Hindawi
Complexity
Volume 2022, Article ID 1253336, 19 pages
https://doi.org/10.1155/2022/1253336

Research Article

Flexible Flow Shop Scheduling Problem with Reliable Transporters and Intermediate Limited Buffers via considering Learning Effects and Budget Constraint

Meysam Kazemi Esfeh, Amir Abbas Shojaie, Hasan Javanshir, and Kaveh Khalili-Damghani

Department of Industrial Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran

Correspondence should be addressed to Amir Abbas Shojaie; a.shojaie@azad.ac.ir

Received 20 July 2022; Revised 22 August 2022; Accepted 7 September 2022; Published 26 September 2022

Academic Editor: Reza Lotfi

In this study, a new mathematical model is presented to solve the flexible flow shop problem where transportation is reliable and there are constraints on intermediate buffers, budgets, and human resource learning effects. Firstly, the model is validated to confirm the accuracy of its performance. Then, since it is an NP-hard one, two metaheuristic algorithms, namely, MOSA and MOEA/D, are rendered to solve mid- and large-scale problems. To confirm their accuracy of performance, two small-scale problems are solved using GAMS exact solution software, and the obtained results have been compared with the output of the algorithms. Since the problem in this study is multiobjective, five comparative indices are used to compare the performance of algorithms. The results show that the answers achieved using the metaheuristic algorithms are very close to the ones achieved via the GAMS exact program. Therefore, the proposed algorithms are validated, and it is proved that they are accurately designed and useable in solving the real-world problems (which have mid- and large-scale) in logical calculation time. By comparing the obtained results, it can be seen that the MOEA/D algorithm performs better in terms of computational time (CPU time) and Mean ideal distance (MID). The MOSA algorithm also performs better according to the index Spread of nondominated solutions (SNS), diversity metric (DM), and number of Pareto solutions (NPS). Considering the confirmation of precision and accuracy of performance of the proposed algorithms, it can be concluded that MOSA and MOEA/D are useful in solving the mid- and large-scale modes of the problem in the study, which is very applicable in the real world.

1. Introduction

Scheduling is the gathering of principles, models, and method for making decisions and determining a time schedule. To do so, it is necessary to define the problem, its dimensions, and objectives accurately. Resources and activities are defined in a system in various ways. Resources can be machinery in a production workshop, air lines in an airport, worker in a construction project, and so on. If the process is done only in one stage and there is only one processor in it, the problem is a single machine mode one. Sometimes, more than one processor is used in a parallel way for optimizing the efficiency and speed of processing. In such a situation, the sequence of operation problem transforms into a problem with parallel machines [1, 2].

The flexible flow shop problem includes a layout of numerous machines in a linear way so that production begins in one in the initial work stage and finishes in the last one in the last stage. All productions must be put inside one machine in all work stages and must be passed through production and process stages one by one until the last stage [3]. The sequence of operation problem can be categorized based on the way of processing and the placement and characteristics of processors. If a process is done only in one stage, and there is only one processor in it, the problem is a single machine mode one. Sometimes, more than one

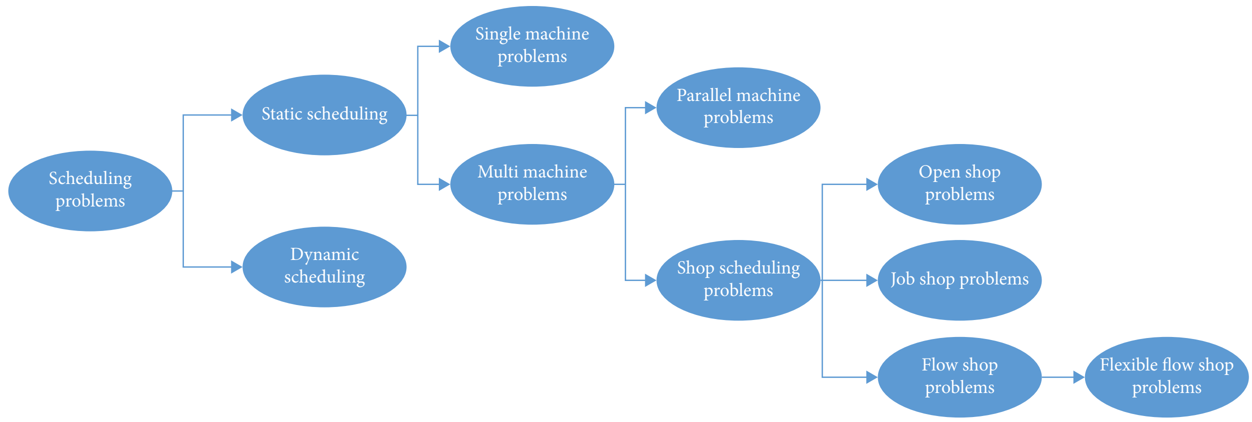

Figure 1: The categorization of the position of the flexible flow shop problem.

processor is used in a parallel way for optimizing the efficiency and speed of processing. In such a situation, the sequence of operation problem transforms into a problem with parallel machines. A more complicated mode occurs when data must be processed in numerous consecutive stages. In this mode, machines are placed in a series way, and each activity must be performed on all machines, respectively. The sequence is the same for all activities. Such a problem is called a flexible flow shop one [4, 5]. Sometimes, in flow shop problems, check points occur, decreasing their efficiency. It is because processing takes a lot of time in some stages. As a result, numerous processors are used in a parallel way instead of just one to fasten and facilitate processing. Such problems with such characteristics (numerous processing stages in a series way, many processors in a parallel way in each stage, and the identical processing for all activities) are called flexible flow shop ones. The flexible flow shop problem includes a layout of numerous machines in a linear way so that production begins in one in the initial work stage and finishes in the last one in the last stage. All productions must be put inside one machine in all work stages and passed through production and process stages one by one until the last stage. The objective of such problems is the quantification of the time of processing all jobs on machinery, so that all jobs are processed in the shortest time. As mentioned earlier, in this study, the main problem is studied under the assumption that parameters and data were predetermined. Figure 1 shows a comprehensive categorization of the sequence of operations and the current flexible flow shop problem position.

In this study, a scenario-based multiobjective problem is focused on in a production system based on the flexible flow shop. The probabilities of each scenario are determined. In each of them, the costs of each machine work and the rate of transportation failures are different in every stage. In the model, the flexible flow shop is presupposed. In all stages of the flow shop, a lot of transportation is done for transporting products and processing on machines. Each time the work is moved by the transporter between each station, it has its own transportation time. The preparation times of each job at each station depend on the previous work done at that station. Performing preventive maintenance and repair operations is another operation that can be carried out on work transport devices between workstations. In this situation, each transportation device becomes defective based on its relative failure distribution function and need for fixing. At this time, operations or productions remain on transportation devices until transformation operations finish and the device becomes available. Afterwards, transportation continues. Repairing the transportation devices is done randomly, and each problematic device will be repaired based on a probability distribution function. Regarding the constraints in the number of transportation devices among stages, intermediate buffers with limited capacities are specified for each work stage. Also, each human resource has a learning rate regarding each machine. In addition, the time of processing and preparing decreases based on human resource learning rates. The objective is the quantification of the mean of completion time value at the end of stages and the penalty of waiting time in buffer and also the cost of processing based on jobs' specification. The number of works performed in intermediate buffers is fewer than the maximum capacity of the relative accumulator. Another constraint affecting the proposed flexible flow shop problem is the budget. The budget limit in this issue is applied in such a way that the total budget available for the maintenance costs and staying of the works in the in intermediate buffers before the workstations is a limited and specific budget. Each work waiting for processing on work shop stages' previous machines in intermediate buffers has a certain cost per time units in the problem, and for all works in the intermediate buffers of each work stage, a maintenance cost is determined for the model. The total cost must be less than the total budget of the project.

In the second section, the review of literature is presented. In the third section, the mathematical model is introduced, and the results will be presented in the fourth section. At the end, conclusions and suggestions are mentioned.

2. Literature Review

The first research regarding the flexible flow shop problem was presented in 1970s by [6], in which the modeling of the production system in a textile supplying factory is studied. After the introduction of this branch of sequence of operation, researchers became gradually interested in it and published numerous articles [7].

Johnson [8] proved that the problem of flexible flow shop in which the function of the quantification of the make span is under the set of NP-hard ones. As a result, researchers were encouraged to render various heuristic and metaheuristic methods to approximately solve problems. Tavakkoli-Moghaddam et al. [9] proposed a two-way model of the work flow problem of parallel machines considering independent initial set up times and the precedence constraint and used a genetic algorithm to solve it.

Naderi et al. [10] presented two algorithms based on electromagnetic one and simulated annealing one to solve the two-objective problem of job shop flow. The objectives were quantifying the completion total time of jobs and the whole delays. The results show that the electromagnetic algorithm had better answers compared with other methods. Wang and Choi [11] presented a multiobjective model of parallel machine problem considering the maintenance operation and preventive multi fixings. The objective function included the quantification of the completion time of all jobs; the first case is the unreliability of machines and the second case is the unreliability of preventive maintenance and repairs. To solve the problem, they used a nondominance sorting genetic algorithm (NSGA). Shahidizadeh et al. [12] presented a two-objective model of a job shop flow with parallel machine, in which the process was a group production one. The objective function included the quantification of the completion time of all jobs, the reduction of delay, and the cost of purchasing machinery. Since their problem was NP-hard and had large-scale, they used a multiobjective harmony search algorithm to solve it. The results showed that their proposed method rendered more qualified answers compared with other algorithms. Liao et al. [13] presented a multiobjective model of the group scheduling problem in the job shop flow system considering the maintenance operation and preventive repairing. The objective function included the quantification of the completion time of all jobs, maintenance costs, and preventive repairing. Zandieh et al. [14] presented a multi objective model of hybrid job shop flow considering the maintenance operation and preventive repairing. Their solution included a hybrid dominance sorting genetic algorithm with two approaches. The results showed that their proposed method was very efficient in hybrid job shop flow problems. Mollaei et al. [15] presented a two-objective model for multistage problems in production systems. To solve it, they used a hybrid of Taguchi and Monte Carlo methods. Their study was about an automotive company in France. The results proved the quality of their solution in the mentioned study. Ozsoydan and Sagir [16] proposed an integrated Greedy algorithm enhanced by metaheuristic-based learning for flexible flow shop problem, considering independent initial set up times. Cheng et al. [17] presented a group scheduling problem in a job shop flow one. They presented two algorithms of multistage simulated annealing and local search-based variance ones. Also, by solving the problem in the previous literature and studies using their method, they indicated that most local search-based variant algorithms render better results, both considering the objective function and computational time. Shen et al. [18] presented a two-objective model for the flexible flow shop problem in the macaroni production industry in Belgium. The objective function of his study included quantifying the time of job completion and the energy and human resources costs. They proposed a customized two-phase genetic algorithm. The results of solving their proposed problem indicated that their algorithm renders better results compared with the Pareto power-based evolutionary one. Pagnozzi and Stuzle [19] rendered a hybrid probability local optimization algorithm to solve their proposed job shop flow problem. The objective function included quantifying the time of job completion and the total of delays.

Lotfi et al. [20] have researched renewable energy. The most important innovations in their study included the use of robust two-level programming techniques and game theory (Stackelberg Competition) for locating renewable energy sites. The results show that the combination of uncertainties can increase energy production and supplier profits. In addition, the objective functions of the proposed model are compared with those under uncertain conditions. The sensitivity analysis of the main parameters is performed to validate the proposed model. As uncertainty increases, the energy produced decreases, and the supplier's profit increases. Supplier profits gradually decrease as the discount rate increases. In addition, as the scale of the problems increases, the energy produced and the profit of the supplier increase.

According to the above-mentioned studies, a comprehensive categorization of studies on the flexible flow shop is summarized in Figure 2. The problem in this study is a scenario-based flexible flow shop one in which reliable transportation devices and intermediate limited buffers are considered in addition to budget constraints, dependent setup time, the learning effect, and also the objective functions of "the mean of completed time weight value," "the mean of the penalty of waiting time in buffer," and the total specified processing costs. Reviewing the literature showed that the problem in this study has not yet been focused on in any research and is completely new.

3. Methodology

In this study, a multiobjective scenario-based problem that exists in a production system is presented. The system is based on a flexible flow shop. In this model, the flexible flow shop problem is presupposed, and inside each job stage, there are some devices for transportation, each having its own specific times. Preventive repairing and maintenance is one of the other operations performed on transportations used in transporting productions. In this situation, each transportation device breaks down based on its relative distribution failure function and need for fixing. At this time, operations remain on transportation devices until transformation operations finish and the instrument becomes available. Afterwards, transportation continues. Repairing the transportation devices is done randomly, and each problematic device will be repaired based on a probability

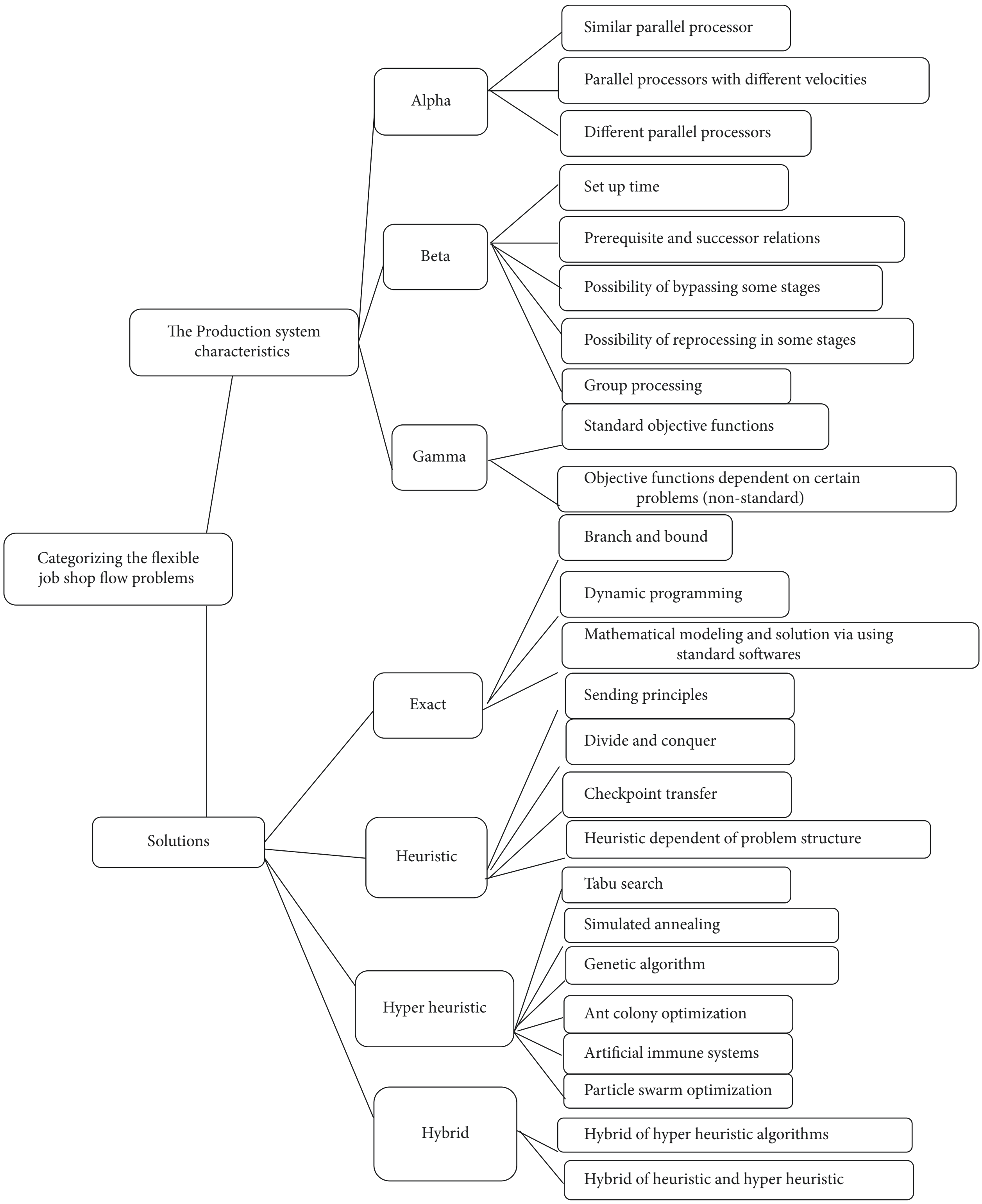

Figure 2: The categorization of studies on flexible flow shop problem.

distribution function. Regarding the constraints in the number of transportation instruments among stages, intermediate buffers with limited capacities are specified for each work stage. The objective is the quantification of the mean of completion time value at the end of stages, so that all productions are transported by transportation devices in job stages and are put on the machinery of all stages. The number of jobs performed in intermediate buffers is lesser than the maximum capacity of the relative accumulator. Another constraint affecting the proposed flexible flow shop

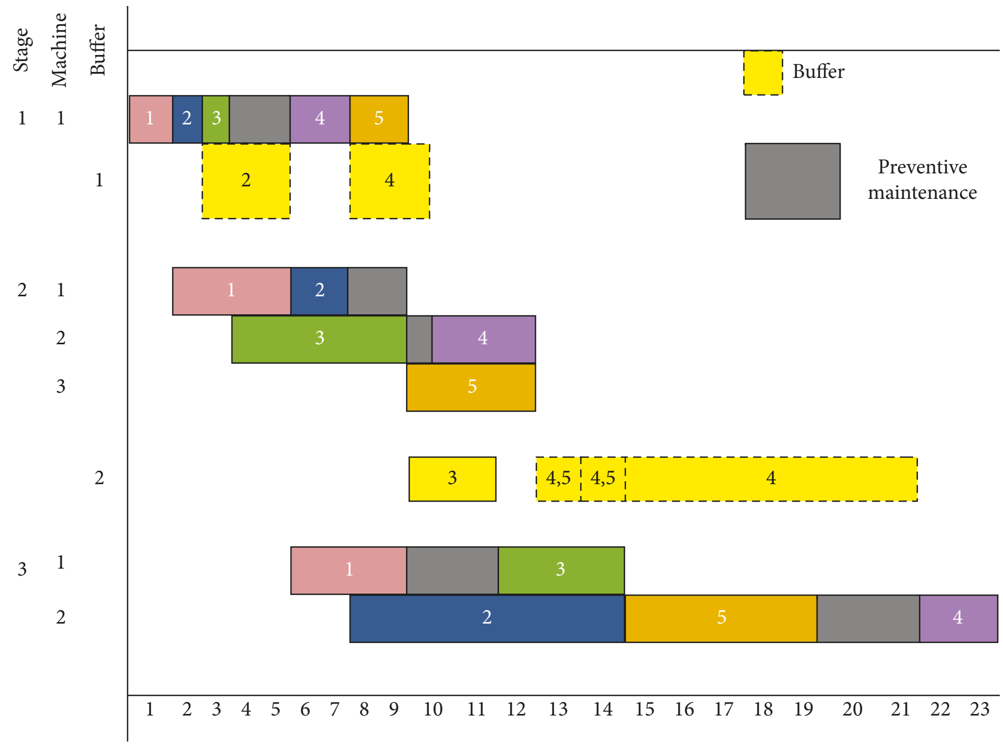

Figure 3: A sample of the Gantt chart of the proposed flexible flow shop problem.

problem is regarding the finances. This kind of constraint is applied in this problem in the following manner: the total of finances is limited and definite for maintenance and remaining productions and operations in intermediate buffers. Each job operation waiting for processing on work shop stages' previous machines in intermediate buffers has a certain cost per time units in the problem, and for all jobs in the intermediate buffers of each job stage, a maintenance cost is determined in the model. The total cost must be less than the total budget of the project.

In the following section, the mathematical model and all its parameters and variables will be explained in detail. Considering Figure 3 of the model, suppose that the job j is being processed in the stage S. In each stage, N_s number stages are placed in a parallel way, and all jobs must be processed on one machine in each of job stages. Preventive maintenance and repairing operation can be performed on all machineries in job stages. Also, there is an intermediate limited buffer where jobs have zero start time or end time. According to Figure 3, which is a proposed Gantt Chart sample of the problem with three job stages and 5 productive machines, jobs 1, 2, and 3 are in the work stages 1, 2, and 3, respectively, with machines. As observed, on machine 1 in work stage 1, the preventive maintenance operation is done after the processing of job 3 was completed. Another preventive maintenance operation was done on job stages 2 and 3. Job 2 was done in the job stage 1 between times 2 and 3. It started from time 6 in the job stage 2. Therefore, this job remains in the intermediate buffer of job stage 1 until the process in job stage 2 begins. If there is no capacity to accept job 2 in the intermediate buffer, this job will remain in the machine, and as a result, it will block. In some cases, it is possible that more than one job is stored in the intermediate buffers. For instance, in the intermediate buffer of job stage 2, jobs 4 and 5 are stored between times 13 and 14.

The mathematical model used in this study is derived from the one in Zabihzadeh and Rezaeian [21]. This model is developed here. This model includes two objective functions and 17 constraints. The variables in this study are of positive and binary types. There are 6 positives and 2 negatives. All indices, parameters, and variables (whether dependent, independent, or control) are defined in the basic model.

Assumptions and constraints in this study are as follows:

(i) All jobs j that are indicated by $J = \{j | j = 1, 2, \ldots, J\}$ are available at the start of jobs (time-zero) and no job stops before final production.

(ii) The job stage groups are indicated by S, defined as $S = \{s | s = 1, 2, \ldots, S\}$. In each job stage, there are some parallel machineries shown by M, defined as $M = \{m | m = 1,2, \ldots, M\}$ and $j \in J$.

(iii) The machinery group shown by j is placed in a parallel way in each job stage to be processed and used for producing materials.

(iv) All productions must be put on all job stages and specified on each of the stages for processing and producing to be done on them. In other words, when processing begins on a job in one job stage, it must continue onstage by stage until the last one.

(v) Each job has a preparation time depending on the previous one.

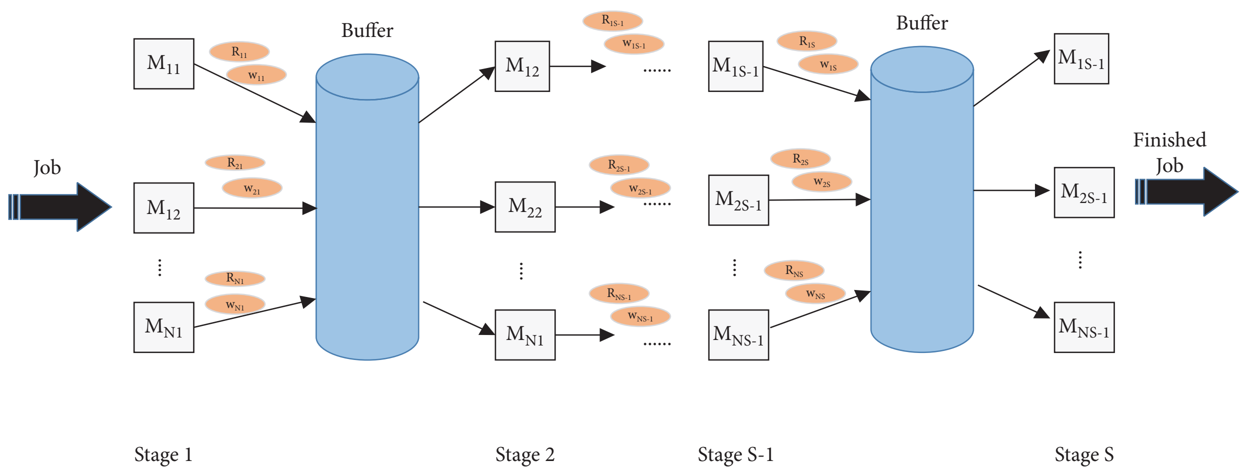

FIGURE 4: Schema of the problem of flexible flow shop.

(vi) Each human resource has a learning rate regarding each machine, based on which the processing and preparing time decreases.

(vii) Inside each job stage, there are some transportation devices to transport jobs. All of them are available at the start of the project. In other words, all of them work perfectly to begin with.

(viii) The failure of the transportation devices occurs randomly after the first job begins and based on the probability distribution function.

(ix) Each job stage has an intermediate limited buffer, which is indicated by K, defined as $K = k|k = 1, 2, \ldots, K\}$

(x) Each production machine only does one job in a specified time.

(xi) Each job is defined only once on a certain machine for production and process. It means that the repetition of jobs is impossible.

(xii) The processing of jobs in the next stage occurs when production and process are finished in the previous one.

(xiii) The storage capacity of jobs in each intermediate buffer does not depend on job types.

The schema of the problem of flexible flow shop in this study can be seen in Figure 4.

3.1. Sets. J: the set of jobs (productions)

O: the set of sequences

S: the set of stages (stages)

T: the set of time

K: the set of scenarios

R: the set of transportation devices

M: the set of machines

W: the set of human resources

3.2. Indices. $j \in J$: index of the set of jobs

$o \in O$: index of the set of sequences

$s \in S$: index of the set of stages (stages)

$t \in T$: index of the set of time

$k \in K$: index of the set of scenarios

$r \in R$: index of the set of transportation devices

$m \in M$: index of the set of machines

$w \in W$: index of the set of human resources

3.3. The Parameters

β_j: the importance coefficient of the completion of the j^{th} job

P_k: the probability of the occurrence of k^{th} scenario

γ_{jmk}: the penalty (cost) of the j^{th} job for the m^{th} machine under the k^{th} scenario

RR_{jmst}: the cost of j^{th} job by m^{th} machine during t period in the s stage

M_s: if machines belong to stage s, they are equal to 1, if not, they are equal to 0

F: the budget constraint (maximum of investment) in each t time period

ws_{ws}: if human resource w belongs to stage s, it is equal to 1, if not, it is equal to 0

α_{wm}: the coefficient of human resource w learning for the m^{th} machine

$e^{-t.\alpha_{wm}}$: the coefficient of the decrease of processing and preparation time based on the human resources learning rate

μ_{rsk}: the amount of breakdown of the r transportation device in s stage under the k scenario

λ_{rs}: the amount of repairing of transportation devices in the s stage

TT_{jrst}: the time needed for transportation in r machine from the starting point (the previous stage machine or buffer) to the final point (the previous stage machine or buffer)

Tr_{rs}: the time for preventive repairing of the r transportation device in s stage

P_{jms}: the time for processing of the j^{th} job by the m^{th} machine in the s stage

ST_{jjrs}: the time needed for the preparation of j job done after j^{th} one in s stage

CAPDE_t: the time capacity of waiting in buffer in t period

3.4. The Positive Variables

Ct_{jmstk}: the time of completion of the j^{th} job in the m^{th} machine in s stage in the t time period under the k scenario.

DE_{jmstk}: the waiting time of the j^{th} job, which is supposed to be processed by the the m^{th} machine in s stage in the t time period under the k scenario.

SST_{jmstk}: the time of start of the j^{th} job in the m^{th} machine in s stage in the t time period under the k scenario.

TTA_{rjstk}: the expected time for the transportation of the j^{th} job based on the availability of the r^{th} transportation device in the t period in the s stage under k scenario.

CTS_{jstk}: the time of completion of the j^{th} job in the previous s stage in the t time period under the k scenario.

A_{rstk}: the availability of r transportation devices in the s station in the t^{th} time period under the k^{th} scenario.

3.5. Binary Variables

X_{jomswt}: if the j^{th} job in the o sequence is specified to the m^{th} machine and the w human resources in the s stage in the t^{th} time period, it equals 1. Otherwise, it is zero.

U_{jrtk}: if the r^{th} transportation device is specified to the j^{th} job in the t^{th} time period under the k scenario, it is 1. Otherwise, it is zero.

3.6. The Objective Function and the Constraints of the Problem.

In this section, the suggested mathematical model is presented as follows:

$$\text{of}_2 = \sum_{j\in J}\sum_{m\in M}\sum_{s\in S}\sum_{t\in T}\sum_{k\in K}\left(\frac{\beta_j * ct_{jmstk}}{\sum_{j'\in J}\beta j\prime}\right) * P_k. \tag{1}$$

The first objective function calculates the mean of the time weight value at the end of a stage as follows:

$$\begin{aligned}\text{of}_2 &= \sum_{j\in J}\sum_{m\in M}\sum_{s\in S}\sum_{t\in T}\sum_{k\in K}\left(P_k * \gamma_{jmk} * DE_{jmstk}\right)\\ &\quad+\sum_{j\in J}\sum_{s\in S}\sum_{m\in M}\sum_{t\in T}\sum_{w\in W}\left(RR_{jmst} * X_{jomswt}\right).\end{aligned} \tag{2}$$

The second objective function shows the penalty of waiting time on buffer and the cost of the j^{th} job (production) process based on its specification.

Subject to

$$\sum_{o\in O}\sum_{m\in M_s}\sum_{w\in W_s} X_{jomwst} = 1\forall j \in J\cdot s \in S\cdot t \in T, \tag{3}$$

$$\sum_{j\in J}\sum_{m\in M_s}\sum_{w\in W_s} X_{jomwst} = 1\forall o \in O\cdot s \in S\cdot t \in T, \tag{4}$$

$$\sum_{o\in O}\sum_{m\in M_s} X_{jomwst} \leq 1\forall j \in J\cdot s \in S\cdot t \in T, \quad w \in W_s, \tag{5}$$

$$\sum_{j\in J} X_{j(o+1)mswt} \leq \sum_{j\in J} X_{jomswt}\forall o\in O\cdot m\in M_s\cdot s\in S\cdot t\in T\cdot w\in W_s, \tag{6}$$

$$A_{rstk} = \frac{\mu_{rsk}}{\mu_{rsk}+\lambda_{rs}} + \frac{\lambda_{rs}}{\mu_{rsk}+\lambda_{rs}} * e^{-\left[\mu_{rsk}+\lambda_{rsk} * \left(t-Tr_{rs}\right)\right]}\forall r\in R\cdot s\in S\cdot t\in T\cdot k\in K, \tag{7}$$

$$\text{TTA}_{rjstk} \geq \text{TT}_{jrst} * A_{rstk} - M\left(1-u_{jrtk}\right)\forall r\in R\cdot j\in J\cdot s\in S\cdot t\in T\cdot k\in K, \tag{8}$$

$$\sum_{r\in R} u_{jrtk} = 1 \forall j \in J \cdot t \in T \cdot k \in K, \tag{9}$$

$$\sum_{o\in O}\sum_{j\in J}\sum_{m\in M}\sum_{s\in S} \mathrm{RR}_{jmst} \cdot X_{jomswt} \leq F \forall t \in T, \tag{10}$$

$$ct_{jmstk} \geq P_{jms} * e^{-t*\alpha_{wm}} + \sum_{r\in R} \mathrm{TTA}_{rjstk} - M\left(1 - X_{jomswt}\right) \forall j \in J \cdot m \in M \cdot s \in S \cdot t \in T \cdot o \in O \cdot w \in W \cdot k \in K; O = 1 \cdot S = 1, \tag{11}$$

$$\begin{gathered} ct_{jmstk} \geq P_{jms} * e^{-t*\alpha_{wm}} + \mathrm{SST}_{\mathrm{jmstk}} - M\left(1 - X_{jomswt}\right) \\ \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot o \in O \cdot w \in W \cdot k \in K; \quad O = 1 \cdot S > 1, \end{gathered} \tag{12}$$

$$\mathrm{SST}_{jmstk} \geq cts_{j(s-1)tk} + \sum_{r\in R} \mathrm{TTA}_{rjstk} - M\Bigg(1 - \sum_{w\in W} X_{jomswt}) \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot o \in O \cdot k \in K; \quad S > 1, \tag{13}$$

$$\begin{aligned} \mathrm{SST}_{j'mstk} \geq & \sum_{r} \mathrm{TTA}_{rj'stk} + ct_{jmstk} \\ & -M\Bigg(1 - \sum_{w\in W} X_{j(o-1)mswt} + 1 - \sum_{w\in W} X_{j'omswt}) \forall j' \cdot j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot o \in O \cdot k \in K; \quad O > 1, \end{aligned} \tag{14}$$

$$\mathrm{SST}_{jmstk} \leq M * \sum_{o\in O}\sum_{w\in W} X_{jomswt} \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot k \in K, \tag{15}$$

$$cts_{jstk} = \sum_{m\in M} ct_{jm(s-1)tk} \forall j \in J \cdot s \in S \cdot t \in T \cdot k \in K, \tag{16}$$

$$ct_{j'mstk} \geq \Bigg[P_{j'ms} + \sum_{j\in J}\sum_{w'\in W} \mathrm{ST}_{jj's} * X_{j(o-1)w'mst}]\left(e^{-t.\alpha_{wm}}\right) + \mathrm{SST}_{j'^{mstk}} - M\left(1 - X_{j'omswt}\right)$$

$$\forall j' \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot o \in O \cdot k \in K \cdot w \in W; \quad o > 1, \tag{17}$$

$$ct_{j'mstk} \leq \Bigg[P_{j'ms} + \sum_{j\in J}\sum_{w'\in W} \mathrm{ST}_{jj's} * X_{j(o-1)w'mst}]\left(e^{-t.\alpha_{wm}}\right) + \mathrm{SST}_{j'^{mstk}} + M\left(1 - X_{j'omswt}\right)$$

$$\forall j' \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot o \in O \cdot k \in K \cdot w \in W; \quad o > 1. \tag{18}$$

Table 1: The function of random distribution of parameters amounts.

$\gamma_{jmk} = U(25, 40)$	$\mu_{rsk} = U(0.1, 0.3)$	$p_{jms} = U(2, 4)$
$\beta_j = U(0.2, 0.4)$	$\lambda_{rs} = U(0.4, 0.7)$	$\mathrm{TT}_{jrst} = \mathrm{round}(U(1, 15))$
$\mathrm{CAPDE}_t = U(10, 20)$	$\alpha_{wm} = U(0.1, 0.3)$	$p_k = U(0.1, 0.8)$
$Tr_{rs} = \mathrm{round}\ (U(1, \lvert T\rvert - 1))$	$\mathrm{RR}_{jj't} = \mathrm{round}(U(500, 700))$	$\mathrm{ST}_{jj't} = \mathrm{round}(U(1, 10))$
	$F = \mathrm{round}\ (U(8000, 9500))$	

$$\mathrm{DE}_{jmstk} \geq \left[\mathrm{SST}_{j'mstk} - ct_{jm(s-1)t,k}\right] - M\left(1 - \sum_{o\in O} X_{jomswt}\right) \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot k \in K;\quad s > 1, \tag{19}$$

$$\mathrm{DE}_{jmstk} \leq \left[\mathrm{SST}_{j'mstk} - ct_{jm(s-1)t,k}\right] - M\left(1 - \sum_{o\in O} X_{jomswt}\right) \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot k \in K;\quad s > 1, \tag{20}$$

$$\mathrm{DE}_{jmstk} \leq M\left(\sum_{o\in O}\sum_{w\in W} X_{jowmst}\right) \forall j \in J \cdot m \in M_s \cdot s \in S \cdot t \in T \cdot k \in K;\quad s > 1, \tag{21}$$

$$\sum_{j\in J}\sum_{m\in M}\sum_{s\in S} \mathrm{DE}_{jmstk} \leq \mathrm{CAPDE}_t \forall t \in T,\quad k \in K. \tag{22}$$

The constraint (3) specifies each job (production) to one sequence and one machine of each stage. Constraint (4) guarantees that each sequence must be specified to one job in each stage. Constraint (5) guarantees that each human resource can only be specified to one machine. The constraint (6) guarantees that the next sequence can only be specified only when the previous one is done. Constraint (7) calculates the availability of the transportation device of each time period under any scenario. Constraint (8) calculates the mean of transportation time based on the availability of transportation devices. Constraint (9) determines the type of the transportation device in order to transport the j^{th} job in the t^{th} time period. Constraint (10) determines the maximum of investment for each time period. Constraint (11) calculates the time of completion of each job in the first stage and for the first sequence. Constraint (12) calculates the time of completion of each job in the stages, except the first one, and for the first sequence. Constraint (13) calculates the time of starting of the j^{th} job as soon as it is completed in the previous stage. Constraint (14) calculates the time of j'^{th} job start based on the completion of the previous one in the previous stage and the previous stage in the same machine. Constraint (15) guarantees that if X_{jomswt} becomes zero, the time of completion will become zero too. Constraint (16) calculates the time of completion of the j^{th} job in the previous stage. Constraints (17) and (18) calculate the completion times of jobs in the same machine in each stage, except for sequence 1. Constraints (19)–(21) calculate the waiting time of the j^{th} job in the buffer before the s station under k scenario in t time period. Finally, constraint (22) controls the limitation of the waiting time in the buffer.

4. The Numerical Results

Since the mathematical model in this study is an NP-hard one, to solve problems of the real world, two metaheuristic algorithms, namely MOSA and MOEA/D, are used to in this section, and the results will be analyzed and compared. To evaluate the quality and data sparsity of the multiobjective metaheuristic algorithms, there are various metrics, which are different from the ones of the one objective metaheuristic ones. The reason is that there are answers with no priority over each other. The metrics used in this study are as follows: the number of Pareto solutions (NPS), diversification metric (DM), mean ideal distance (MID), spread of nondominance solutions (SNS), and computational time (time). To make these comparisons, 10 experimental problems are made, which are small, medium, and large-scale, and the results of their solutions via algorithms are mentioned in tables. Based on Table 1, the parameters of the problem are randomly made and used as primary data for problems.

The primary data of the problem are as follows: process time, transportation time, probability of each scenario, preparation time, the job costs, the amount of breakdown and repairing of the transportation devices, the learning coefficient of human resources, the penalty (cost) of jobs, the coefficient of the importance of jobs, the maximum investment in each time period, and the time for preventive

Table 2: The levels of experiments on parameters in the MOSA algorithm.

Algorithm parameters		First level	Second level	Third level
Maximum of iteration	MaxIt1	90	100	120
Maximum of internal iteration	MaxIt2	40	60	80
Primary temperature	T0	10	20	30
The coefficient of heating	Alpha	0.89	0.9	0.98
The maximum capacity of answer archive	nRep	10	15	20
Number of grids in each dimension	nGrid	4	8	10
The leading choice coefficient	Beta	1	2	3
The choice coefficient for deleting archive	Gamma	2	3	4

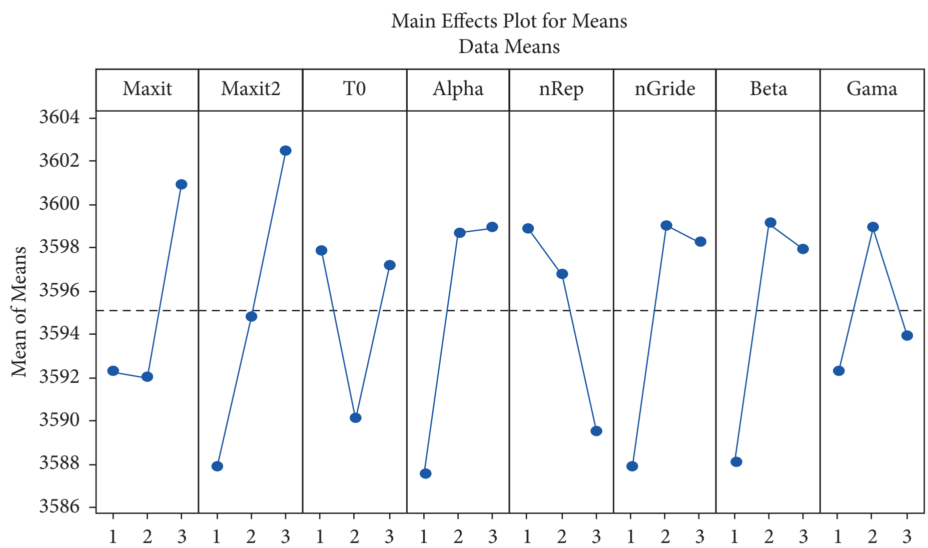

Figure 5: The solution of means for MOSA algorithm.

repairing. The distribution function of each of them are mentioned in Table 1.

4.1. Adjusting the Parameters for the Solution Algorithms. Adjusting parameters is vital for the efficiency of metaheuristic algorithms. Therefore, determining apposite parameters for metaheuristic algorithms is an important step in making them. Therefore, a set of calibration experiments are usually performed for finding the optimized mode of various amounts of control parameters of algorithms. Since the increase in test levels and the number of factors leads to an exponential increase in time and cost, the Taguchi method is used to perform tests. For the two metaheuristic algorithms used in this study, the control parameters are as follows: in the case of the MOSA algorithm, the parameters are the maximum of iteration (MaxIt1), maximum of internal iteration (MaxIt2), primary temperature (T0), the coefficient of heating (Alpha), the maximum capacity of answer archive (nRep), number of grids in each dimension (nGrid), the leading choice coefficient (Beta), and the choice coefficient for deleting archive (gamma). In the case of MOEA/D, parameters include the maximum of iteration (MaxIt), number of population (nPop), maximum number of archive (nArchive), and the coefficient of creating harmony in the intersection operator (gamma). Each parameter considered for the proposed algorithms has a salient effect on the quality of answers and time. For example, if a large number of population is considered for MOEA/D, the algorithm will have a long range of answers, and consequently, it can take a long time to achieve them. If a small number of populations is considered, the optimized answer could be locally optimized. Regarding the experiments and results, the effective range of each control parameter of algorithms has been determined.

4.2. The Design of Experiments Using the Taguchi Method. After determining the effective range of the control parameters, a set of design experiments (DOE) is designed using the Taguchi approach to find the effects of these parameters on the performance of the proposed algorithm and achieving their optimized mode.

In this study, the number of Pareto solutions (NPS), diversification metric (DM), mean ideal distance (MID), spread of nondominance solutions (SNS), and time (Time) are used to compare the performance of the suggested algorithms. The necessary experiments for analyzing the various modes of parameters and relative answers are presented in Table 2 of each algorithm. The data were analyzed using MINITAB20, and the results are shown in Figures 5 and 6 and Table 3.

Subsequently, the same procedure is repeated for MOEA/D. Table 4 indicates the test level of each parameter, and Figures 7 and 8 and Table 5 shows the results.

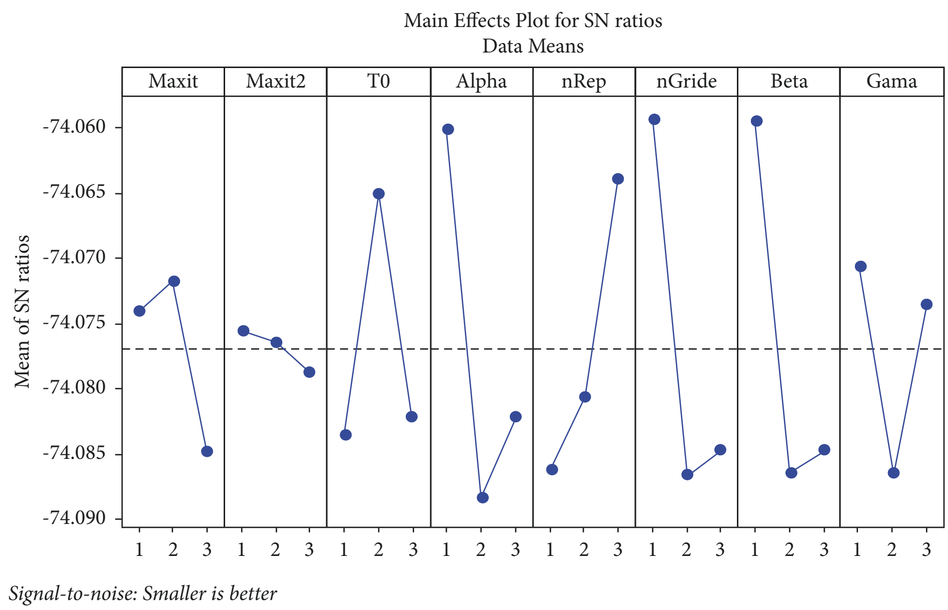

FIGURE 6: The answer to S/N propositions for MOSA algorithm.

TABLE 3: The best parameters for MOSA algorithm.

Algorithm parameters		Amounts
Maximum of iteration	MaxIt1	100
Maximum of internal iteration	MaxIt2	40
Primary temperature	T0	20
The coefficient of heating	Alpha	0.89
The maximum capacity of answer archive	nRep	20
Number of grids in each dimension	nGrid	8
The leading choice coefficient	Beta	1
The choice coefficient for deleting archive	Gamma	2

TABLE 4: The experiment levels of parameters for MOEA/D.

Algorithm parameters		First level	Second level	Third level
Maximum of iteration	MaxIt	90	100	120
Number of populations	nPop	40	60	80
Maximum number of archive	nArchive	10	20	30
The coefficient of creating harmony in the intersection operator	Gamma	0.5	0.6	0.7

4.3. Evaluating the Performance of Multiobjective Algorithms. To measure the quality and diversity of multiobjective metaheuristic algorithms, there are various comparison metrics. In multiobjective optimization, evaluating the algorithm performances could be difficult since there are differences presented in the final answers and paradoxes in objectives. When the visual analysis of results is very difficult, it is vital to have various metrics to evaluate the performances to recognize the best set of nondominance solutions.

To compare the capability of various algorithms in solving multiobjective problems, various methods and instruments are used. Some of them, which were used more in the multiobjective literature, will be discussed in the following session;

Number of Pareto solutions (NPS): this metric calculates the number of nondominance solutions achieved in every run of algorithm. According to it, the more the number of nondominance solutions, the better the algorithm performance.

Diversification metric (DM): this metric is about the range of Pareto solutions of algorithms. It can be calculated by the equation (23). The more the DM, the better the algorithm's performance.

$$\mathrm{DM} = \sqrt{\left(\frac{\max f_{1i} - \min f_{1i}}{f_{1\cdot\mathrm{total}}^{\max} - f_{1\cdot\mathrm{total}}^{\min}}\right)^2 + \left(\frac{\max f_{2i} - \min f_{2i}}{f_{2\cdot\mathrm{total}}^{\max} - f_{2\cdot\mathrm{total}}^{\min}}\right)^2}. \tag{23}$$

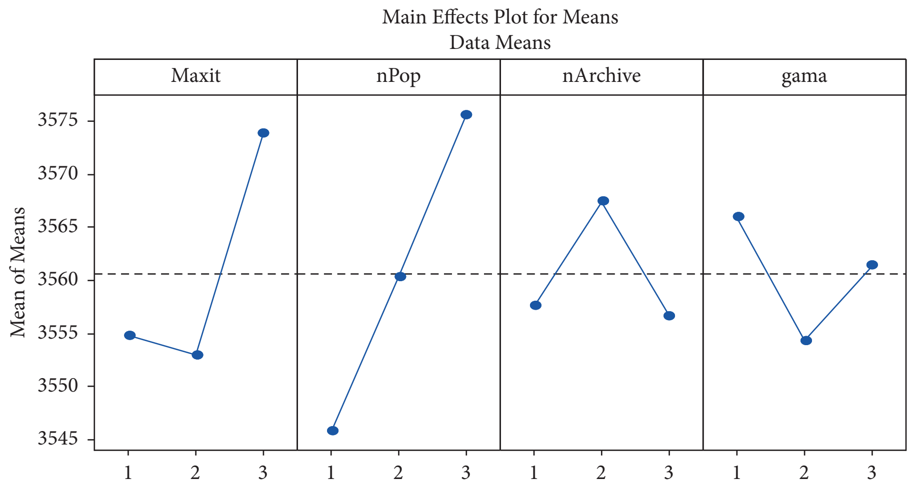

Figure 7: The answers to means for MOEA/D.

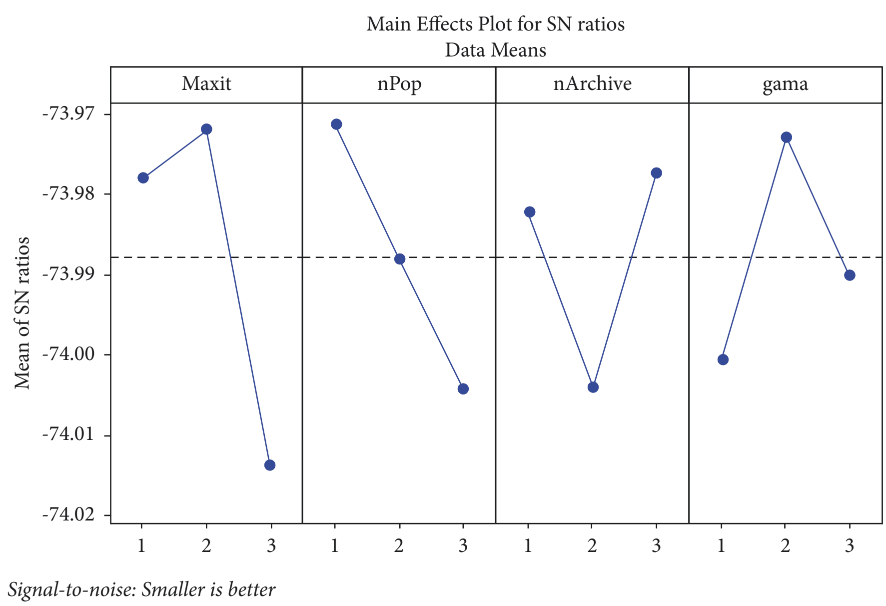

Figure 8: The answer of propositions of S/N for MOEA/D algorithm.

Table 5: The best parameters for MOEA/D algorithm.

Algorithm parameters		Amounts
Maximum of iteration	MaxIt	100
Number of populations	nPop	40
Maximum number of Archive	nArchive	30
The coefficient of creating harmony in the intersection operator	Gamma	0.6

Mean ideal distance (MID): the closeness of the achieved Pareto set is calculated by the algorithm to the optimized Pareto edge. Since achieving the Pareto edge is impossible for most problems, the distance between Pareto points and the ideal ones, (0,0), is calculated via this metric. The equation is as follows:

$$\mathrm{MID} = \frac{\sum_{i=1}^{n} c_i}{n}, \qquad c_i = \sqrt{f_{1i}^2 + f_{2i}^2 + \cdots}. \tag{24}$$

Table 6: The randomly made problems.

No.	Sets							
	$\|S\|$	$\|J\|$	$\|M\|$	$\|R\|$	$\|T\|$	$\|O\|$	$\|K\|$	$\|W\|$
1	2	3	3	2	2	3	2	3
2	2	4	3	3	3	4	3	3
3	4	5	3	3	3	5	4	4
4	5	6	4	4	4	6	4	4
5	6	7	4	5	5	7	4	4
6	7	8	4	6	5	8	5	5
7	8	9	5	6	6	9	5	6
8	8	12	5	7	7	12	6	7
9	9	15	6	7	8	15	6	8
10	10	18	6	8	12	18	6	9

Table 7: The results of the first example.

Solver	F1	F2	ci	MID	DM	SNS	NPS	CPU time
GAMS	3.9	6846.397	6846.398	6833.279	196.2954	69.49664	3	8.319
	3.265	6931.123	6931.124					
	18.375	6735.41	6735.435					
MOSA	16.79092	7107.18	7107.038	7125.3481	57.354102	10.74361	6	21.438
	15.50293	7122.977	7122.994					
	15.01459	7153.992	7154.008					
	15.90668	7097.138	7097.155					
	15.7337	7116.395	7116.413					
	15.73039	7154.464	7154.481					
MOEA/D	16.0243	6793.523	6793.535	6989.0047	301.463869	47.93552	3	20.862
	15.95949	7094.987	7095.005					
	16.15406	7078.449	7078.468					

In the above-mentioned equation, n is the number of achieved nondominance solutions, and $i = 1,2, \ldots, n$

Regarding MID, the more it is, the farther the algorithm is from the ideal Pareto edge.

Spread of nondominance solution (SNS): the spread among the solutions in the nondominance solutions achieved by the algorithm is calculated by this. The equation is follows:

$$\text{SNS} = \sqrt{\frac{\sum_{i=1}^{n} (\text{MID} - c_i)^2}{n-1}}. \quad (25)$$

The more SNS in the algorithm, the more the algorithm spreads, which is more favorable.

Computational time (time): the time for the performance of an algorithm. The lesser time the algorithm takes, the better.

In the following sessions, examples are categorized as small-, medium-, and large-scale ones regarding the model nature. For this purpose, two small scale examples, four mid-dimensional, and 4 high-dimensional ones were randomly made to evaluate the model when its design is real and parameters are random. These problems are listed in Table 6.

The examples one and two are solved using metaheuristic algorithms, and their results will be compared with the ones achieved using the GAMS exact program to validate the answers. The purpose is to confirm whether to continue with metaheuristic algorithms. Also, the Pareto front chart of each problem is presented, and the five metrics (MID, DM, SNS, NPS, and CPU time) are considered in it. The results are shown in Tables 7 and 8 and Figures 9–14. F_1 is the weight objective function of the completion time, and F_2 is the objective one of the total specification cost and waiting time for jobs.

4.4. The Comparison of Algorithm Results for Small-Scale Problems. Each algorithm was run for five times, and the best results were considered. The results of comparison between the proposed algorithms and one of the GAMS exact programs used for low-dimensional problems are mentioned in Table 9.

As mentioned earlier, the more DM, SNS, and NPs and the fewer MID and time, the more efficient the algorithm with respect to the metric. According to the results mentioned in Table 9 for problems 1 and 2, for time and SNS, the results of GAMS are better. In case of MID and DM, GAMS and MOEA/D have proved more fruitful. Regarding the NPS metric, the results achieved from GAMS and MOSA are relatively better. Nevertheless, since there is not much difference between results achieved from the GAMS and the ones of the proposed algorithms regarding any metrics, it can be concluded that they are well-designed and efficient, and thus, they are useable in solving problems with high dimensions.

Table 8: The results of the second example.

Solver	F1	F2	ci	MID	DM	SNS	NPS	CPU time
	96.687	17083.71	17083.99					
	96.954	16364.31	16364.6					
	93.629	16275.86	16276.13					
GAMS	11.918	14512.08	14512.08	16316.24	2975.562	510.3927	7	53.866
	101.701	17486.18	17486.48					
	105.125	16964.74	16965.07					
	95.016	16292.83	16293.1					
	83.29856	16175.07	16175.29					
	83.86166	15761.25	15761.47					
MOSA	83.60755	16997.18	16997.38	15846.64	1811.04972	297.0425	6	50.040
	87.97861	15654.26	15654.51					
	9331055	15304.49	15304.77					
	94.26282	15186.16	15186.46					
	85.18027	16962.28	16962.49					
	85.52582	15986.74	15986.97					
MOEA/D	85.66592	15900.14	15900.37	16337.45	1582.83487	305.2107	6	49.029
	90.99155	15899.48	15899.74					
	84.44851	17428.74	17428.95					
	92.54549	15845.93	15846.2					

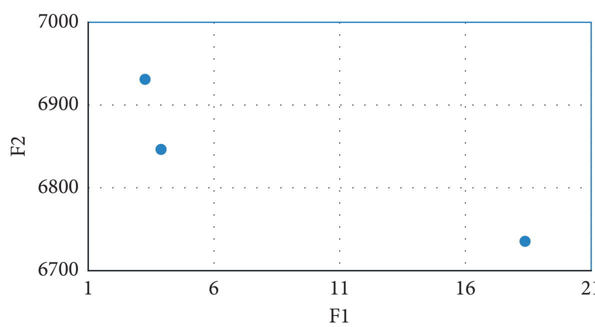

Figure 9: The Pareto front chart for the results of GAMS: the first example.

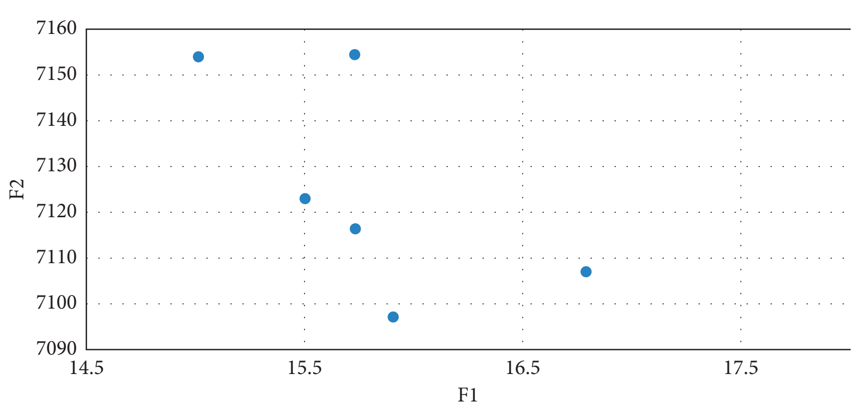

Figure 10: The Pareto front chart for the results of MOSA: the first example.

4.5. The Comparison between Algorithms for Mid- and High-Dimensional Problems. The results of the proposed algorithms used on mid- and high-dimensional problems after five times of running are shown in Table 10 and Figures 15–19. These results are based on the best answers achieved from algorithms.

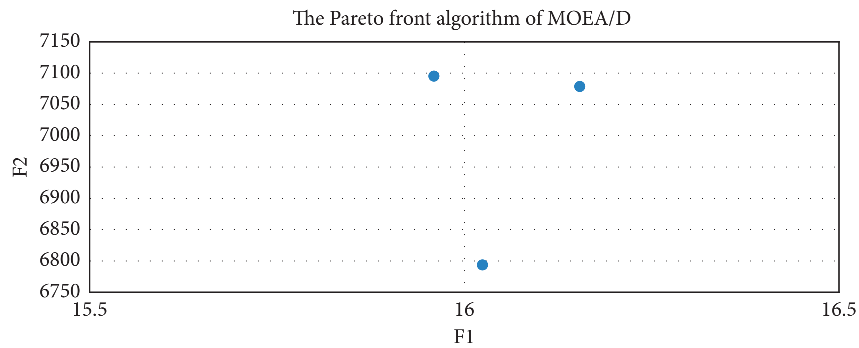

Figure 11: The Pareto front chart for the results of MOEA/D: the first example.

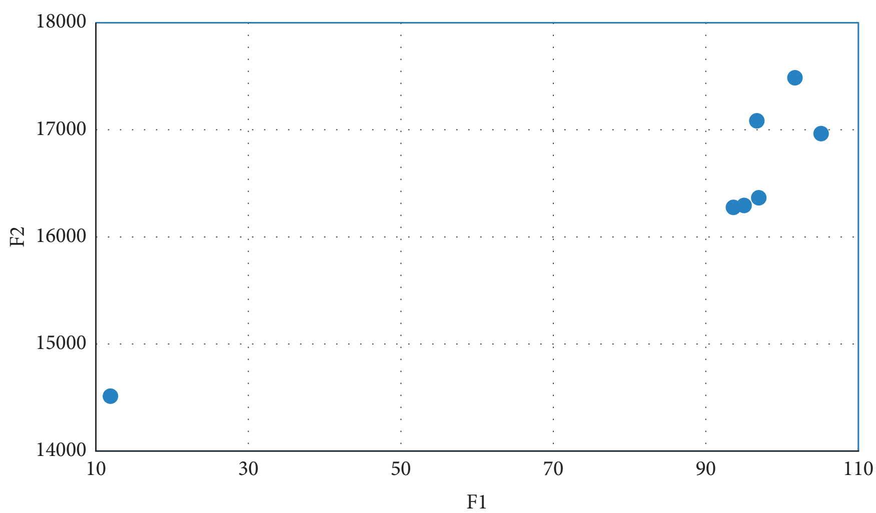

Figure 12: The Pareto front chart for the results of GAMS: the second example.

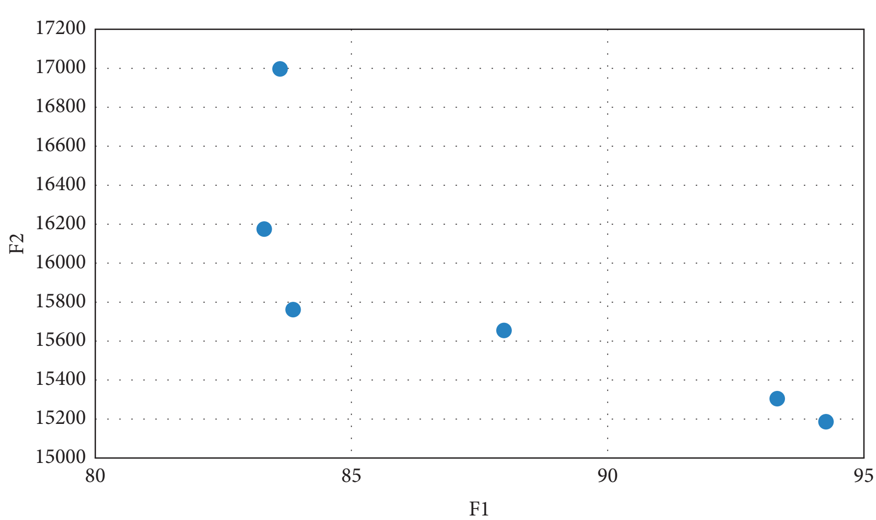

Figure 13: The Pareto front chart for the results of MOSA: the second example.

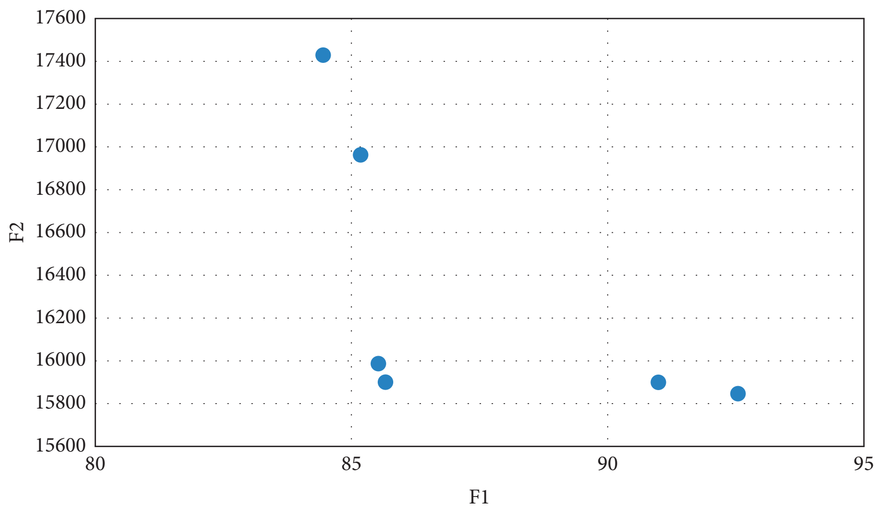

Figure 14: The Pareto front chart for the results of MOEA/D: the second example.

Table 9: The results of metrics for each algorithm run in small-scale problems.

No.	Time			MID			SNS			DM			NPS		
	MOSA	MOEAD	Gams	MOSA	MOEAD	Gams	MOSA	MOEAD	Gams	MOSA	MOEAD	Gams	MOSA	MOEAD	Gams
1	21.438	20.862	8.319	7125.348	6989.004	6833.279	10.74361	47.93552	69.49664	57.3541026	301.4638625	196.2954	6	3	3
2	50.040	49.02	53.86	16846.6	16337.454	16316.2	297.04	305.2107	510.3927	1811.04972	1582.834	2975.562	6	6	7

Table 10: The results of metrics for each algorithm used in mid- and high-dimensional problems.

No.	Time(s)		MID		SNS		DM		NPS	
	MOSA	MOEAD	MOSA	MOEAD	MOSA	MOEAD	MOSA	MOEAD	MOSA	MOEAD
3	173.2312	171.9171	45128	45066	542.2878	1915.129	4336.2	1314.3	4	4
4	490.708	425.816	150620	142280	661.9441	11370.84	21569	1612.2	3	4
5	954.5742	846.1652	264590	230780	1619.275	13641.11	33162	3232.5	4	3
6	1693.4007	1474.5585	349960	336930	3799.9	43734.55	83724	5378	3	2
7	3456.254	3283.7964	935947.19	930970	18821.77	33256.27	74419.3722	32944	6	3
8	9167.7022	9059.0237	2405000	2436500	58758.09	54630	124940	138560	5	4
9	46908.3333	45855.676	10119000	10179000	354790	443370.4	912610	502130	4	2
10	6554.5178	6402.0004	124420000	112720000	2109300	17022889	29934000	2986800	3	2

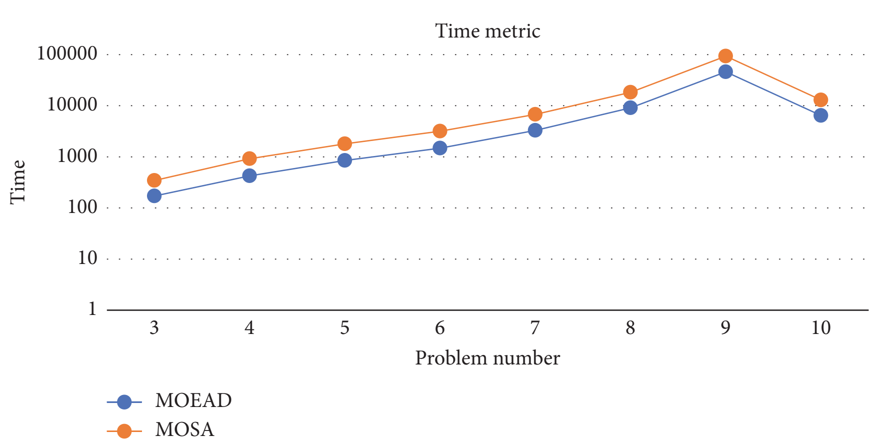

Figure 15: The comparison between the proposed algorithms done on mid- and high-dimensional problems based on time.

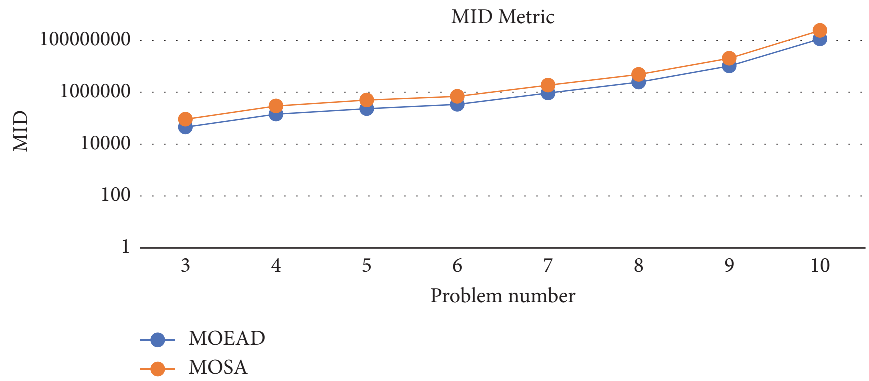

FIGURE 16: The comparison between the proposed algorithms done on mid- and high-dimensional problems based on MID.

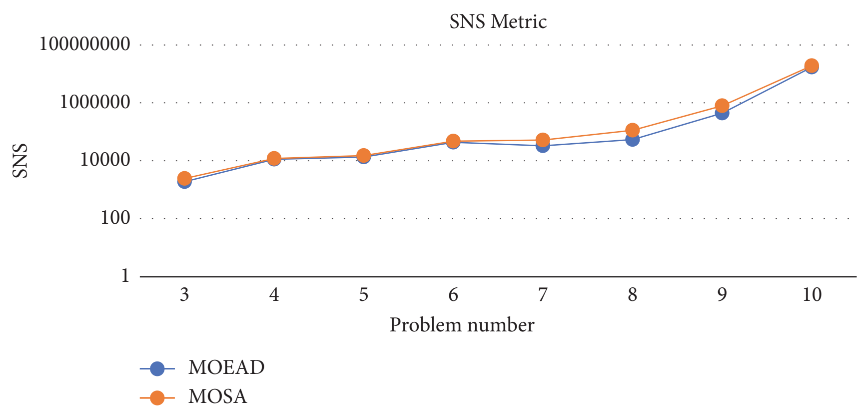

FIGURE 17: The comparison between the proposed algorithms done on mid- and high-dimensional problems based on SNS.

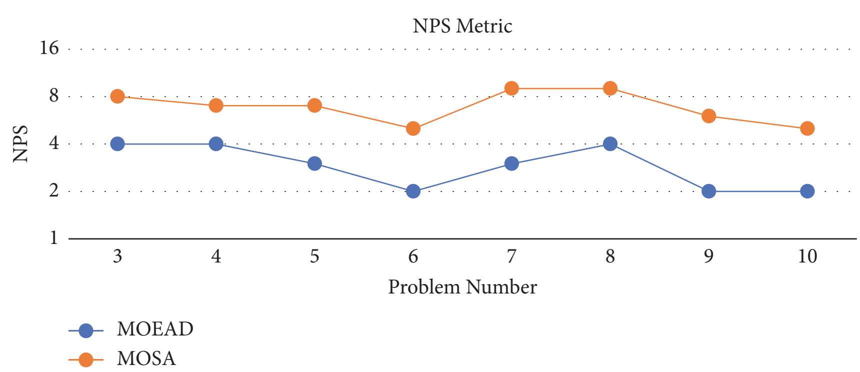

FIGURE 18: The comparison between the proposed algorithms done on mid- and high-dimensional problems based on NPS.

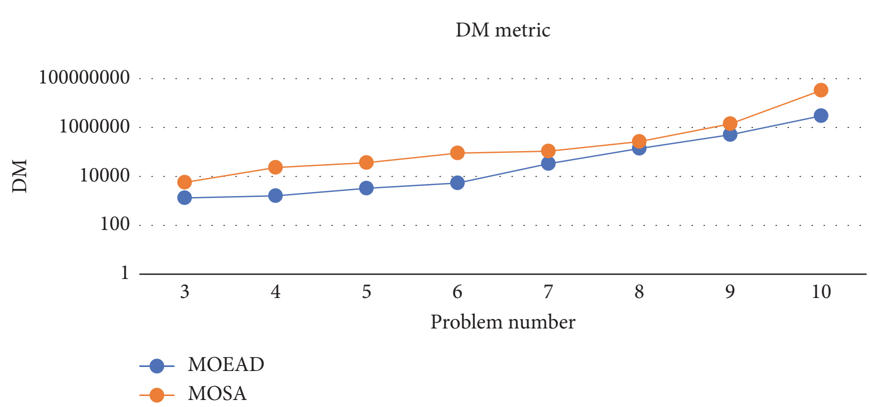

FIGURE 19: The comparison between the proposed algorithms done on mid- and high-dimensional problems based on DM.

According to Table 10 and Figures 15–19, MOEAD has a better performance regarding MID and time. On the other and, MOSA performs more fruitfully regarding SNS, DM, and NPS.

5. Conclusion

According to the studies on scenario-based flexible flow shop systems, a multiobjective problem has never been considered for the following characteristics: reliable transportation devices and constraint in capacity of intermediate buffers, intermediate limited buffers, dependent set-up time, and learning effect and budget constraints. Considering the reliability constraints of transportation devices and one of the finances has resulted the problem in becoming closer to the ones in the real world. The budget issues, like production costs, which are important for most companies and production factories, are also considered in this study. They can be considered in many of such problems in the real world. In the proposed model, scenarios are supposed with definite probabilities in each of which the cost of jobs and the amount of failure of transportation differ for each machine in different stages. In this study, it is supposed that in job stages, there are many transportation devices on machines for transporting and processing jobs and productions. The transportation of jobs and production in different stages has a specified time. The time of transportation of productions, which can be done by conveyors, lift-tracks, or automatic guided vehicles (AGV) is different. The reliability of transportation is also considered in this problem. It means that not all transportation devices are available all the time and preventive maintenance operations can be done on them. Also, because of the limited number of means of transportation between workstations, an intermediate limited buffer has been considered for each workstation. The number of jobs in each buffer must be less than the maximum capacity of them. Each time the jobs cannot be processed on machinery since machines being busy or under preventive maintenance operations, they remain in buffers. For each time unit that jobs wait to be processed in intermediate buffers, an amount of maintenance cost is considered for the project. The total budget for maintaining jobs in intermediate buffers is limited, and their total cost must be less than the total available budget. Each job is done depending on the previous one in each stage, and there is a preparation time for them. There is a learning rate for human resources regarding each machine, based on which processing and preparation time decreases. According to the above-mentioned issues, it can be concluded that parameters affecting the objective functions in the problem are as follows: processing time on productive machines, time of transportation using transportation devices, dependent preparation times, learning effect, waiting time of jobs in intermediate buffers, and reliability of transportation devices. As a result, the importance of discovering the best sequence of jobs and optimizing the objective functions becomes clear.

In this study, a new mathematical model with reliable transportation and constraints in intermediate buffers regarding budget and the effects of human resources learning was presented for solving the flexible flow shop problem. Firstly, the model was validated to confirm its accuracy of performance. Then, since it is an NP-hard one and the exact methods are not useful in logical calculation time, two metaheuristic algorithms, namely MOSA and MOEA/D, were presented to solve mid- and high-dimensional problems. To ensure the proposed algorithms accuracy of performance, two small-scale problems were solved by GAMS exact solution software, and the results obtained were compared with the output of the algorithms. Since the problem in this study is multiobjective, five comparative indices, nsmely the number of Pareto solutions (NPS), diversification metric (DM), mean ideal distance (MID), spread of nondominance solutions (SNS), and computational time (CPU Time), were used to compare the performance of algorithms. The results show that the answers achieved using the metaheuristic algorithms are very close to the ones achieved via the GAMS exact program. Therefore, the proposed algorithms were validated, and it was proven that they were accurately designed and useable in solving the real-world problems (which have mid- and high-dimensions) in logical computational time. Comparing the answers with the ones in the real-world, it was observed that regarding the time and MID indices, the MOEA/D algorithm, SNS, DM, and MPS indices, MOSA had relatively better performance. Considering the confirmation of the accuracy of the proposed algorithms previously done by comparing the answers with the ones of GAMS, it can be concluded that both MOSA and MOEA/D are useful in solving the mid- and high-dimensional modes of the problem in the study, which is very applicable in the real world.

5.1. Recommendations. The flexible flow shop problem can be developed considering other assumptions and parameters in machinery, in which, among them, the following are mentioned. The mathematical model of the present problem can be developed by considering assumptions, such as independent time for initial set up, failure, maintenance operation, process of operations considering the constraints in availability, preventive repairs on productive machinery regarding maintenance and repairing costs, the time needed for machine installation, process of operations with availability constraints to machinery, availability time the legal change in sequence, intervals for restarting, and so on. Also, another objective function can be considered for the quantification of delays or time consumption in intermediate buffers or total time of operations. In addition, in solving the problem, other multiobjective metaheuristic algorithms can be used, such as neural network, ant colony, tabu search, water flow-like, etc. Metaheuristic and heuristic algorithms, such as branch and bound, Lagrange or a hybrid of two or more algorithms, and a hybrid coding, can also be used to solve such problems.

Data Availability

Data will be available upon request to the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Allahverdi and J. Mittenthal, “Dual criteria scheduling on a two-machine flow shop subject to random breakdowns,” *International Transactions in Operational Research*, vol. 5, no. 4, pp. 317–324, 1998.

[2] M. Abolghasemian, A. P. Chobar, M. AliBakhshi, A. Fakhr, and S. Moradi, “Delay scheduling based on discrete-event simulation for construction projects,” *Iranian Journal of Operations Research*, vol. 12, no. 1, pp. 49–63, 2021, þ.

[3] M. Rahmaty, A. Daneshvar, F. Salahi, M. Ebrahimi, and A. P. Chobar, “Customer churn modeling via the grey wolf optimizer and ensemble neural networks,” *Discrete Dynamics in Nature and Society*, vol. 2022, pp. 1–12, Article ID 9390768, 2022.

[4] F. Maadanpour Safari, F. Etebari, and A. Pourghader Chobar, “Modelling and optimization of a tri-objective Transportation-Location-Routing Problem considering route reliability: using MOGWO, MOPSO, MOWCA and NSGA-II,” *Journal of optimization in industrial engineering*, vol. 14, no. 2, pp. 83–98, 2021.

[5] A. P. Chobar, M. A. Adibi, and A. Kazemi, “Multi-objective hub-spoke network design of perishable tourism products using combination machine learning and meta-heuristic algorithms,” *Environment, Development and Sustainability*, pp. 1–28, 2022.

[6] M. S. Salvador, “A solution to a special class of flow shop scheduling problem,” in *Proceedings of the Symposium of the Theory of Scheduling and Applications*, pp. 83–91, New York, NY, USA, 1973.

[7] J. N. D. Gupta, “Two-stage, hybrid flowshop scheduling problem,” *Journal of the Operational Research Society*, vol. 39, no. 4, pp. 359–364, 1988.

[8] S. M. Johnson, “Optimal two and three-stage production schedules with setup times included,” *Naval Research Logistics Quarterly*, vol. 1, pp. 61–68, 1954.

[9] R. Tavakkoli-Moghaddam, A. Rahimi-Vahed, and A. H. Mirzaei, “A hybrid multi-objective immune algorithm for a flow shop scheduling problem with Bi-objectives: weighted mean completion time and weighted mean tardiness,” *Information Sciences*, vol. 177, no. 22, pp. 5072–5090, 2007.

[10] B. Naderi, M. Zandieh, and V. Roshanaei, “Scheduling hybrid flow shops with sequence dependent setup times to minimize makespan and maximum tardiness,” *International Journal of Advanced Manufacturing Technology*, vol. 34, p. 986, 2010.

[11] K. Wang and S. H. Choi, “A holonic approach to flexible flow shop scheduling under stochastic processing times,” *Computers & Operations Research*, vol. 43, pp. 157–168, 2014.

[12] R. Shahidizadeh, A. Tavakkoli-Moghaddam, T Taheri-Moghaddam, and I. Rastegar, “Solving Bi-objective unrelated parallel batch processing machines scheduling problem: a comparison study,” *Computers & Operational Research*, vol. 875, 2017.

[13] C. J. Liao, W. C. Yu, and C. B. Joe, “Bicriterion scheduling in the two-machine flowshop,” *Journal of the Operational Research Society*, vol. 48, no. 9, pp. 929–935, 1997.

[14] M. Zandieh, S. Fatemi Ghomi, and S. Moattar Husseini, “An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times,” *Applied Mathematics and Computation*, vol. 180, no. 1, pp. 111–127, 2006.

[15] A. Mollaei, M. Mohammadi, and B. Naderi, “A bi-objective MILP model for blocking hybrid flexible flow shop scheduling problem: robust possibilistic programming approach,” *International Journal of Management Science and Engineering Management*, vol. 14, no. 2, pp. 137–146, 2019.

[16] F. Ozsoydan and M. Sagir, “Iterated greedy algorithms enhanced by hyper-heuristic based learning for hybrid flexible flowshop scheduling problem with sequence dependent setup times: a case study at a manufacturing plant,” *Computers & Operations Research*, vol. 125, 2020.

[17] C.-Y. Cheng, P. Pourhejazy, K.-C. Ying, and Y.-H. Liao, “New benchmark algorithms for No-wait flowshop group scheduling problem with sequence-dependent setup times,” *Applied Soft Computing*, vol. 111, Article ID 107705, 2021.

[18] K. Shen, T. De Pessemier, L. Martens, and W. Joseph, “A parallel genetic algorithm for multi-objective flexible flowshop scheduling in pasta manufacturing,” *Computers & Industrial Engineering*, vol. 161, Article ID 107659, 2021.

[19] F. Pagnozzi and T. Stützle, “Automatic design of hybrid stochastic local search algorithms for permutation flowshop problems with additional constraints,” *Operations Research Perspectives*, vol. 8, Article ID 100180, 2021.

[20] R. Lotfi, N. Mardani, and G. W. Weber, “Robust bi-level programming for renewable energy location,” *International Journal of Energy Research*, vol. 45, no. 5, pp. 7521–7534, 2021.

[21] S. S. Zabihzadeh and J. Rezaeian, “Two metaheuristic algorithms for flexible flow shop scheduling problem with robotic transportation and release time,” *Applied Soft Computing*, vol. 40, pp. 319–330, 2016.