Robust Multi-Objective Job Shop Scheduling Under Uncertainty

Mohamed S. Al-Ashhab[1,2] **and Jaber S. Alzahrani**[*3]
msashhab@uqu.edu.sa jszahrani@uqu.edu.sa
[1]Department of Mechanical Engineering, College of Engineering and Islamic Architecture, Umm Al-Qura University, Makkah 24211, Saudi Arabia
[2]Design & Production Engineering Department, Faculty of Engineering, Ain-Shams University, Cairo 11511, Egypt
[3]Department of Industrial Engineering, Engineering College at Alqunfudah, Umm Al-Qura University, Saudi Arabia.

Summary

In this study, a multi-objective robust job-shop scheduling (JSS) model was developed. The model considered multi-jobs and multi-machines. The model also considered uncertain processing times for all tasks. Each job was assigned a specific due date and a tardiness penalty to be paid if the job was not delivered on time. If any job was completed early, holding expenses would be assigned.In addition, the model added idling penalties to accommodate the idling of machines while waiting for jobs. The problem assigned was to determine the optimal start times for each task that would minimize the expected penalties. A numerical problem was solved to minimize both the makespan and the total penalties, and a comparison was made between the results. Analysis of the results produced a prescription for optimizing penalties that is important to be accounted for in conjunction with uncertainties in the job-shop scheduling problem (JSSP).

Keywords: *Job-Shop Scheduling; Optimization; Uncertainty; Xpress-IVE; Robust; Penalty.*

1 Introduction

Shop scheduling problems are well-known optimization problems in the field of operations research [1], [2].

1.1 Common Objectives of Optimization

Different objective functions are used in scheduling problems, the makespan measure being the most common and widely implemented measure that is used. The makespan of a system is a measure of the total completion time required for all operations of all jobs in the system. Banu Calis et al. [3] summarized the well-established objective functions of the job-shop scheduling problem (JSSP). Table 1 shows the list of commonly used objectives followed by their definition.

Table 1: Commonly used objectives in Shop Scheduling

Objective	Description
Makespan	Total completion time: the time taken to complete all jobs.
The total workload	The total processing time across all machines.
Max workload	The machine(s) with the largest processing time(s).
Max lateness	The largest difference between the completion time and due date.
Mean flow time	The average amount of time a given job spends on the shop floor.
Tardiness	The difference between a job's completion time and its deadline.
Total tardiness	The sum of all the jobs' tardiness.
Mean tardiness	The average of all the jobs' tardiness.
Weighted tardiness	The tardiness of a given job multiplied by a cost weight.

The majority of studies on job-shop scheduling (JSS) have aimed to minimize the makespan [4]–[10]. Al-Ashhab et al. [11] proposed a mixed integer programming (MIP) model to minimize the makespan, total tardiness, and total earliness separately, whereas Al-Ashhab [12] used the lexicographic procedure to optimize the same three objectives instantly. Singh et al. [13] proposed a hybrid algorithm using cuckoo search optimization (CSO) with an enhancement scheme to solve the problem of minimizing the makespan.

In other approaches, Lai et al. [14] solved the JSSP considering the mean flow time criterion subject to random processing times, whereas Chan et al. [15],[16] considered the minimization of the late cost, inventory cost, penalty cost, and setup cost in addition to the makespan. Huang [17] considered minimizing the costs of the setup time, material processing, and inventory. Alzahrani [18] formulated a multi-objective JSS procedure to optimize the makespan,

Manuscript received August 5, 2022
Manuscript revised August 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.8.6

total earliness, and total tardiness without considering their cost effects, using the pre-emptive constraint procedure.

1.2 Shop Layouts

Clewett [19] introduced a pioneering classification of scheduling problems into single-stage and multi-stage production systems. Single-stage layouts consist of single-machine or parallel-machine systems, whereas multi-stage layouts include flow shop, job shop, and open-shop systems.

Otala et al. [20] proposed a comprehensive classification of system layouts. They incorporated complicated modern layouts such as mixed-shop, cellular, and flexible manufacturing systems into their classification. Job shops (JSs) are well-known manufacturing systems built for a wide range of applications. The JSS problem is a frequent topic of study due to its NP-hardness [21].

1.3 Solving Methods

Lin et al. [22] classified the methods used to solve scheduling problems based on the precision of their solutions. Many efforts have been made to solve the JSSP using different methods that have been developed for this purpose. These include mathematical programming [11], [23], goal programming [12], tabu search [24], ant colony optimization [25], memetic algorithms [26], simulated annealing [27], genetic algorithms [28], particle swarm optimization [29], [30], and differential evolution algorithms [31].

1.4 Uncertainty Optimization

The majority of studies have approached the JSSP by assuming deterministic processing times, whereas others have been based on its uncertainty. Robustness and stability have also been considered [32]. Lu et al. [33] addressed the problem of finding a robust and stable schedule for a single machine.

Al-Ashhab et al. [34] developed a single objective model to solve the JSSP that considered the uncertainties in the processing times. Wang et al. [35] also considered uncertain processing times. Tavakkoli-Moghaddam et al. [36] studied the JSSP with random operations in order to minimize the idling cost of machines.

Varthanan et al. [37] developed an efficient particle swarm algorithm to solve both the deterministic and the stochastic problems and minimize the total cost. Golenko et al. [38] developed a single-objective optimization model to solve the JSSP with random durations and varying costs and expenses.

Most of the aforementioned studies aimed at optimizing time-based objectives rather than cost-based objectives and most of the developed heuristics in the JSSP targeted the problem of time to minimize the makespan. A minority of studies considered how to solve the JSSP by considering the roles of time and cost simultaneously. Moreover, most studies have assumed job processing to be deterministic. A number of studies have considered uncertain processing times.

The present study is an extension of prior work done by Al-Ashhab et al. [34] and Alzahrani [18] to solve the JSSP by processing several jobs with uncertain processing sequences on several machines, optimizing both time and cost objectives instantaneously, in the form of the lexicographic procedure. The model optimizes several objectives, including the makespan, maximum lateness, total lateness, total earliness, and total idling time of machines and other components. In the present study, the makespan, total tardiness, and total penalty costs due to earliness and tardiness of all jobs were optimized, in addition to the idling of all machines. The earliness penalty of a job refers to holding expenses where most products need specific storing conditions in addition to the storing space, a scenario that makes the holding process costly.

2 Model Formulation

The JSSP consists of the processing of multi-jobs on multi-machines and the determination of the starting and finishing times for each job on each machine that optimize the required objectives.

The formulation of the model used in this study was based on the following assumptions:

- The processing times are uncertain;
- Each job is independent of the other jobs;
- Each job has its own due date;
- Each job will visit the same machine no more than once;
- All jobs and machines are ready at time zero;
- Each machine can process only one job at a time;
- Recirculation is not allowed;
- Each job may follow a unique path through the machines to fulfill all operations.

The following sets, parameters, and decision variables were incorporated in the model:

i) Sets:

J: The set of jobs

M: The set of machines

ii) Parameters:

P_{jm}: The processing time for job j on machine m

PM_{jm}: The mean value of the processing time for job j on machine m

PS_{jm}: The standard deviation of the processing time for job j on machine m

D_j: The due date of the job j, with $j \in J$

SEQ: The processing sequence array

SC_j: The storage expenses of job j per unit time

DC_j: The penalty of a delay in job j per unit time

PC_j: The penalty of a delay in job j

IP_m: The idling penalty of machine m per unit time

W_j: The weight of job j

iii) Decision Variables

S_{jm}: The starting time of job j on machine m

$S_{\mathrm{SEQ}(j,m),j}$: The starting time of job j on machine m in its sequence matrix

$S_{\mathrm{SEQ}(j,m+1),j}$: The starting time of job j on the next machine ($m+1$) in its sequence matrix

$P_{\mathrm{SEQ}(j,m),j}$: The processing time for job j on machine m in its sequence matrix

F_{jm}: The finishing time of job j on machine m

C_j: The completion time of job j

E_j: The earliness of job j

T_j: The tardiness of job j

IT_m: The idle time of machine m

SP_j: The single penalty for not performing job j on time (to be paid once)

TP_j: The tardiness penalty for job j

EP_j: The earliness penalty for job j

TJ_j: A binary variable of tardy jobs

M: Big number

$$Y_{mij} = \begin{cases} 1, \text{if job } i \text{ processed on machine } m \text{ after job } j \\ 0, \quad \text{Otherwise} \end{cases}$$

2.1 Objective Functions

The developed model minimizes three objectives instantaneously: the makespan, total tardiness, and total penalty cost.

2.1.1 Total Penalty Cost

The total penalty cost is calculated by summing the costs due to the idling penalty of machines, the tardiness penalty, and the early penalty.

The idling penalty of machines due to waiting for jobs is calculated using Eq. (1). It may be regarded as a nonutilized machine penalty. The idling penalty for each machine is calculated by multiplying its idle time (which is calculated using Eq. (2) by its idling penalty per unit of time.

$$\text{Idling (non} - \text{utilizing) Penalty} = \sum_{m \in M} (IT_m * IP_m) \tag{1}$$

$$IT_m = \max\left(F_{jm}\right) - \sum_{j \in J} P_{jm} \tag{2}$$

The tardiness penalty for each job is calculated by multiplying its tardiness (calculated using Eq. (3)) by the tardiness penalty for the same job. The sum of the tardiness penalties of all jobs produces the total tardiness penalty expressed in Eq. (4).

$$T_j = \begin{cases} C_j - D_j & \text{if } C_j > D_j \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

$$\text{Tardiness Penalty} = \sum_{j \in J} (T_j * TP_j) \tag{4}$$

The earliness penalty of each job is calculated by multiplying its earliness time (calculated using Eq. (5)) by the earliness penalty associated with the storing of the same job. The sum of the earliness penalties of all jobs produces the total earliness penalty expressed in Eq. (6).

$$E_j = \begin{cases} D_j - C_j & \text{if } D_j > C_j \\ 0 & \text{Otherwise} \end{cases} \tag{5}$$

$$\text{Earliness (Storage) Penalty} = \sum_{j\in J}(E_j * W_j * EP_j) \tag{6}$$

The objective of the total penalty is calculated using Eq. (7).

$$\begin{aligned}\text{Total Penalty} = \sum_{m\in M}(IT_m * IP_m) + \sum_{j\in J}(T_j * TP_j) \\ + \sum_{j\in J}(E_j * W_j * EP_j)\end{aligned} \tag{7}$$

2.1.2 Total Tardiness

The total tardiness is the sum of the tardiness values of all jobs and is calculated using Eq. (8).

$$\text{Total Tardiness} = \sum_{j\in J}(T_j) \tag{8}$$

2.1.3 Makespan function

The makespan function represents the maximum completion time as expressed in Eq. (9).

$$\text{Makespan} = \max(C_j) \tag{9}$$

2.2 Constraints

The constraints expressed in Eq. (10) and Eq. (11) are called disjunction constraints. Their purpose is to avoid the overlapping of jobs on any machine. The constraint expressed in Eq. (12) is called the conjunction constraint. Its purpose is to assure processing precedence.

$$S_{im} \geq S_{jm} + P_{jm} - M\,Y_{mij}, \qquad \forall\, i,j \in J, \forall\, m \in \tag{10}$$

$$\begin{aligned}S_{jm} \geq S_{im} + P_{im} - M\,(1 - Y_{mij}), \\ \forall\, i,j \in J, \forall\, m \in M\end{aligned} \tag{11}$$

$$\begin{aligned}S_{\mathrm{SEQ}(j,m),j} + P_{\mathrm{SEQ}(j,m),j} \geq S_{\mathrm{SEQ}(j,m+1),\mathrm{j}}\,, \\ \forall\, j \in J, \forall\, m \in M-1\end{aligned} \tag{12}$$

3 Computational Results and Analysis

The proposed model was solved using FICO XpressIVE optimization software using the Mosel language and ran on an Intel (R) Core (TM) i7-7700 CPU @ 3.60 GHz computer with 8 GB RAM.

3.1 Model Verification

In this sub-section, the results of applying the proposed model are presented and analyzed. The accuracy and capability of the model were verified by solving a numerical problem. In order to simplify the verification discussion in this sub-section, the problem parameters were assumed to be deterministic, and the maximum allowable deviation of the objectives was assumed to be zero.

Three problems with different ordering priorities of the three objectives presented in Table 2 were solved using the lexicographic procedure, assuming deterministic durations to verify the capability of the model. The corresponding processing hours, processing sequences, and due dates are presented in Table 3.

Table 2: The orders of the three objectives for respective problems

	Objective 1	**Objective 2**	**Objective 3**
Prob. 1	Makespan	Total tardiness	Total penalties
Prob. 2	Total tardiness	Makespan	Total penalties
Prob. 3	Total penalties	Makespan	Total tardiness

Table 3: Durations, processing sequences, and due dates of jobs

	Machine	**Job 1**	**Job 2**	**Job 3**
Duration (hours)	**M/C 1**	50	100	50
	M/C 2	0	20	30
	M/C 3	40	50	60
Processing sequence	**M/C 1**	1	2	1
	M/C 2	0	1	3
	M/C 3	2	3	2
Due date		100	200	300
Earliness penalty per unit of time		1	1	1
Tardiness penalty per unit of time		1	1	1

(i) Problem 1

Table 4 and Fig. 1 display the schedule and Gantt chart, respectively, of Problem 1. The optimal makespan of this problem was 240 hours. The resulting values of the completion time, earliness, and tardiness of the three jobs are presented in Table 4.

Table 4: Schedule of Problem 1 (in hours)

	Job 1	**Job 2**	**Job 3**	**Idling**
M/C 1	150 – 200	50 – 150	0 – 50	0
M/C 2		0 – 20	210 – 240	190
M/C 3	200 - 240	150 - 200	50 - 110	90
Finishing Time	240	200	240	
Due date	100	200	300	
Tardiness	140	0	0	
Earliness	0	0	60	

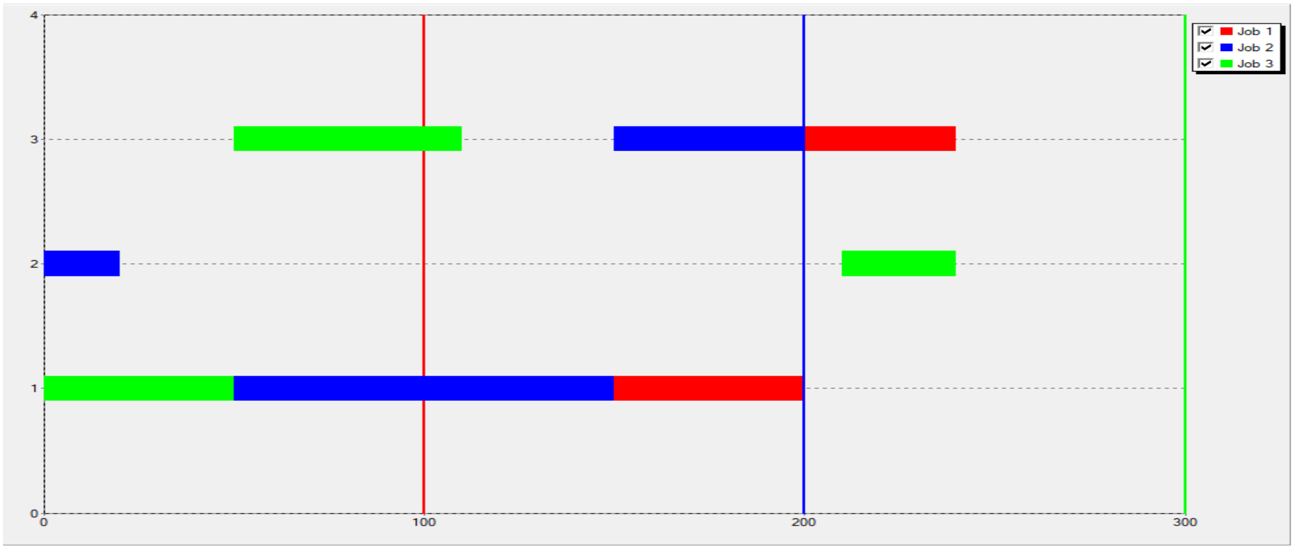


Fig. 1 The Gantt chart of Problem 1

(ii) Problem 2

Table 5 and Fig. 2 display the schedule and Gantt chart, respectively, of Problem 2. The optimal makespan of this problem was 290 hours. The resulting values of the finishing time, earliness, and tardiness of the three jobs are presented in

Table 5.

Table 5: Schedule of Problem 2

	Job 1	**Job 2**	**Job 3**	**Idling**
M/C 1	0 – 50	50 – 150	150 – 200	0
M/C 2		0 – 20	260 – 290	240
M/C 3	60 - 100	150 - 200	200 - 260	110
Finishing Time	100	200	290	
Due date	100	200	300	
Tardiness	0	0	0	
Earliness	0	0	0	

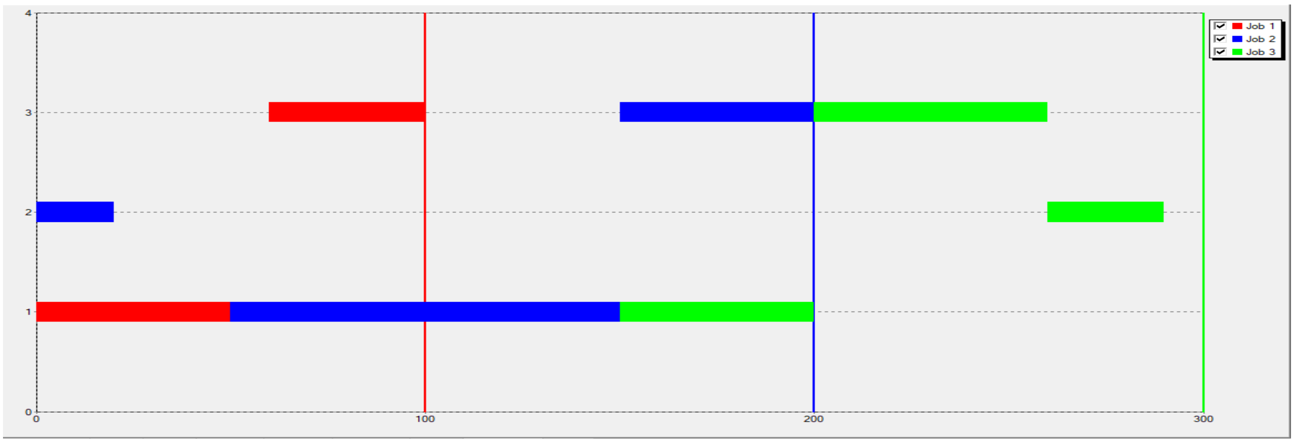


Fig. 2 The Gantt chart of Problem 2

(iii) Problem 3

The resulting schedule of Problem 3 is the same as the resulting schedule of Problem 2 as shown in Table 5, because the first optimized objectives in the two problems coincide in reducing the tardiness of all jobs.

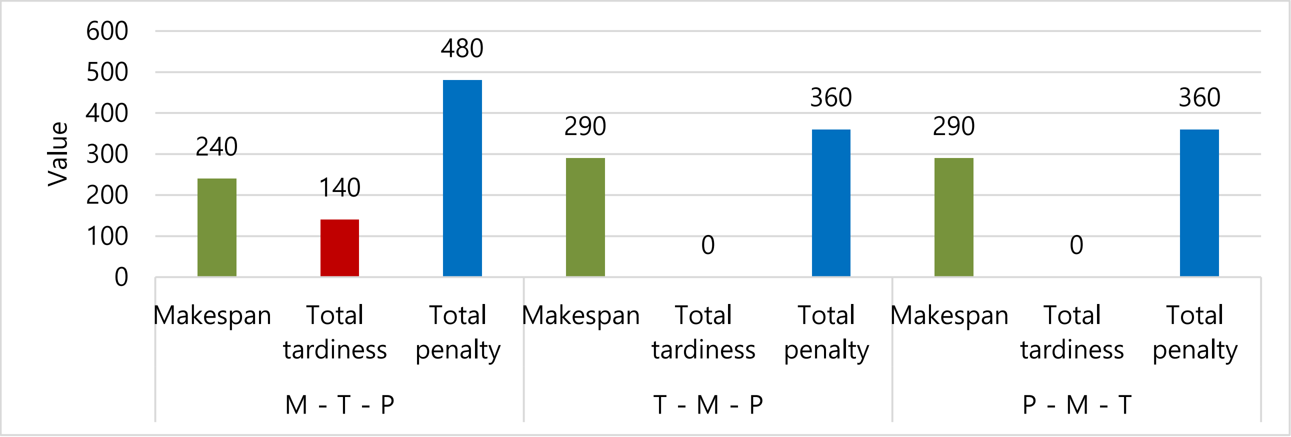


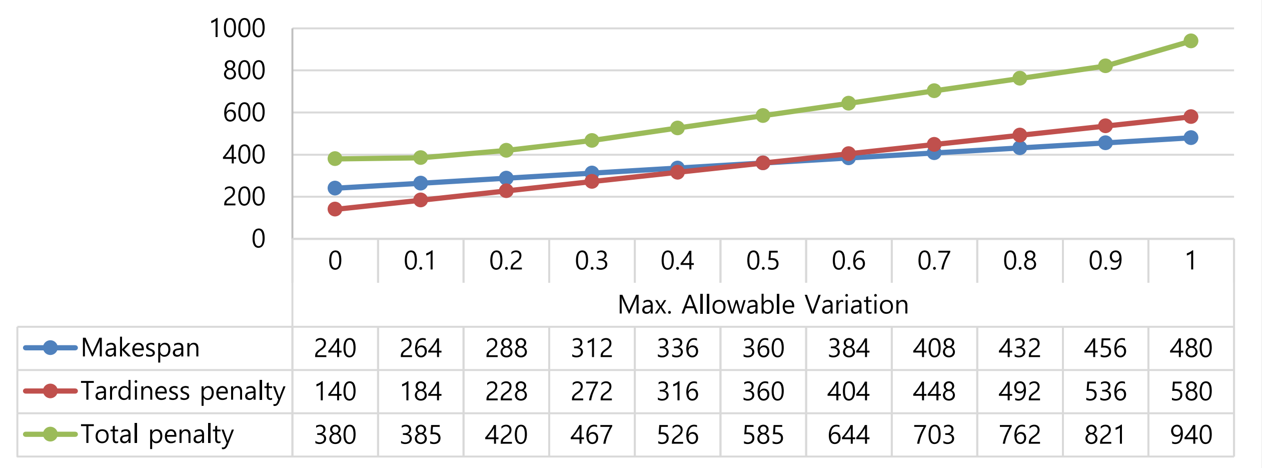
Fig. 3 Values of objectives at the different orders

Fig. 3 represents the optimum resulting values of the three objectives in the three orders. It is clear that optimization of only the makespan, without consideration of other performance factors like tardiness, earliness, or the idling of machines, and without an allowance for an amount of deviation, did not produce practically acceptable schedules.

It is concluded that the optimal ordering requires assigning priority to the minimization of the total penalty. The results obtained in the three problems provide clear evidence of the accuracy of the model.

3.2 Effect of the Maximum Variation of the Duration Times

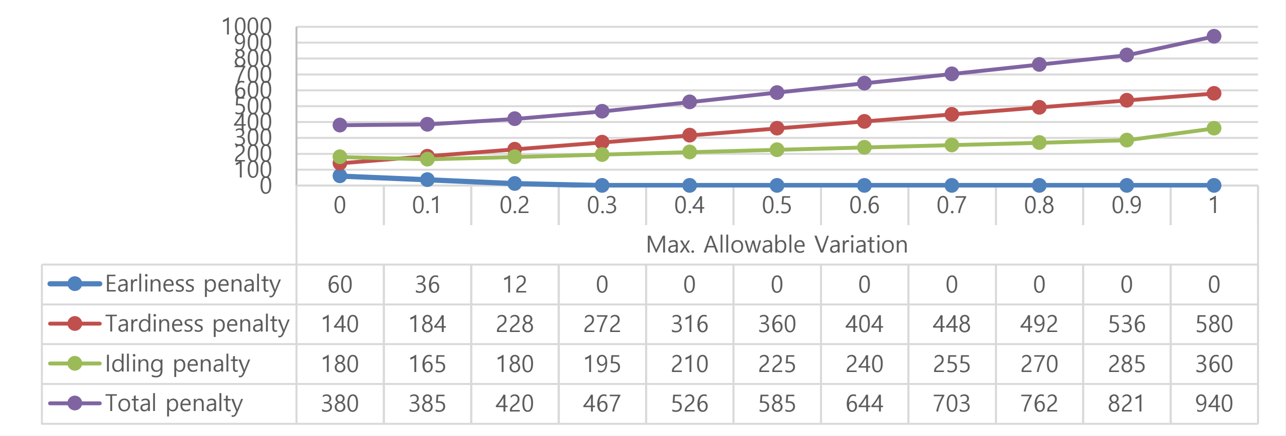
In this sub-section, the effects of the maximum variations of the duration times on the makespan, tardiness penalty, and total penalty are presented. The earliness, tardiness, and idling penalties per unit of time were each assumed to equal $1/unit of time to simplify the analysis. Fig. 4 shows that an increase in the variability of the processing time led to linear increases in both the makespan and the tardiness penalty. However, there was a brief period at the start of the process where the total penalty remained constant before it also followed a linear increase.



	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Makespan	240	264	288	312	336	360	384	408	432	456	480
Tardiness penalty	140	184	228	272	316	360	404	448	492	536	580
Total penalty	380	385	420	467	526	585	644	703	762	821	940

Fig. 4 Effect of the maximum variation of the duration times on the makespan, tardiness penalty, and total penalty

Fig. 5 illustrates the effects of the maximum variations of the processing times on the components of the total penalty. A decrease in the initial penalty is observed. The decrease in the earliness penalty was due to the increase of the makespan.

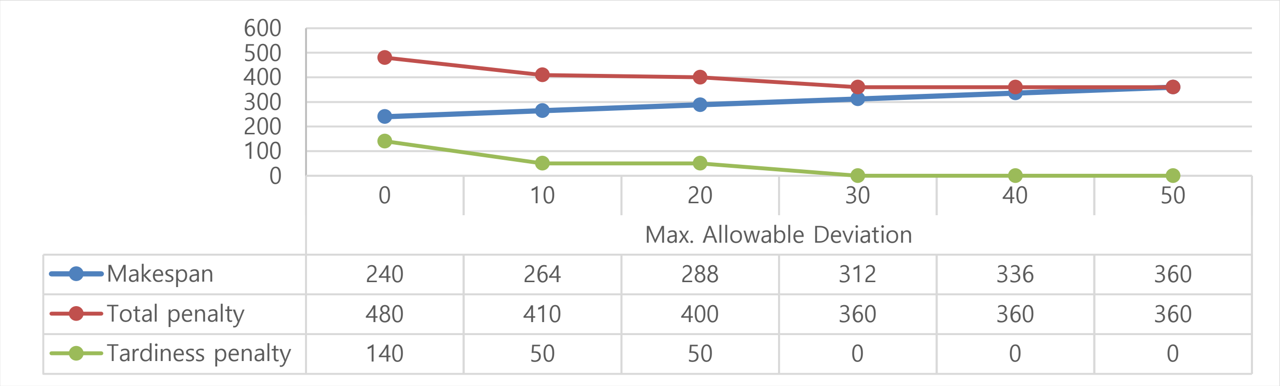


	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Earliness penalty	60	36	12	0	0	0	0	0	0	0	0
Tardiness penalty	140	184	228	272	316	360	404	448	492	536	580
Idling penalty	180	165	180	195	210	225	240	255	270	285	360
Total penalty	380	385	420	467	526	585	644	703	762	821	940

Fig. 5 Effect of the maximum variation of the components of the processing times on the total penalty

3.3 Effect of the Maximum Allowable Deviation

In this sub-section, the effects of the maximum allowable deviations of the prior objectives are presented. Objectives were ordered as follows: First the makespan, followed by the total penalty, followed by the tardiness penalty. Fig. 6 illustrates that an increase in the maximum allowable deviations increased the value of the makespan while decreasing the total penalty and the tardiness penalty.



	0	10	20	30	40	50
Makespan	240	264	288	312	336	360
Total penalty	480	410	400	360	360	360
Tardiness penalty	140	50	50	0	0	0

Fig. 6 Effect of the maximum allowable deviation of the prior objectives

4 Conclusions

In the very competitive modern industrial sector, even small gains in cost and time can be very valuable to a business. Therefore, the refinements introduced by this study and the results it presented could be of substantial benefit to certain process engineers.

The developed model successfully solved the JSSP to optimize multi-objectives simultaneously considering uncertain processing times. The developed model was implemented using Robust MILP, coded using the Mosel language and solved using the XpressIVE solver.

The accuracy and effectiveness of the model were verified by analyzing the obtained results. The model is capable of assessing many objectives beyond those presented in this study. A comparison was made between the results obtained in different cases.

The penalties were found to play an important role in addition to the makespan. The proposed model is suitable for working in just-in-time environments.

It is clear that optimization of only the makespan, without consideration of other performance factors like tardiness, earliness, or the idling of machines, and without an allowance for an amount of deviation, did not produce practically acceptable schedules.

It is concluded that the optimal ordering requires assigning priority to the minimization of the total penalty.

Future research work should consider the following aspects in the context of the model presented in this study:

- unexpected disruptions of machines
- the layout of the workshop
- the dynamic environment
- the time and cost of transportation

5 Declaration of Conflicting Interests

The author(s) declare that they have no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

[1] L. Lamorgese and C. Mannino, “A noncompact formulation for job-shop scheduling problems in traffic management,” *Operations Research*, vol. 67, no. 6, pp. 1586–1609, 2019.

[2] O. Sobeyko and L. Mönch, “Heuristic approaches for scheduling jobs in large-scale flexible job shops,” *Computers and Operations Research*, vol. 68, pp. 97–109, 2016, doi: 10.1016/j.cor.2015.11.004.

[3] B. Çaliş and S. Bulkan, “A research survey: review of AI solution strategies of job shop scheduling problem,” *Journal of Intelligent Manufacturing*, vol. 26, no. 5, pp. 961–973, 2015, doi: 10.1007/s10845-013-0837-8.

[4] A. Dabah, A. Bendjoudi, A. AitZai, D. El-Baz, and N. N. Taboudjemat, “Hybrid multi-core CPU and GPU-based B&B approaches for the blocking job shop scheduling problem,” *Journal of Parallel and Distributed Computing*, vol. 117, pp. 73–86, 2018, doi: 10.1016/j.jpdc.2018.02.005.

[5] G. Madraki and R. P. Judd, “Recalculating the length of the longest path in perturbed directed acyclic graph,” *IFAC-*

Manuscript received August 5, 2022
Manuscript revised August 20, 2022
https://doi.org/**10.22937/IJCSNS.2022.22.8.6**

PapersOnLine, vol. 52, no. 13, pp. 1560–1565, 2019, doi: 10.1016/j.ifacol.2019.11.422.

[6] A. Dabah, A. Bendjoudi, A. AitZai, and N. N. Taboudjemat, "Efficient parallel tabu search for the blocking job shop scheduling problem," *Soft Computing*, vol. 23, no. 24, pp. 13283–13295, 2019, doi: 10.1007/s00500-019-03871-1.

[7] G. Madraki and R. P. Judd, "Accelerating the calculation of makespan used in scheduling improvement heuristics," *Computers and Operations Research*, vol. 130, p. 105233, 2021, doi: 10.1016/j.cor.2021.105233.

[8] Y. N. Sotskov, "Mixed graph colorings: A historical review," *Mathematics*, vol. 8, no. 3, pp. 1–24, 2020, doi: 10.3390/math8030385.

[9] T. C. Lai, Y. N. Sotskov, N. Y. Sotskova, and F. Werner, "Optimal makespan scheduling with given bounds of processing times," *Mathematical and Computer Modelling*, vol. 26, no. 3, pp. 67–86, 1997, doi: 10.1016/S0895-7177(97)00132-5.

[10] T. C. Lai and Y. N. Sotskov, "Sequencing with uncertain numerical data for makespan minimisation," *Journal of the Operational Research Society*, vol. 50, no. 3, pp. 230–243, 1999, doi: 10.1057/palgrave.jors.2600690.

[11] M. S. Al-Ashhab, S. Munshi, M. Oreijah, and H. A. Ghulman, "Job Shop Scheduling Using Mixed Integer Programming," *International journal of Modern Engineering Research*, vol. 7, no. 3, 2017.

[12] M. S. Al-Ashhab, "Multi-Objective Job Shop Scheduling Using a Lexicographic Procedure," *International Journal of Engineering Science Invention*, vol. 7, no. 1, pp. 47–56, 2018.

[13] S. Singh and K. P. Singh, "Cuckoo Search Optimization for Job Shop Scheduling Problem," Das, K., Deep, K., Pant, M., Bansal, J., Nagar, A. (eds) Proceedings of Fourth International Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing, vol. 335, pp. 99–111, 2015.

[14] T. C. Lai, Y. N. Sotskov, N. Sotskova, and F. Werner, "Mean flow time minimization with given bounds of processing times," *European Journal of Operational Research*, vol. 159, no. 3, pp. 558–573, 2004, doi: 10.1016/S0377-2217(03)00424-7.

[15] F. T. S. Chan, T. C. Wong, and L. Y. Chan, "The application of genetic algorithms to lot streaming in a job-shop scheduling problem," *International Journal of Production Research*, vol. 47, no. 12, pp. 3387–3412, 2009.

[16] F. T. S. Chan, T. C. Wong, and L. Y. Chan, "Lot streaming for product assembly in job shop environment," *Robotics and Computer-Integrated Manufacturing*, vol. 24, no. 3, pp. 321–331, 2008, doi: 10.1016/j.rcim.2007.01.001.

[17] R. H. Huang, "Multi-objective job-shop scheduling with lot-splitting production," *International Journal of Production Economics*, vol. 124, no. 1, pp. 206–213, 2010, doi: 10.1016/j.ijpe.2009.10.026.

[18] J. S. Alzahrani, "Multi-objective Job Shop Scheduling Using Pre-emptive Constraint Procedure," *American Journal of Modeling and Optimization*, vol. 7, no. 1, pp. 8–13, 2019, doi: 10.12691/ajmo-7-1-2.

[19] A. J. Clewett, "Introduction to sequencing and scheduling," *Journal of the Operational Research Society*, vol. 28, no. 2, pp. 352–353, 1977.

[20] J. Otala, A. Minard, G. Madraki, and S. Mousavian, "Graph-based modeling in shop scheduling problems: Review and extensions," *Applied Sciences*, vol. 11, no. 11, p. 4741, 2021, doi: 10.3390/app11114741.

[21] Y. Mati, S. Dauzere-Pérès, and C. Lahlou, "A general approach for optimizing regular criteria in the job-shop scheduling problem," *European Journal of Operational Research*, vol. 212, no. 1, pp. 33–42, 2011, doi: 10.1016/j.ejor.2011.01.046.

[22] L. Lin, M. Gen, Y. Liang, and K. Ohno, "A hybrid EA for reactive flexible job-shop scheduling," *Procedia Computer Science*, vol. 12, pp. 110–115, 2012, doi: 10.1016/j.procs.2012.09.039.

[23] K. R. Baker and B. Keller, "Solving the single-machine sequencing problem using integer programming," *Computers & Industrial Engineering*, vol. 59, no. 4, pp. 730–735, 2010.

[24] A. Ponsich and C. A. Coello Coello, "A hybrid Differential Evolution - Tabu Search algorithm for the solution of Job-Shop Scheduling Problems," *Applied Soft Computing Journal*, vol. 13, no. 1, pp. 462–474, 2013, doi: 10.1016/j.asoc.2012.07.034.

[25] A. Udomsakdigool and V. Kachitvichyanukul, "Multiple colony ant algorithm for job-shop scheduling problem," *International Journal of Production Research*, vol. 46, no. 15, pp. 4155–4175, 2008.

[26] L. Gao, G. Zhang, L. Zhang, and X. Li, "An efficient memetic algorithm for solving the job shop scheduling problem," *Computers and Industrial Engineering*, vol. 60, no. 4, pp. 699–705, 2011, doi: 10.1016/j.cie.2011.01.003.

[27] 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, no. 6, pp. 609–629, 2015.

[28] R. Qing-Dao-Er-Ji and Y. Wang, "A new hybrid genetic algorithm for job shop scheduling problem," *Computers and Operations Research*, vol. 39, no. 10, pp. 2291–2299, 2012, doi: 10.1016/j.cor.2011.12.005.

[29] D. Y. Sha and C. Y. Hsu, "A hybrid particle swarm optimization for job shop scheduling problem," *Computers and Industrial Engineering*, vol. 51, no. 4, pp. 791–808, 2006, doi: 10.1016/j.cie.2006.09.002.

[30] T. L. Lin *et al.*, "An efficient job-shop scheduling algorithm based on particle swarm optimization," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2629–2636, 2010, doi: 10.1016/j.eswa.2009.08.015.

[31] P. A. N. Quanke, W. Ling, G. A. O. Liang, and S. Hongyan, "Differential evolution algorithm based on blocks on critical path for job shop scheduling problems," *Journal of Mechanical Engineering*, vol. 46, no. 22, pp. 182–188, 2010.

[32] S. Goren and I. Sabuncuoglu, "Robustness and stability measures for scheduling: Single-machine environment," *IIE Transactions (Institute of Industrial Engineers)*, vol. 40, no. 1, pp. 66–83, 2008, doi: 10.1080/07408170701283198.

[33] Z. Lu, W. Cui, and X. Han, "Integrated production and preventive maintenance scheduling for a single machine with failure uncertainty," *Computers and Industrial Engineering*, vol. 80, pp. 236–244, 2015, doi: 10.1016/j.cie.2014.12.017.

[34] M. S. Al-Ashhab, T. Attia, and A. Shaaban, "Penalty Minimization Job Shop Scheduling under Uncertainty," *International Journal of Business and General Management*, vol. 8, no. 3, pp. 23–38, 2019.

[35] B. Wang, X. Wang, F. Lan, and Q. Pan, "A hybrid local-search algorithm for robust job-shop scheduling under scenarios," *Applied Soft Computing Journal*, vol. 62, pp. 259–271, 2018, doi: 10.1016/j.asoc.2017.10.020.

[36] R. Tavakkoli-Moghaddam, F. Jolai, F. Vaziri, P. K. Ahmed, and A. Azaron, "A hybrid method for solving stochastic job shop scheduling problems," *Applied Mathematics and Computation*, vol. 170, no. 1, pp. 185–206, 2005, doi: 10.1016/j.amc.2004.11.036.

[37] P. A. Varthanan, N. Murugan, and G. M. Kumar, "A simulation based heuristic discrete particle swarm algorithm for generating integrated production-distribution plan," *Applied Soft Computing Journal*, vol. 12, no. 9, pp. 3034–3050, 2012, doi: 10.1016/j.asoc.2012.05.001.

[38] D. Golenko-Ginzburg and A. Gonik, "Optimal job-shop scheduling with random operations and cost objectives," *International Journal of Production Economics*, vol. 76, no. 2, pp. 147–157, 2002, doi: 10.1016/S0925-5273(01)00140-2.

M. S. Al-Ashhab received the B. E., M. E., and Ph.D. degrees from Ain Shams University in 1996, 2003, and 2008, respectively. After working as a research assistant (from 1996), an assistant professor (from 2008) in the Dept. of Design and Production Engineering, Ain Shams University, and an assistant professor and an associate professor (from 2011 and 2019) in the Mechanical Engineering Department, Umm Al-Qura University. His research interest includes SCM, Job shop Scheduling and Production planning.

JABER S. ALZAHRANI received the Ph.D. degree from Lamar University, in 2015. He is currently an Associate Professor at the Department of Industrial Engineering, College of Engineering at Al-Qunfudhah, Umm Al-Qura University, Saudi Arabia. He has many peer-reviewed articles. His research interests include optimization, supply chain, and scheduling