ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: https://doi.org/10.29121/granthaalayah.v7.i1.2019.1036

INTERNATIONAL JOURNAL OF RESEARCH – GRANTHAALAYAH
A knowledge Repository

Science

JOB SHOP SCHEDULING CONSIDERING MAKESPAN, PENALTIES OF MACHINE IDLING, AND JOB OUT OF TIME

Jaber S. Alzahrani [*1]

[*1] Department of Industrial Engineering, Engineering College at Alqunfudah, Umm Al-Qura University, Saudi Arabia

Abstract

In this study, the job shop scheduling problem is analysed with respect to numerous objectives, and a multi-objective model is developed to optimise the makespan, maximum job earliness, maximum job tardiness, maximum machine idle time, total machine idling penalties, total job untimed penalties, and total penalties (multi-objective) using the genetic algorithm.

Keywords: JSS; Genetic Algorithm; Multi-Objective Optimisation.

Cite This Article: Jaber S. Alzahrani. (2019). "JOB SHOP SCHEDULING CONSIDERING MAKESPAN, PENALTIES OF MACHINE IDLING, AND JOB OUT OF TIME." *International Journal of Research - Granthaalayah*, 7(1), 73-82. 10.29121/granthaalayah.v7.i1.2019.1036.

1. Introduction

The adoption of job shop scheduling (JSS) enables the achievement of various benefits, such as cost reduction, time reduction, and customer satisfaction, which are the most desirable objectives in modern industries. Most previous research has been concerned with time reduction objectives, such as minimising the makespan and/or flow time.

Reducing the overall time undoubtedly has an effect on cost reduction, but in certain cases, reducing the holding time of one machine may have a significant effect on other machines. This cost may be defined as the leasing cost if the machine is leased, as is the case in most projects, and may be the nonutilised capacity or depreciation cost in other cases. This holding time may be significantly reduced by decreasing the idle time between starting and finishing the machine use. Moreover, finishing jobs before or after the predefined due date may add certain costs as penalties; thus, reducing the earliness and tardiness of the jobs should affect the total cost, although not all jobs have the same effect or penalty. French [1] and Sridhar and Rajendran [2] stated that the objective of the minimisation of the makespan, flow time, and idle time is the reduction of scheduling costs.

Numerous studies have been conducted in the scheduling field. The majority of these studies have involved solving the flow shop scheduling problem (FSSP) by focusing on minimising the makespan. However, various other important objectives exist in addition to the makespan, such as the total flow time and total machine idle time, which are very important performance measures for minimising the total scheduling cost [3]. The first research concerned with the FSSP was carried out by Johnson (1954) [4].

The FSSP has been the subject of numerous efforts. Rajendran [3] presented one branch-and-bound algorithm and two heuristic algorithms for a two-machine FSSP to optimise the makespan and total flow time. Neppalli, Chen, and Gupta [5] proposed two genetic algorithms for the FSSP. Gupta, Palanimuthy, and Chen [6, 7] developed a heuristic algorithm and a tabu search algorithm for this purpose. Moreover, T'kindt, Gupta, and Billaut [8] presented a branch-and-bound algorithm, mathematical programming formulations, and a heuristic algorithm. Later, Huang and Lim [9] presented a fragmental optimisation heuristic algorithm combining dynamic programming and local search strategies. For two-machine FSSPs, Nagar, Haddock, and Heragu [10] developed a branch-and-bound algorithm to minimise a weighted combination of the total flow time and makespan. Furthermore, Yeh [11] created a memetic algorithm to solve this problem.

Rajendran [12], Framinan, Leisten, and Ruiz-Usano [13], as well as Ravindran, Noorul Haq, Selvakuar, and Sivaraman [14], proposed efficient heuristic algorithms in which weights were assigned to each objective with respect to its importance. Moreover, Daniels and Chambers [15], Ishibuchi and Murata [16], Chang, Hsieh, and Lin [17], Varadharajan and Rajendran [18], and Pasupathy, Rajendran, and Suresh [19] presented multi-objective algorithms in which sets of efficient (or nondominated or Pareto-optimal) solutions were developed.

Sridhar et al. [2] developed a flow shop multi-objective model that minimised the makespan, total flow time, and machine idle time; however, their model aimed to optimise the makespan and flow time in the cell; therefore, we cannot introduce exceptional elements into this approach.

In job shop scheduling problems (JSSPs), minimising the maximum completion time has been the common objective of the majority of the research. The JSSP is an NP-hard problem; therefore, it is difficult to determine an exact solution within a reasonable computation time [20]. A variety of optimisation methods have been developed to solve the JSSP, including mathematical programming [21, 22], the tabu search [23, 24], genetic algorithms [25, 26], simulated annealing [27, 28], particle swarm optimisation [29, 30], goal programming [31], and ant colony optimisation [32, 33].

The majority of studies have tackled scheduling problems as mono-objective problems [22, 25, 26, 34, 35]. The scheduling process requires multi-objective treatment to consider conflicting objectives [36]. Thus, researchers often deal with problems involving multiple, usually conflicting, criteria [37].

The aim of this study is to present a multi-objective JSSP model using genetic algorithms to optimise the makespan, maximum job earliness, maximum job tardiness, maximum machine idle time, total machine idling penalty, total job untimed penalty, and total penalty (multi-objective). The spreadsheet-based genetic algorithm solver "Evolver" [38] was used to optimise this problem.

2. Problem Description and Assumptions

The problem considered in this research is the scheduling of N jobs with predefined processing sequences and due dates on M different machines for predetermined durations, to optimise different objectives such as makespan, maximum job earliness, maximum job tardiness, maximum machine idle time, total machine idling penalties, total job untimed penalties, and total penalties (multi-objective), considering the following assumptions:

- All jobs are ready for processing at time zero.
- All machines are available at the starting time of the first processed job.
- Any machine may be used in any other work directly after finishing the required jobs.
- Each job does not visit the same machine twice.
- Processing times are known and deterministic, and include the setup time.
- The operation cannot be interrupted.
- The idle time is the time between processing different jobs.
- No breakdown of machines is considered.

3. Model Formulation

Parameters:
N: Number of jobs
M: Number of machines
P_{ji}: Processing time for job j on m/c, j = 1, 2, …, N and i = 1, 2, …, M
D_j: Due date of job j, j =1, 2, …, N
EP_j: Job j earliness penalty per unit time
TP_j: Job j tardiness penalty per unit time
MIP_i: Machine i idling penalty per unit time
SEQ: Processing sequence array
NUMT: Number of machines (tasks) for each job
NUMJ: Number of jobs per machine J
DISJ: Disjunction array

Decision Variables:
C_j: Completion time of job j
S_{ji}: Starting time of job j on machine i
F_{ji}: Finishing time of job j on machine i
E_j: Earliness of job j = (D_j - C_j) if $D_j > C_j$, and 0 otherwise
T_j: Tardiness of job j = (C_j - D_j) if $C_j > D_j$, and 0 otherwise
P_{ji}: Processing time for job j on m/c, j = 1, 2, …, N and i = 1, 2, …, M
MITi: Idle time of machine i $= MAX(Fji) - \sum_{j\in N} Pji$
MIT_i: Idle time of machine i = (C_j - D_j) if $C_j > D_j$, and 0 otherwise

3.1. Objective Functions

The following seven objectives were considered in the developed model:

1) minimisation of the makespan;

2) minimisation of the maximum job earliness;
3) minimisation of the maximum job tardiness;
4) minimisation of the maximum machine idle time;
5) minimisation of the total machine idling penalties;
6) minimisation of the total job untimed penalties; and
7) minimisation of the total penalties (multi-objective).

These seven objectives are formulated in Equations (1) to (7), respectively.

$$\text{minimise f } 1 = \text{MAX}\ (\text{C}_j),\ \forall\ \text{j}\ \in \text{N} \tag{1}$$

$$\text{minimise f } 2 = \text{MAX}\ (\text{T}_j),\ \forall\ \text{j}\ \in \text{N} \tag{2}$$

$$\text{minimise f } 3 = \ \text{MAX}\ (\text{E}_j),\ \forall\ \text{j}\ \in \text{N} \tag{3}$$

$$\text{minimise f } 4 = \ \text{MAX}\ (\text{MIT}_i),\ \forall\ \text{i}\ \in \text{M} \tag{4}$$

$$\text{minimise f } 5 = \ \textstyle\sum_{i \in M}(\text{MIT}_i * \text{MIP}_i) \tag{5}$$

$$\text{minimise f } 6 = \ \textstyle\sum_{j \in N}(\text{T}_j * \text{TP}_j + \text{E}_j * \text{EP}_j) \tag{6}$$

$$\text{minimise f } 7 = \ \textstyle\sum_{i \in M}(\text{MIT}_i * \text{MIP}_i) + \ \sum_{j \in N}(\text{T}_j * \text{TP}_j + \text{E}_j * \text{EP}_j) \tag{7}$$

3.2. Constraints

Disjunction constraints (to avoid overlapping between tasks):

$$\left(\text{S}_{\text{ij}} - \text{S}_{\text{ij}}\right) \geq \text{P}_{\text{ij}} - \text{M Y}_{\text{hij}}\ , \forall \text{i}, \text{j}\ \in \text{N}, \forall \text{i} \in \text{M} \tag{8}$$

$$\left(\text{S}_{\text{ij}} - \text{S}_{\text{ij}}\right) \geq \text{P}_{\text{ij}} - \text{M}\ (1 - \text{Y}_{\text{hij}})\ , \forall \text{i}, \text{j}\ \in \text{N}, \forall \text{i} \in \text{M} \tag{9}$$

Conjunction constraints (to satisfy operational precedence):

$$\textstyle\sum_{\text{i} \in \text{M}}\left(\text{S}_{\text{SEQ(j,i),j}} + \text{P}_{\text{SEQ(j,l),j}}\right) \geq \sum_{\text{i} \in \text{M}} \text{S}_{\text{SEQ(j,i+1),j}}\ , \forall\ \text{j}\ \in \text{N}, \forall \text{i} \in \text{M} - 1. \tag{10}$$

4. Computational Results and Analysis

In this section, the results of applying the proposed model are discussed. The model was solved using the Evolver solver, which runs on an Intel® Core™ i3-2310M CPU @2.10 GHz (3 GB RAM). The genetic algorithm parameters include: population size N = 50, number of generations G = 40,000, probability of crossover Pc = 0.5, and probability of mutation Pm = 0.1.
The model accuracy and capability were verified through solving and analysing a 5J*4M problem, which was solved seven times to optimise seven different objectives, as follows:

1) Minimum makespan
2) Minimax job earliness

3) Minimax job tardiness
4) Minimax machine idle time (to increase machine utilisation and reduce leasing costs)
5) Minimum total machine idling penalties (to decrease machine leasing costs)
6) Minimum total job penalties (to reduce customer dissatisfaction)
7) Minimum total penalties (multi-objective).

The model inputs, processing sequences, durations, and due dates were assumed as displayed in Tables 1 and 2, while Table 3 presents a summary of the results of the seven cases.

Table 1: Job processing sequences

Job	Processing sequence			
J1	1	3	4	
J2	1	2	3	4
J3	4	3	2	
J4	2	3	4	1
J5	1	3		

Table 2: Duration matrices of processes and job due dates

M/J	J1	J2	J3	J4	J5
M1	19	10		14	15
M2		30	15	10	
M3	10	18	18	20	16
M4	19	11	31	19	
Due dates	100	115	90	85	31

The earliness and tardiness penalties per unit time may be determined from the customer contracting terms, but in this research, the values were assumed as 100, 10, 10, 10, and 10 for the five jobs for both earliness and tardiness, respectively.

The idling penalty per unit time for each machine may be equal to the leasing cost for leased machines, or to the depreciation rate for owned machines. In this research, these values were assumed as 50, 5, 5, and 5 for the four machines, respectively.

Table 3: Obtained results of seven objectives

Case	1	2	3	4	5	6	7
Makespan	92	220	109	93	93	115	115
Maximum earliness	31	0	6	26	26	0	21
Maximum tardiness	61	189	0	62	62	0	15

ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: 10.5281/zenodo.2550095

Maximum idle time	18	146	31	11	11	35	35
Total m/c idling penalties	475	8,745	1,675	445	445	1,780	700
Total job penalties	4,370	14,660	310	3,550	3,550	0	360
Total penalties	4,845	23,405	1,985	3,995	3,995	1,780	1,060

The first case of minimising the makespan is considered as the master case, in which optimising the makespan is the objective of the majority of research on JSS, and all other case behaviours are compared therewith. The Gantt chart of the master case of 92 optimal makespans is depicted in Figure 1.

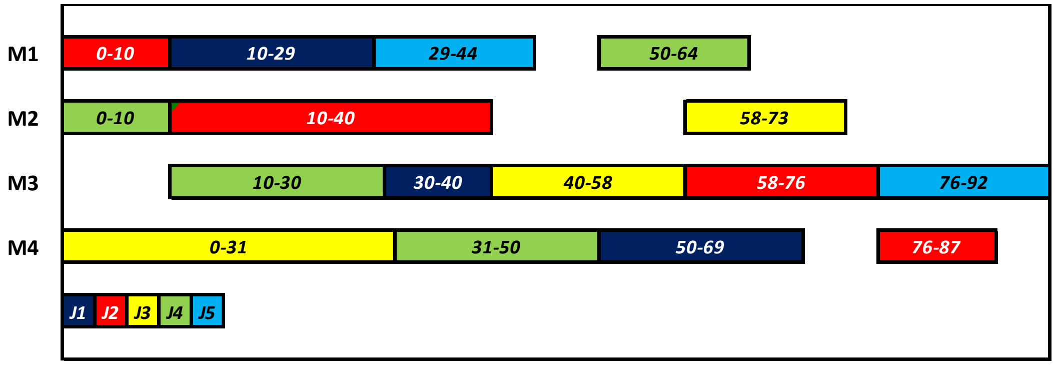

Figure 1: Gantt chart of makespan objective

The detailed schedule of the second case of minimising the maximum earliness is presented in Table 3, from which it can be observed that the maximum earliness is zero, but the makespan and all other objectives obtained the lowest and unacceptable values. Thus, it is not recommended to optimise the maximum earliness. Figure 2 depicts the Gantt chart of the third case of minimising the maximum tardiness, which is more reasonable but does not consider penalty costs.

Table 3: Schedule of second case of optimising maximum earliness

	J1	J2	J3	J4	J5
M1	127 - 146	146 - 156	---	177 - 191	158 - 173
M2	---	156 - 186	186 - 201	128 - 138	---
M3	158 - 168	186 - 204	168 - 186	138 - 158	204 - 220
M4	177 - 196	204 - 215	127 - 158	158 - 177	---

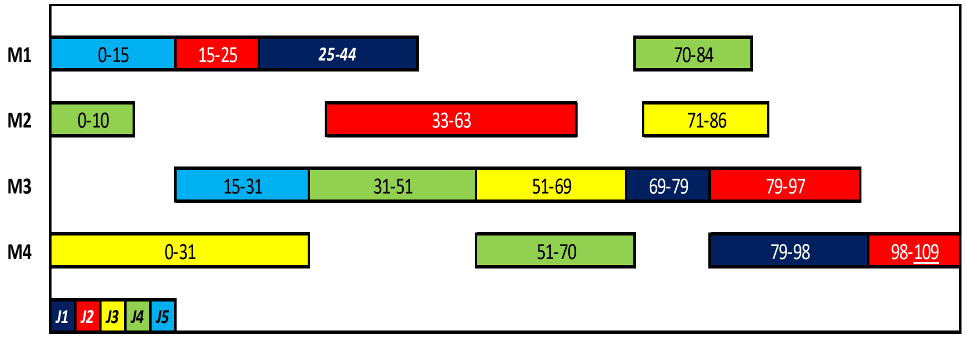

Figure 1: Gantt chart of maximum tardiness objective

ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: 10.5281/zenodo.2550095

The schedules of the fourth and fifth cases of optimising the maximum machine idle time and total idling penalties are presented in Figures 3 and 4, respectively. Both schedules have the same values, with a slight difference in the loading of the third machine, where the two objectives are almost compatible. The shortcoming of these two cases is neglecting the job earliness and tardiness penalties, while considering only the machine idling, in contrast to the sixth case illustrated in Figure 5, which considered only the job earliness and tardiness penalties, while neglecting the machine idling.

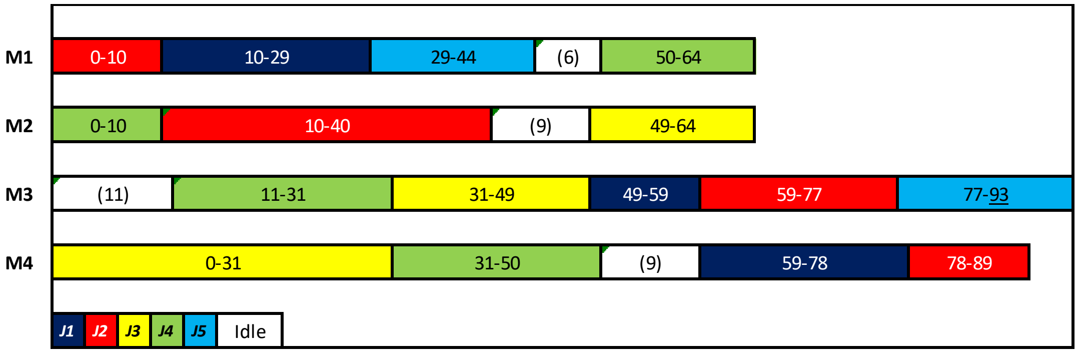

Figure 3: Gantt chart of maximum idle time objective

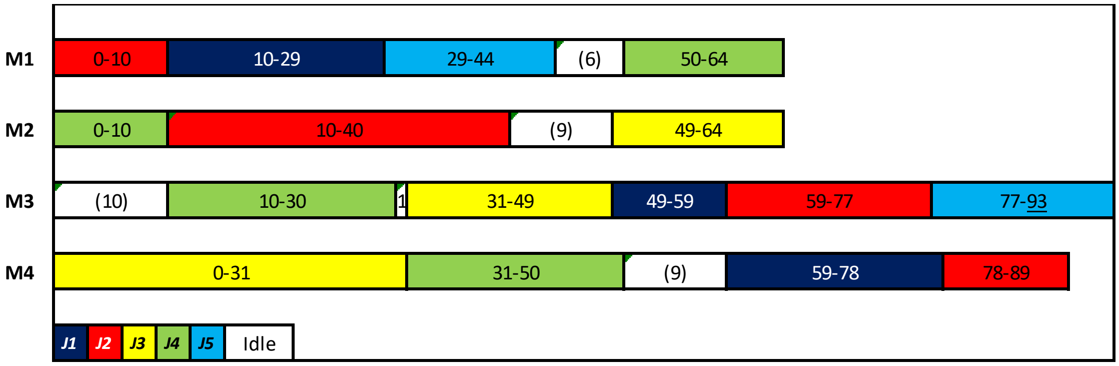

Figure 4: Gantt chart of total idling penalties objective

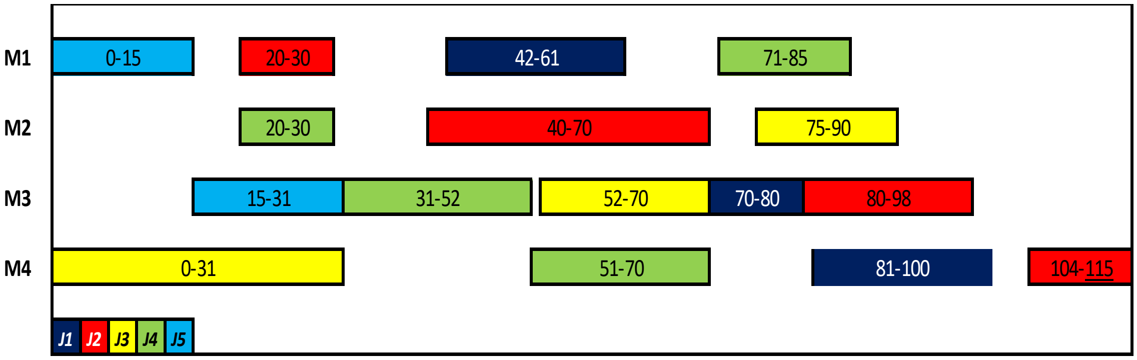

Figure 5: Gantt chart of total job penalties objective

Figure 6 presents the schedule of the seventh case of optimising the total penalties relating to job earliness and tardiness in addition to the machine idling. It can be concluded that the sixth and seventh cases are the most reasonable, as the sixth case is more suitable for ensuring the Just In Time (JIT) strategy, while case seven is more suitable for reducing the total cost.

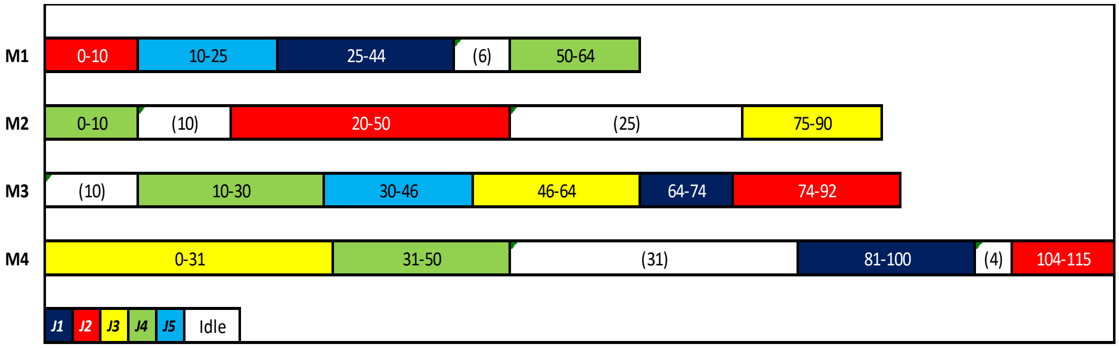

Figure 6: Gantt chart of total penalties objective

5. Conclusion

This research successfully solved the JSSP respecting numerous objectives, such as makespan, maximum job earliness, maximum job tardiness, maximum machine idle time, total machine idling penalties, total job untimed penalties, and total penalties (multi-objective), by using the genetic algorithm.

Reducing the makespan undoubtedly has an effect on decreasing the cost, but in many cases, the holding cost of one machine may expend substantial costs for other machines in the form of the leasing cost or nonutilised capacity or depreciation cost. Moreover, finishing jobs before or after the predefined due date may add certain costs as penalties, so reducing the earliness and tardiness of the jobs also affects the total cost, although not all jobs have the same effect or penalty. Therefore, it is necessary to optimise the total penalty in order to overcome the aforementioned penalties.

It is also concluded that optimising the maximum earliness time is not recommended.

It can be concluded that optimising the total job penalties is more suitable for ensuring the JIT strategy, while optimising the total penalties, including those of jobs and machines, is more suitable for reducing the total cost.

References

[1] S. French, "Sequencing and scheduling," An Introduction to the Mathematics of the Job-Shop, 1982.

[2] J. Sridhar and C. Rajendran, "Scheduling in flowshop and cellular manufacturing systems with multiple objectives—a genetic algorithmic approach," Production Planning & Control, vol. 7, pp. 374-382, 1996.

[3] B. Yagmahan and M. M. Yenisey, "A multi-objective ant colony system algorithm for flow shop scheduling problem," Expert Systems with Applications, vol. 37, pp. 1361-1368, 2010.

[4] S. M. Johnson, "Optimal two-and three-stage production schedules with setup times included," Naval research logistics quarterly, vol. 1, pp. 61-68, 1954.

[5] V. R. Neppalli, C.-L. Chen, and J. N. Gupta, "Genetic algorithms for the two-stage bicriteria flowshop problem," European journal of operational research, vol. 95, pp. 356-373, 1996.

[6] J. N. Gupta, "Designing a tabu search algorithm for the two-stage flow shop problem with secondary criterion," Production Planning & Control, vol. 10, pp. 251-265, 1999.

[7] J. N. Gupta, V. R. Neppalli, and F. Werner, "Minimizing total flow time in a two-machine flowshop problem with minimum makespan," International Journal of Production Economics, vol. 69, pp. 323-338, 2001.

[8] V. T'kindt, J. N. Gupta, and J.-C. Billaut, "Two-machine flowshop scheduling with a secondary criterion," Computers & Operations Research, vol. 30, pp. 505-526, 2003.

[9] G. Huang and A. Lim, "Fragmental optimization on the 2-machine bicriteria flowshop scheduling problem," in Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on, 2003, p. 194.

[10] A. Nagar, S. S. Heragu, and J. Haddock, "A branch-and-bound approach for a two-machine flowshop scheduling problem," Journal of the Operational Research Society, vol. 46, pp. 721-734, 1995.

[11] W.-C. Yeh, "A Memetic Algorithm for the n/2/Flowshop/α F+ β C max Scheduling Problem," The International Journal of Advanced Manufacturing Technology, vol. 20, pp. 464-473, 2002.

[12] C. Rajendran, "Heuristics for scheduling in flowshop with multiple objectives," European journal of operational research, vol. 82, pp. 540-555, 1995.

[13] J. M. Framinan, R. Leisten, and R. Ruiz-Usano, "Efficient heuristics for flowshop sequencing with the objectives of makespan and flowtime minimisation," European Journal of Operational Research, vol. 141, pp. 559-569, 2002.

[14] D. Ravindran, S. Selvakumar, R. Sivaraman, and A. N. Haq, "Flow shop scheduling with multiple objective of minimizing makespan and total flow time," The international journal of advanced manufacturing technology, vol. 25, pp. 1007-1012, 2005.

[15] R. L. Daniels and R. J. Chambers, "Multiobjective flow-shop scheduling," Naval Research Logistics (NRL), vol. 37, pp. 981-995, 1990.

[16] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 28, pp. 392-403, 1998.

[17] P.-C. Chang, J.-C. Hsieh, and S.-G. Lin, "The development of gradual-priority weighting approach for the multi-objective flowshop scheduling problem," International Journal of Production Economics, vol. 79, pp. 171-183, 2002.

[18] T. Varadharajan and C. Rajendran, "A multi-objective simulated-annealing algorithm for scheduling in flowshops to minimize the makespan and total flowtime of jobs," European Journal of Operational Research, vol. 167, pp. 772-795, 2005.

[19] T. Pasupathy, C. Rajendran, and R. Suresh, "A multi-objective genetic algorithm for scheduling in flow shops to minimize the makespan and total flow time of jobs," The International Journal of Advanced Manufacturing Technology, vol. 27, pp. 804-815, 2006.

[20] M. R. Garey, D. S. Johnson, and R. Sethi, "The complexity of flowshop and jobshop scheduling," Mathematics of operations research, vol. 1, pp. 117-129, 1976.

[21] M. S. Al-Ashhab, Munshi, S., Oreijah, M., & Ghulman, H., "Job Shop Scheduling Using Mixed Integer Programming," International Journal Of Modern Engineering Research, vol. 7, p. 7, 2017.

[22] K. R. Baker and B. Keller, "Solving the single-machine sequencing problem using integer programming," Computers & Industrial Engineering, vol. 59, pp. 730-735, 2010.

[23] M. Dell'Amico and M. Trubian, "Applying tabu search to the job-shop scheduling problem," Annals of Operations research, vol. 41, pp. 231-252, 1993.

[24] A. Ponsich and C. A. C. Coello, "A hybrid differential evolution—tabu search algorithm for the solution of job-shop scheduling problems," Applied Soft Computing, vol. 13, pp. 462-474, 2013.

[25] R. Cheng, M. Gen, and Y. Tsujimura, "A tutorial survey of job-shop scheduling problems using genetic algorithms—I. Representation," Computers & industrial engineering, vol. 30, pp. 983-997, 1996.

[26] Y. Wang, "A new hybrid genetic algorithm for job shop scheduling problem," Computers & Operations Research, vol. 39, pp. 2291-2299, 2012.

[27] H. R. Lourenco, "Job-shop scheduling: Computational study of local search and large-step optimization methods," European Journal of Operational Research, vol. 83, pp. 347-364, 1995.

[28] M. Faccio, J. Ries, and N. Saggiorno, "Simulated annealing approach to solve dual resource constrained job shop scheduling problems: layout impact analysis on solution quality," International Journal of Mathematics in Operational Research, vol. 7, pp. 609-629, 2015.

[29] D. Sha and C.-Y. Hsu, "A hybrid particle swarm optimization for job shop scheduling problem," Computers & Industrial Engineering, vol. 51, pp. 791-808, 2006.

[30] T.-L. Lin, S.-J. Horng, T.-W. Kao, Y.-H. Chen, R.-S. Run, R.-J. Chen, et al., "An efficient job-shop scheduling algorithm based on particle swarm optimization," Expert Systems with Applications, vol. 37, pp. 2629-2636, 2010.

[31] M. Al-Ashhab, "Multi-Objective Job Shop Scheduling Using a Lexicographic Procedure."

[32] K.-L. Huang and C.-J. Liao, "Ant colony optimization combined with taboo search for the job shop scheduling problem," Computers & operations research, vol. 35, pp. 1030-1046, 2008.

[33] A. Udomsakdigool and V. Kachitvichyanukul, "Multiple colony ant algorithm for job-shop scheduling problem," International Journal of Production Research, vol. 46, pp. 4155-4175, 2008.

[34] H. M. Wagner, "An integer linear-programming model for machine scheduling," Naval Research Logistics (NRL), vol. 6, pp. 131-140, 1959.

[35] A. S. Manne, "On the job-shop scheduling problem," Operations Research, vol. 8, pp. 219-223, 1960.

[36] A. Scaria, K. George, and J. Sebastian, "An Artificial Bee Colony Approach for Multi-objective Job Shop Scheduling," Procedia Technology, vol. 25, pp. 1030-1037, 2016.

[37] K. P. Yoon and C.-L. Hwang, Multiple attribute decision making: an introduction vol. 104: Sage publications, 1995.

[38] N. Palisade, "Guide to Evolver–The genetic algorithm solver for Microsoft Excel," ed: Newfield, NY: Palisade Corporation, 1998.

*Corresponding author.
E-mail address: jszahrani@ uqu.edu.sa