Journal of Mechanical, Civil and Industrial Engineering
www. al-kindipublisher.com/index.php/jmcie
Al-Kindi Center for Research and Development

JMCIE

Optimization of Job Shop Scheduling Problem

Habtamu Tesfaye
School of Mechanical and Industrial Engineering, Dire Dawa Institute of Technology, Dire Dawa University, Dire Dawa, Ethiopia
Corresponding Author: Habtamu Tesfaye, E-mail: habte623@gmail.com

ARTICLE INFORMATION

Received: September 02, 2020
Accepted: November 20, 2020
Volume: 1
Issue: 1

KEYWORDS

Job shop scheduling problem, Lekin, Makespan, Shifting bottleneck heuristics

ABSTRACT

Timely and reliable delivery is becoming a more important factor for manufacturing companies since customer satisfaction becomes the paradigm of world class manufacturers. In some industries, keeping due dates is the bottom line for survival. But due to the NP-hard combinatory characters of the job shop scheduling problems, Hibret Manufacturing and Machine Building Industry (HMMBI) has faced the problem to determine the optimal schedules that minimizes the job completion time and thus failed to meet the promised delivery due date. Therefore, determining the optimal schedule that minimizes the makespan of the job using a shifting bottleneck heuristic was the aim of this study. Secondary data was collected from the production route sheets and Lekin® scheduling software is employed for comparing solution of different heuristics. The findings of the study showed that the shifting bottleneck (DASH) results in a total makespan of 23 hours with 6 hours (20.69%) improvement as compared to the current scheduling system that the HMMB uses

1. Introduction

In today's global market competition, one of the competitive dimensions is delivery lead time. Customers are looking for the producers who can meet their delivery promise without delay. Manufacturers, therefore, have to respond quickly to the orders and meet promised delivery due dates to retain its customer and stay competitive through a satisfied customer. To do so, the company requires an optimum scheduling system that minimizes the makespan and lateness in job shop. Scheduling is the problem of allocating limited shared resources to competing activities over time such that certain objectives are optimized. These objectives can be minimization of makespan time, minimization of maximum lateness (tardiness) time, or minimization of the number of tardy jobs, etc. It has been the subject of a significant amount of literature in the operational research field where the emphasis has been on investigating machine schedule problems in which jobs represent activities and machines represent resources.

The job shop scheduling problem (JSSP) is NP-hard, that have well-earned reputation of being one of the most computationally intractable combinatorial problems considered to date (Kassu & Eshetie, 2015; Gamal & SHassanein, 2013; Raman & Brian F., 1993; Seyda & Gamze, 2009; Balas & Vazacopoulos, 1998). Which means that the number of possible schedules grows exponentially with the number of orders. This is dramatically illustrated in (Gamal & SHassanein, 2013; Balas & Vazacopoulos, 1998) considering the fact that an instance involving 10 jobs and 10 machines, proposed by Muth and Thompson (Gerald Luther, 1963), which have remained unsolved for more than 20 years, until it is solved by Carlier and Pinson (Carlier & Pinson, 1989).

For the solution of JSSPs, exact methods such as integer programming formulation (Carlier & Pinson, 1989), dynamic programming (da Silva, Costa, da Silva, & Pereira, 2012) and branch & bound algorithms (Adams, Balas, & Zawack, 1988; OVACIK & UZSOY, 1992) have been developed to produce optimal solutions. However, the size of the problems and their complexity render exact solutions impractical for job shop scheduling problems. Therefore, there are a plenty heuristic procedures and rules used by researchers to assist in this endeavor. The shifting bottleneck heuristic (SBH) which decomposes the problem into a number of single-machine sub-problems is one of the most successful heuristic approaches for JSSP to minimize the job completion time.

Even though Hibret Manufacturing and Machine Building Industry (HMMBI) have implemented flexible manufacturing system to achieve the flexibilities in part variety, volume, machine flexibility, delivery flexibility, product flexibility, etc., they face the problem of meeting the delivery due date promised to customers. This is due to the lack of the optimal scheduling system. Thus, this study was intended to alleviate the job shop scheduling problems of HMMBI through application of heuristic shifting bottleneck algorithm that minimizes the maximum makespan time (C_{max}) and maximum job lateness time (L_{max}). Lekin® scheduling software is used to compare the results of different heuristics method inbuilt with the software.

The remaining parts of this article is organized in the following manner; section 2 covers the literature review; in section 3 research methodology is presented; section 4 presents the job shop scheduling problem formulation, in section 5 result and discussions are presented and finally in section 6 gives the conclusion of the study.

2. Literature Review

Job shop scheduling problem is one of the most popular scheduling problems and has attracted many researchers due to its both practical importance and complexity. Katuru Phani *et al.* (Raja Kuma, Kumar, & Kamala, 2013) have defined scheduling as the process of assigning the resources over time for the competing activities. In manufacturing, activities correspond to parts or jobs that need to be processed on a set of machines. Thus, scheduling can be considered as optimization problem, with the goal of finding the best schedule. In jobs shops scheduling problem, the purpose is to find a schedule that is an allocation of the operations to time intervals on the machines, which has a minimum duration required to complete all jobs (Carlier & Pinson, 1989).

The job shop scheduling problem is described by *J.* Adams *et al* (Adams, Balas, & Zawack, 1988); jobs are to be processed on machines with the objectives of minimizing some function of the completion times of the jobs, subject to the constraints that the sequence of machines for each job is prescribed and each machine can process only one job at a time.

In the job shop scheduling problem, there are a set 'J' of n-jobs J_1, J_2, J_3...J_n that must be processed on a set 'M' of m-different machines M_1, M_2 M_3,...M_m. Each job J_j consists of a sequence of mj operations (O_{j1}, O_{j2}, O_{j3},... O_{jm}), where O_{jk} (the k^{th} operation of job j) must be processed without interruption on a predefined machine (m_{jk}) during p_{jk} time units. The operations O_{j1}, O_{j2}, O_{j3},..., O_{jm} must be processed one after another in the given order and each machine can process at most one operation at a time (Kassu & Eshetie, 2015; Arisha, Young, & ELBaradie, 2001; Manie & Manier, 2011). The objective is to find an operating sequence for each machine such as to minimize a particular function of job completion times, and in such a way that two operations are never processed on the same machine at any time instance.

To solve the job shop scheduling problem, a large number of exact and heuristics solution methods have already been developed in the past four decades (S. & S, 1999). The branch and bound method described by M. Singer & M. Pinedo (M & M, 1998) which aimed to solve 10-jobs and 10-machine job shop scheduling problem was considered as an exact method, until better solution is found by Mati Y. *et al* (Y, S, & C, 2011).

The works presented in (Balas & Vazacopoulos, 1998; Adams, Balas, & Zawack, 1988; D & Cook, 1991) all have used an exact algorithm to solve a sub problem within a local search heuristic for the job shop scheduling problem. Because of the number of possible sequences growth exponentially as the problem size, the exact algorithms become very computationally intensive for even small-sized job-shop and it does not guarantee optimal solution. As noted in (Seyda & Gamze, 2009), for industrial problems, the computational time of any algorithm must be short enough that the resulting schedule can be used. Hence, a variety of approximation or heuristics procedures have been proposed for finding "good" rather than optimal solution in a reasonably short time. Much of research has involved around dispatching rules (Shutler, 2003; Uzorh & Innocent, 2014), decomposition methods (Raman & Brian F., 1993), meta-heuristic search techniques (Seyda & Gamze, 2009), and Arena simulation (Gamal & SHassanein, 2013)

Most practical job shop scheduling problems have been addressed using dispatching rules which use local information to select a new job to process each time a machine become free. While these rules are computationally efficient and easy to use, they are generally myopic in both space and time and may result in poor long-term performance.

Decomposition methods aim at developing solutions to complex problems by decomposing them into a number of small sub-problems which are more tractable and easier to understand. The most successful of these is the shifting bottleneck heuristic (SBH) proposed originally by Joseph Adams *et al* (Adams, Balas, & Zawack, 1988) for minimizing the makespan.

Katuru Phani *et al*. (Raja Kuma, Kumar, & Kamala, 2013) proposed optimization of job shop scheduling using shifting bottleneck technique to reduce total flow time of job shop scheduling problem and arrive at the optimal solution. Kai-Pei Chen (Chen, 2007) proposed an assembly job shop scheduling problem with component availability constraints, a modified disjunctive graph formulation. Also, he has developed a mixed integer programming model with objective of minimizing the total weight tardiness. WQ. Huang and Z. Huang (WQ. & Z, 2004) propose a modified shifting bottleneck procedure for job shop scheduling with objective to minimize the makespan time. From the literatures reviewed, shifting bottleneck heuristics is efficient method to solve the job shop scheduling problem, when the objectives are to minimize the maximum makespan (C_{max}) and maximum lateness (L_{max}) or tardiness (T_{max}). Thus, shifting bottleneck heuristics selected for this study so that the optimal job sequence can be set for the job shop scheduling problem.

3. Methods and Materials

The study was conducted by considering different materials and methods to achieve the goal of this study. The literatures published on job shop scheduling field are reviewed and shifting bottleneck heuristics method is used determine optimal job sequence. The secondary data regarding jobs under consideration such processing sequence or precedence constraint, processing time, machines on which they are to be processed, was collected from the HMMBI's production route sheet for individual part. The Lekin® scheduling software is then used to generate an alternative schedule using the different heuristics algorithms such as shifting bottleneck heuristics, local search heuristics, hybrid methods and priority dispatching rules built-in the program.

3.1 Shifting Bottleneck Heuristics Algorithm

The shifting bottleneck (SB) was developed by J. Adams *et al* (Adams, Balas, & Zawack, 1988) to solve the general sequencing problem, where the makespan time (C_{max}) and lateness time (L_{max}) was to be minimized. Thus, the SB heuristic is an efficient method to find optimal C_{max} and L_{max} s for job shop scheduling problem (Balas & Vazacopoulos, 1998; Adams, Balas, & Zawack, 1988; Raja Kuma, Kumar, & Kamala, 2013). The SB algorithm sequences machines sequentially one at a time and it is an iterative method. The machines that have not been sequenced are ignored and machines that have been sequenced have their sequences held fixed. At each iteration, the bottleneck machine is identified using $1|r_j|\ L_{max}$ approach.

The SB Algorithm representation: Let be consider the set of machine groups M = {1, 2,...,m}. Each machine group consists of parallel and disjoint machines. The set of machine groups which are already scheduled is designated by M_0. Then the algorithm was summarized by Topaloglu S. and Gamze K. (Seyda & Gamze, 2009) as bellows:

Step 0: Represent the problem as a disjunctive graph G
Step 1: Denote the set of machines by M. Let M_0 be the set of machines already sequenced. Initially, set $M_0 := \emptyset$.
Step 2: Identify and solve the sub-problems for each machine $i \in M-M_0$.
Step 3: Identify a critical machine $k \in M-M_0$.
Step 4: Sequence the critical machine using the sub-problem solution obtained from step 2 by fixing the disjunctive arcs associated with the critical machine in the appropriate direction. Set $M_0 := M_0 \cup \{k\}$.
Step 5: (Optionally) re-optimize the sequence for each machine $m \in M_0-k$ by exploiting the information provided by the newly added disjunctive arcs for machine k.
Step 6: If $M_0 = M$, stop. Otherwise, go to step 2.

4. Job Shop Problem and its formulation

Hibret Manufacturing and Machine Building Industry (HMMBI) is one of metal factory among 17 industries ruled under the Metals and Engineering Corporation of Ethiopia. HMMBI has five factories under it, machine building factory, precision machinery factory, materials treatment, conventional manufacturing factory, and Engineering factory. Machine building factory was selected as specific focus areas of the study. In this section variety of machine components & spare parts are produced as per the customers' specification and thus it is a job shop production system.

For this study five parts (jobs) frequently arriving at the company are considered namely; pulley motor (J_1), Roller scraper shaft (J_2), Can carrier shaft (J_3), Pulley (J_4), and sprocket gear (J_5). These jobs are to be processed on four machines namely power hack saw (M_1), CNC lathe machine (M_2), CNC milling machine (M_3), and grinding machine (M_4). The processing sequence or precedence constraint, processing time, machines on which each job to be processed was collected from the company's production route sheet for individual part. The job shop scheduling problem of HMMBI is formulated as shown in table 1 and it was solved under the following assumptions:

i. All machines are available during scheduling period
ii. Jobs are available at a time $t_0 = 0$ & $r_j = 0$
v. Each machine can process only one job at a time
vi. Each job visits each machine only once
vii. Pre-emption is not allowed
viii. No job splitting
ix. Setup times is assumed to be small & included in the processing time
x. Processing time is deterministic & constant.

Table 1: Routing matrix for five jobs and four machines

Job	Operations precedence constraints			
J_1	$M_1(2)$	$M_2(3)$	$M_3(2)$	$M_4(2)$
J_2	$M_1(2)$	$M_3(4)$	$M_2(2)$	$M_4(1)$
J_3	$M_1(3)$	$M_2(5)$	$M_3(2)$	$M_4(3)$
J_4	$M_1(1)$	$M_2(3)$	$M_3(1)$	$M_4(1)$
J_5	$M_1(3)$	$M_2(6)$	$M_3(3)$	$M_4(3)$

5. Result and Discussion

The job shop scheduling problem formulated for HMMBI under section four is solved using Lekin flexible job shop scheduling software that has four heuristics method inbuilt with it under the job shop scheduling workspace layout. Using these heuristic methods, different performance measures such as;

i. Makespan (C_{max}),
ii. Tardiness (T_{max}),
iii. Total number of tardy jobs ($\sum U_j$),
iv. Total flow time ($\sum C_j$),
v. Total tardiness ($\sum T_j$),
vi. Total weighted flow time ($\sum w_j C_j$), and
vii. Total weighted tardiness ($\sum w_j T_j$) are evaluated and compared.

The finding of the study has revealed that the shifting bottleneck (DASH) with the objective to minimize maximum tardiness (T_{max}) yields optimal schedule. The shifting bottleneck (DASH) has resulted in equal minimum makespan and tardiness of 23 hours which is determined by the critical job sequence J_4- J_5- J_3- J_2- J_1. The general SB routing and local search heuristics with makespan minimization objective has resulted similar minimum makespan and tardiness to the DASH heuristics algorithm. But the general SB routing and local search heuristics are not best algorithm if the objective is to minimize the total flow time ($\sum C_j$) and total tardiness of the jobs ($\sum T_j$). The computational results of the simulation run using Lekin scheduling software and the job sequence on each machine (machine loading) is presented in table 2 and table 3 respectively.

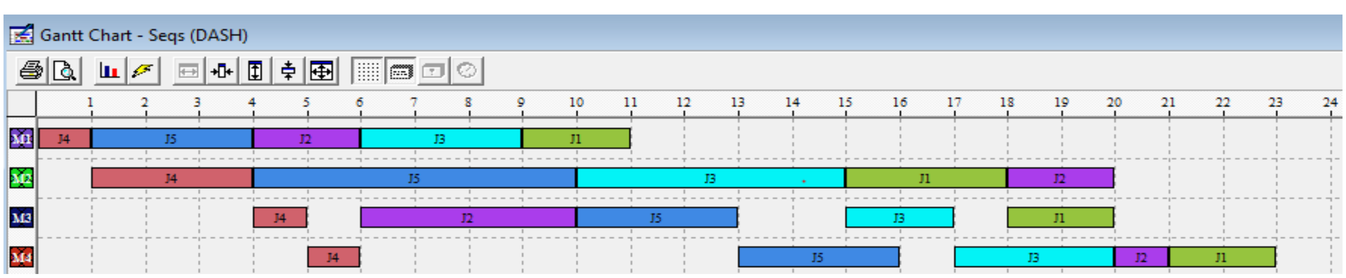
Table 2: Results obtained from different simulation run using heuristic methods built in Lekin® software

No.	**Heuristic method built in Lekin**	**Run Time**	C_{max}	T_{max}	$\sum U_j$	$\sum C_j$	$\sum T_j$	$\sum w_jC_j$	$\sum w_jT_j$
1	General SB Routing/C_{max}	1	23	23	5	92	92	92	92
2	Shifting Bottleneck/Sum(wT)	1	27	27	5	80	80	80	80
3	Shifting Bottleneck/T_{max} (DASH)	2	23	23	5	86	86	86	86
4	Local Search/C_{max}	61	23	23	5	92	92	92	92
5	First come first served priority rule	1	29	29	5	93	93	93	93

Table 3: Optimum job sequence on each machine (Machine Loading)

S/N	**Heuristic Method**	**Machine**	**Job Sequence**
1	General SB Routing with objectives of minimizing makespan time (C_{max})	M1	$J_1, - J_5 - J_2 - J_3 - J_4$
		M2	$J_1 - J_5 - J_3 - J_4 - J_2$
		M3	$J_1 - J_2 - J_5 - J_3 - J_4$
		M4	$J_1 - J_5 - J_3 - J_2 - J_4$
2	Shifting Bottleneck with objectives of minimizing total weighted tardiness ($\sum w_jT_j$)	M1	$J_4 - J_2 - J_1 - J_3 - J_5$
		M2	$J_4 - J_1 - J_2 - J_3 - J_4$
		M3	$J_2 - J_4 - J_1 - J_3 - J_5$
		M4	$J_4 - J_2 - J_1 - J_3 - J_5$
3	Shifting Bottleneck (DASH) with objectives of minimizing tardiness (T_{max})	M1	$J_4 - J_5 - J_2 - J_3 - J_1$
		M2	$J_4 - J_5 - J_3 - J_1 - J_2$
		M3	$J_4 - J_2 - J_5 - J_3 - J_1$
		M4	**$J_4 - J_5 - J_3 - J_2 - J_1$**
4	Local Search with objectives of minimizing makespan time (C_{max})	M1	$J_1 - J_5 - J_2 - J_3 - J_4$
		M2	$J_1 - J_5 - J_3 - J_2 – J_4$
		M3	$J_1 - J_2 - J_5 - J_3 - J_4$
		M4	$J_1 - J_5 - J_3 - J_2 - J_4$
5	First come first served priority rule	M1	$J_1 - J_2 – J_3 - J_4 - J_5$
		M2	$J_1 - J_3 - J_2 - J_4 – J_5$
		M3	$J_2 - J_1 - J_3 - J_4 - J_5$
		M4	$J_1 - J_2 - J_3 - J_4 - J_5$

The Gantt chart that shows the sequences of jobs or machine loading for the optimal schedule developed by Lekin flexible job shop scheduling system using shifting bottleneck (DASH) algorithm is presented on the Fig. 1.



Whereas the Gantt chart showing job sequence or machine loading based on first come first served priority rule which currently HMMBI uses is shown on Fig.2.

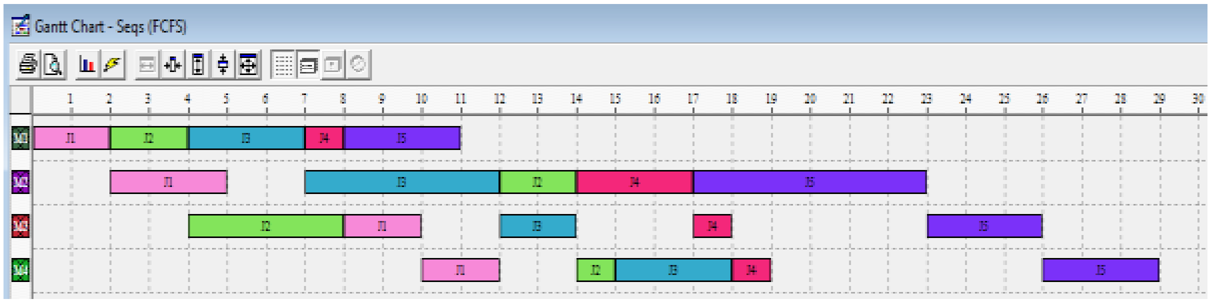


Figure 2: Gantt chart for HMMBI's existing job sequence or machine loading

The findings of the study also showed that the percentage utilization of each machine which was obtained by dividing the total processing time on each machine by the makespan. Accordingly, the percentage utilization of machine one is 47.83%, machine two 91.30%, machine three 43.48%, and machine four 43.48%. Machine two (CNC lathe) is the busiest machine running for 21 hours of its available time during the schedule period (*i.e.* C_{max} = 23 hours) whereas other machines percentage utilization is less than 50%.

6. Conclusion

This case study aims at scheduling of 4-machines and 5-jobs using shifting bottleneck heuristic methods built in Lekin® scheduling software based on secondary data collected from HMMBI production route sheets. The findings of the study showed that the local search and shifting bottleneck (DASH) are resulted in a total makespan of 23 hours with 6 hours (20.69%) improvement as compared to the current scheduling system (*i.e* First Come First Served priority rule) that resulted in makespan time of 29 hours and resulted in better machine hour utilizations rate.

Funding Information

This research received no external funding.

Acknowledgement

The author would like to express sincerely appreciations to all the HMMBI employees who have helped by providing information during interview and for their willingness to provide necessary secondary data.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] Adams, J., Balas, E., & Zawack, D. (1988). The Shifting Bottleneck Procedure for Job Shop Scheduling. *Management Science, 34*(3), 391-401. Retrieved from http://www.jstor.org/stable/2632051

[2] Arisha, A., Young, P., & ELBaradie, M. A. (2001). Job Shop Scheduling Problem: An Overview. *International Conference for Flexible Automation and Intelligent Manufacturing (FAIM 01)*, (pp. 682-693). Dublin, Ireland.

[3] Balas, E., & Vazacopoulos, A. (1998). Guided Local Search with Shifting Bottleneck for Job Shop Scheduling. *Management Science, 44*(2), 262-275.

[4] Carlier, J., & Pinson, E. (1989). An Algorithm for Solving the Job Shop Problem. *Management Science, 13*(2), 164-176. Retrieved from http://www.jstor.org/stable/2631909

[5] Chen, K.-p. (2007). Assembly job shop scheduling problems with component availability constraints. Norman, Ocklahoma.

[6] D, A., & Cook, W. (1991). A Computational Study of the Job-Shop Scheduling Problem. *ORSA Journal on Computing, 3*(2), 149-156.

[7] da Silva, E. B., Costa, M. G., da Silva, M. F., & Pereira, F. H. (2012). Evaluation of Production Sequencing Rules in Job Shop and Flow Shop Environment through Computer Simulation. *nternational Conference on Industrial engineering and Operations Management.* Portugal.

[8] Dan, D., Stoean, R., & Stoean, C. (2007). Genetic Chromo-Dynamics for the Job Shop Scheduling Problem. *Proceedings of International Conference on Knowledge Engineering, Principles and Techniques* (pp. 153-160). Cluj-Napoca (Romania): Research Gate.

[9] Gamal, M., & SHassanein, E. W. (2013, March). Solving the Job-Shop Scheduling Problem by Arena Simulation Software. *International Journal of Engineering Innovation & Research, 2*(2), 161-166.

[10] Gerald Luther, T. (1963). *Industrial Scheduling.* (E. C. N.J, Ed.) Prentice-Hall.

[11] Kassu, J., & Eshetie, B. (2015). Job Shop Scheduling Problem for Machine Shop With Shifting Heuristic Bottleneck. *GLOBAL JOURNAL OF RESEARCHES IN ENGINEERING; J GENERAL ENGINEERING, 15*(1), 21-26.

[12] M, S., & M, P. (1998). A computational study of branch and bound techniques for minimizing the total weighted tardiness injob shops. *IIE Transactions, 28*(1), 193-207.

[13] Manie, Q. Z., & Manier, M.-A. (2011). A disjunctive graph and shifting bottleneck heuristics for hoists scheduling problem. *Preprints of the 18thInternational Federation of Automatic Control (IFAC) World Congress.* Milano, Italy.

[14] OVACIK, I. M., & UZSOY, R. (1992). A Shifting Bottleneck Algorithm for Scheduling Semiconductor Testing Operations. *Journals of Electronics manufacturing, 2*(03), 119-134. doi:https://doi.org/10.1142/S0960313192000157

[15] Raja Kuma, K. P., Kumar, D. A., & Kamala, D. V. (2013). Optimisation Of Job Shop Scheduling Problem For Batch Production. *International Journal of Engineering Research & Technology (IJERT), 2*(7), 1634-1636. Retrieved from www.ijert.org

[16] Raman, N., & Brian F., T. (1993). The Job shop Tardiness Problem: A Decomposition Approach. *European Journal of Operations Research, 69*, 187-199.

[17] S., A., & S, M. (1999). Deterministic Job Shop Scheduling: Past, Present and Future. *European Journal of Operational Research, 113*, 390-434.

[18] Seyda, T., & Gamze, K. (2009). A modified shifting bottleneck heuristics for the reentrant job shop scheduling problems with amkespan minimization. *International Journal of Advanced Manufacturing, 44*, 781-794.

[19] Shutler, P. M. (2003). A Priority List Based Heuristic for the Job Shop Problem. *The Journal of the Operational Research Society, 54*(6), 571-584.

[20] Uzorh, A., & Innocent, N. (2014). Solving Machine Shops Scheduling Problemsusing Priority Sequencing Rules Techniques. *The International Journal Of Engineering And Science (IJES), 3*(6), 15-22.

[21] WQ., H., & Z, H. (2004). Algorithm based on taboo search and shifting bottleneck for job shop scheduling. *J. Comput. Sci. & Technol. , 19*(776). doi:https://doi.org/10.1007/BF02973438

[22] Y, M., S, D.-P., & C, L. (2011). A general approach for optimizing regular criteriain the job-shops cheduling problem. *European Journal of Operational Research, 212*(1), 33-42.