduling system: a constraint satisfaction and penalty minimisation scheduling model. *International journal of research in industrial engineering*, *7*(4), 396-422.

[16] Barzegar, B., Motameni, H., & Bozorgi, H. (2012). Solving flexible job-shop scheduling problem using gravitational search algorithm and colored Petri net. *Journal of applied mathematics*, *2012*. https://doi.org/10.1155/2012/651310

©2021 by the authors. Licensee ***International Journal of Research in Industrial Engineering****. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).*