

Applying an Intelligent Dynamic Genetic Algorithm for Solving a Multi-Objective Flexible Job Shop Scheduling Problem with Maintenance Considerations

Abbasian M¹, Nosratabadi HE² and Fazlollahtabar H^{3*}

¹Department of Industrial Engineering, Tarbiat Modares University, Tehran, Iran

²Department of Information Technology Management, Science and Research Branch, Islamic Azad University, Tehran, Iran

³Faculty of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract

In this paper, a multi-objective flexible dynamic job shop scheduling problem (MO-FDJSPM) with maintenance constraint is studied. The objectives of the scheduling are maximizing the completion time, mean job rotation time and mean components' tardiness. Also, in order to adapt with the internal disruptions of the manufacturing system, such as breakdown of existing machines, we consider the machines availability (so called maintenance) as a constraint. The multi-objective mathematical model is formulated and a genetic algorithm (GA) with dynamic bi-dimensional chromosomes along with a heuristic algorithm to handle maintenance sub-problem is developed as solution approach. In proposed algorithm, since the control parameters change intelligently and dynamically during implementation and optimization process, the early convergence and trapping in local optimum are reduced leading to performance improvement. The performance of the proposed approach is evaluated with respect to convergence speed and solutions quality. The results of computations verify and confirm both two evaluation criteria.

Keywords: Flexible dynamic jobshop; Multi-objective scheduling; Maintenance; Genetic algorithm (GA)

Introduction

Many researchers have worked out on multi-objective scheduling problem in manufacturing systems. The objectives were mostly concentrated on resource allocation, optimal job rotation and accurate delivery. Generally, machines' availability includes two statuses namely, random (such as damage or breakdown of machines) and definite (such as preemptive maintenance, fundamental repairs and pre-scheduling) [1]. MO-FDJSPM problem considers maintenance constraints being an optimization problem in discrete space. Due to being non-convex and non-linear, solving this problem leads to several local optimal [2]. Reviewing the literature points out that for solving scheduling problem meta-heuristic approaches are being extensively applied. Among those methods, GA has been found to be better in performance than other methods. Studies indicated that one of the disadvantageous of classic GA is parameter convergence property [3]. GA converges after some replications towards obtaining optimal solution and/or near-solution. But there are relatively noticeable variety in how utilizing parameters, type of parameters, choose mechanism, how create initial population, how select parents and etc., leading different configurations of GA in various works.

The objective of this paper is modeling and proposing an effective solution approach for handling a multi-objective flexible dynamic job shop scheduling problem considering maintenance constraint. The problem includes dynamism in production because of jobs arrival in a non-zero time as well as flexible of operation and flexibility resulted from parallel machines and also being multi-objective where maintenance constraint is noted. Given this fact that MO-FDJSPM considering maintenance constraint always trap in several local optimal and the GA's premature convergence, in our work, a genetic algorithm is developed having control parameters change dynamically during algorithm's implementation and optimization process. This way, the premature convergence and trapping into local optimal points are reduced.

The remainder of our work is organized as follows. In Section 2, the related literature is reviewed. The proposed problem is defined and formulated in Section 3. The solution approach is comprehensively developed and explained in Section 4. A numerical example is fully illustrated in Section 5 to point out all aspects of the proposed model and the solution methodology associated with validity testing. The experimental results are discussed and justified in Section 6. We conclude in Section 7.

Literature Review

Machines' availability

Generally, the constraint of machines' availability in production systems divided into two classifications called definite and indefinite availability. In indefinite availability, the lack of availability period is considered to be flexible in being determined during scheduling process. Of course, maintenance activity should be worked out in certain time windows [4]. Scheduling problem given constraint of availability to machines was introduced by Schmidt in 1984 for the first time. In 1989, Adiri and et al. [5] showed that scheduling problem of single-machine is NP-hard problem. Then Schmidt [6], considering availability constraint to machines, developed single-machine, parallel machines and flow shop scheduling problems.

***Corresponding author:** Fazlollahtabar H, Faculty of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran, Tel:+98 21 77240540-50; E-mail: hfazl@iust.ac.ir

Received March 16, 2015; **Accepted** June 10, 2015; **Published** June 17, 2015

Citation: Abbasian M, Nosratabadi HE, Fazlollahtabar H (2015) Applying an Intelligent Dynamic Genetic Algorithm for Solving a Multi-Objective Flexible Job Shop Scheduling Problem with Maintenance Considerations. J Appl Computat Math 4: 227. doi:10.4172/2168-9679.1000227

Copyright: © 2015 Abbasian M, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In 2002, Xie and Wang [7] presented scheduling problem of two stages flexible flow shop considering parallel machines and considering availability constraint to machines and production regime of R type. Cheng and Liu [8] in 2003 presented scheduling problem of flow shop given no-wait job and constraint on each machines. In 2005, Kubzin and Strusevich presented scheduling problem of two-machine flow shop given no-wait job and apply constraint on each machine. Xie and Wang [9] presented scheduling problem of two stage flow shop with parallel machines given availability constraint on machines. In 2006, Lee and Wu [10] expressed scheduling problem of single-machine given the effect of learning and availability constraint on machines.

Zribi and Kamel in 2007 presented scheduling problem of jobshop with multi-machines and the presence of availability constraint on machines. They were expressed a little researches about scheduling problem with indefinite availability constraint. In 1999, Kayo et al. indicated that single-machine problem considers several returns of inaccessibility is NP-hard. At the same year, single-machine problem with indefinite inaccessibility period by Keravas and Lee were studied. After them, Cheng and Lee in 2000 studied scheduling problem of parallel machines. In 2004, Agoni combined scheduling problem of production and maintenance in flow shop environment. In 2006 Breit [11] expressed scheduling problem of single-machine given availability constraint on machines. At the same year, Goo and et al. [4] presented scheduling problem of jobshop with possibility of an inaccessibility period for each machine. In 2010, Zegordi and Rahimi [12] investigated scheduling single-objective given availability constraint on machines and production regime of N-R type and also they utilized a classic GA in order to solve it. Next, we review the works made use of GA to solve scheduling problems.

Genetic algorithm in scheduling problems

Many of the optimization problems in production systems, in particular, optimization problems of scheduling in shops were so complex and usual optimization methods are not able to find their solutions. Scheduling of flexible dynamic job-shop is one of the most difficult problems in combinatorial optimization topics. Generally, studies about scheduling in dynamic environment are divided into two main groups. The first group is queue-based and the second group is rolling time horizon technique [13,14]. Regarding to the complexity of the problem, the above analytical methods are often employed to solve single machine problems. But, for problems with more machines, meta-heuristic methods are used. In the literature, approaches such as simulated annealing, particle swarm optimization, artificial intelligence, evolutionary algorithms, fuzzy theory and etc. were applied for solving scheduling problems. Among those methods, GA has better performance than other methods. Some researchers think that GA is an efficient method to solve scheduling problem as an optimization problem [4,15].

In 1997, Brandimarte [16] solved the problem of flexible process program and proposed two methods for solving namely, Dual-based and GA based. At the same year Ghedjati [17] presented a synthetic approach from Meta heuristic methods, GA-based for solving FJS problem. Concerning the strategy with constraints and non-search chromosome, he also advised it was much better to use method-based rejection non-search chromosome for solving the problem that have non convex solution space. Kacem et al. [18] in 2002 studied FJS problem in multi- objective case for the first time. Their presented method was compounding of fuzzy logic and evolutionary algorithms using fuzzy logic for access to pareto-optimal solutions. At the same year, Lee et al. [19] presented a GA for solving similar problem with

FJS in supply chain. In 2004, Tay and Wibawo [2] utilized a certain representation for their GA and solved scheduling problem of flexible job-shop. They introduced such scheduling problem to be an NP-hard one. At the same year, Cheng and Gen [20] presented a GA for solving studied problem by Kacem and et al. In this research, first sub problem of allocation solved by prioritizing rule SPT, then selecting one appropriate representation of chromosome a good quality solution was obtained. The results of calculations indicated the effectiveness of their algorithm for large sized problems. Kurz and Askin [21,22] presented RKGGA Meta heuristic algorithm for solving FSPM problem. They employed hierarchical approach for solving their problem and analyzed them for two sub problems of allocation and assignment sequence of operation. In 2007 Tay and HO [15] utilized Genetic programming for solving their flexible job shop problem. Ho et al. [23] developed a methodology for solving FJSP problem supposed to secondary rotation of jobs introduced inordinate counting on evolutionary algorithms. Secondary synthetic mechanisms were most constraints of this method and random selection. In other to overcome these constraints and obtain effective solutions for FJSP problem, they presented their proposed GA called LEGA. They considered combination of these patterns and production of population simultaneously. In that approach a simple distribution rule of module learning has been used. Koliner [24] employed Genetic agents that using agents of two or more section points having priority to other agents. Dagli and Sittisathachai [25] introduced premature convergence and fell them in to local optimal points trap as a one of the mast defects in classic GA. Gao et al. [4] at the same year used combination of GA with innovative method of transfer of neck bottle for solving FJSP problem. In 2009, Nahavandi and Abbasian [26,27] presented simple GA with two dimensional chromosomes for solving FDJSPM problem and they also showed the priority of their method towards a similar method in literature. At this same year, Amiri et al. [28] used a compound plan for simulation of chromosome behavior. Moreno et al. [29] proposed binary representation of GEP chromosomes for search solutions. They indicated that this kind of representation improved GEP scaling considerably. In 2010, Nahavandi and Abbasian [30] presented GA with two dynamic dimensional chromosomes for solving their problem. Verma et al. [31] proposed a method so called calculation data- intensive technique and showed that this technique has basic role in scaling and estimating GA distribution. Generally, with exhaustive study that carried out about background research in Field of "availability constraint machine" and "methods of presented solution for flexible job shop problem, in particular using the majority of scheduling models of problems" become specified that first in the majority of scheduling models of flexible job shop suppose that machines are available all the time. Although in actual environments of production and manufacture it is possible machine in several reasons are not available over planning horizon. Second, such scheduling problems because of their non-convex nature and non-linear usually have defects of classic GA is premature convergence and fell them in to local optimal points trap [3]. Therefore, with identifying present locks objective of the paper considered modeling and solving scheduling problem of flexible dynamic multi- objective job shop considering Maintenance constraint. In the proposed GA for solving the problem as well as to overcome constraint of premature convergence and fell in to local optimal points trap two strategies were used. First strategy attempted to increase the capability of the searching algorithm. The next strategy makes the optimization process intelligent using the Genetic agents. So, in the proposed GA the possibility of each agent in the optimization process are determined dynamically and according

to the number of elite chromosomes that transfer to next generation population directly.

Proposed Problem and Mathematical Modeling

In this section, MO-FDJSPM problem is defined and then the indices, parameters, variables and the mixed integer mathematical model are presented.

Definition of problem

Many researchers of scheduling science considered nature of scheduling science in production and manufacture environments to be multi-objective in many cases and defined basic objectives of scheduling as follows [32]:

- 1) Optimal use of resources and existing machines in shop
- 2) Minimizing supply in operation and accelerate to flow job
- 3) High commitment against customers and considered delivery dates

To obtain these objectives it is necessary to adapt several performance criteria to the concepts of scheduling and control them during problem solving, simultaneously. For example, along with considering performance criterion "Maximum time of components completion" (for optimal use of resources and machines availability) to reach high commitment against customers and also assignment delivery dates and/or in other words just in time (JIT) components, control performance criterion adaptable to philosophy of JIT such as "an average time of components delivery delay" can be included to this objective. But it should be noted that minimizing performance criterion for components that complete earlier than delivery date is significant. Therefore, it means that it is necessary here along with considering performance criterion also control a reflective criterion of these costs as well as performance criterion of an average flow time components. These objectives adapted to philosophy of on time production and managerial objectives of supply chain and their simultaneous controlling ended up to improvement of system performance [33].

MO-FDJSPM problem considering Maintenance constraint defined as follows: There are in a number of n component, m job station, and in addition to completion each component is necessary to specific operation set (like O); the order of movement each component between job stations are different from several components. In present research for as much as possible problem to actual environment of production and manufacture, shop environment are considered dynamic. So, each of the components arrives to shop at a time r (a non-zero and random time). In one way, to avoid making bottleneck and increasing production rate in addition to each of job stations have process ability more than one component type (operation flexibility) each job station M , compounding of parallel machine set that are able to process allocated job to its job station (flexibility resulted from existing parallel machines). There are operation flexibility, machines and sequence of several operations. But the operation that should be done are definite and stable [3]. In this research, viability constraint to machines considered definite type and so called Maintenance constraint. Maintenance constraint investigates in indefinite availability case and unavailable period are flexible at this condition. Of course, each Maintenance activity should do in predetermined time window and its breakup is not permissible. Characteristics of jobs considering this constraint are investigated in permissible breakup operation case and process regime of starting job again. More over basis hypothesis, there are cases in the problem that, the breakdown of operation is permissible, there is not any possibility of coherent

availability to machines, Maintenance limitation (constraint) is definite and indefinite types; process regime is starting job again, break down of Maintenance activity is not permissible. Flexible job-shop scheduling problem considering Maintenance constraint divided in to three sub problems as follows: routing, assignment to single machine, sub problem of sequence assignment, sequence of allocated operation to machines, and specify sub problem of Maintenance scheduling each of Maintenance activities. It is supposed that decision maker has adequate information about weight of objectives or their priority. The objective of solving problem found best weighted sum of objectives of problem and presented them to decision maker.

Mathematical model of the problem

In the present work, a nonlinear program, mix integer mathematical model is presented for the proposed problem. In this model, there is n job (component) in number of L processing stage (job station) that doing each job in dynamic shop is necessary to specify operation set. Job i^{th} has O_i operation with definite sequence. Each component such as component i^{th} for process in shop arrives to dynamic shop in a non-zero r_i time. Job station of L^{th} has same parallel machine m_l . Operation o_{ik} by machine m_{lj} processed from available machine set M_{ikl} and in job station predetermined w_l for P_{iklj} time unit. Machine m_{lj} has maintenance activity R_{lj} that it should be done during programming horizon. Also, Maintenance activity P_{mijr} has been done for t_{ijr} time unit and completed specified time window. In continuation, parameters and decision variables are introduced and then mathematical model of the problem is presented.

Parameters: o_i : the number of job operation i^{th}

o_{ik} : operation k^{th} from job i^{th}

r_i : arrival time of component i^{th} to shop

w_l : job (work) station l^{th}

m_l : the number of parallel machines in work station l^{th}

m_{lj} : machine j^{th} from work station l^{th}

P_{iklj} : time period of processing o_{ik} on m_{lj}

M_{ikl} : set of available machines for operation processing o_{ik} in work station l^{th} .

t_{mij} : available time m_{ij}

R_{lj} : the number of Maintenance activity on m_{lj}

PM_{ijr} : Maintenance activity r^{th} on m_{ij}

t_{ijr} : time period of doing PM_{ijr}

U_{ijr}^E : Earliest time of completion PM_{ijr}

U_{ijr}^L : Latest time of completion PM_{ijr}

$[U_{ijr}^E, U_{ijr}^L]$: Time window for completion PM_{ijr}

Decision variables: C_{ik} : completion time o_{ik}

U_{ijr} : completion time PM_{ijr}

$$V_{ik1jr} = \begin{cases} 1 & \text{if } o_{ik} \text{ do before } PM_{1jr} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{ik1j} = \begin{cases} 1 & \text{if } m_{1j} \text{ choose for doing } o_{ik} \\ 0 & \text{otherwise} \end{cases}$$

$$Y_{ikhq} = \begin{cases} 1 & \text{if } o_{ik} \text{ do before } o_{hq} \\ 0 & \text{otherwise} \end{cases}$$

Research problem formulate as follows: Min

$$F = \alpha_1 F_1 + \alpha_2 F_2 + \alpha_3 F_3 \text{ (Minimization Weighted Sum of Three Objectives)} \quad (1)$$

$$F_1 = C_{\max} = \text{Max}\{C_i \mid i=1, \dots, N\} \text{ (Minimization of make span)} \quad (2)$$

$$F_2 = \bar{F} = \frac{1}{N} \sum_{i=1}^N \max\{C_i - r_i\} \text{ (Minimization of Mean flow time)} \quad (3)$$

$$F_3 = \bar{T} = \frac{1}{N} \sum_{i=1}^N \max\{\beta_i(C_i - d_i), 0\} \text{ (Minimization of Mean Tardiness)} \quad (4)$$

S.t

$$(c_{ik} - c_{i,k-1}) \geq p_{iklj} \cdot x_{iklj} \quad k=2,3,\dots,o_i; \forall i,l,j \quad (5)$$

$$(c_{ik} - c_{hq} - p_{iklj}) \cdot x_{iklj} \cdot x_{hqij} \cdot (1 - y_{ikhq}) \geq 0 \quad \forall \{i,h \mid i \neq h\}, k,q,l,j \quad (6)$$

$$(c_{hq} - c_{ik} - p_{hqij}) \cdot x_{hqij} \cdot x_{iklj} \cdot y_{ikhq} \geq 0 \quad \forall \{i,h \mid i \neq h\}, k,q,l,j \quad (7)$$

$$(c_{ik} - u_{ljr} - p_{iklj} - r_i) \cdot x_{iklj} \cdot (1 - v_{ikljr}) \geq 0 \quad \forall i,k,l,j,r \quad (8)$$

$$(u_{ljr} - c_{ik} - t_{ljr} - r_i) \cdot x_{iklj} \cdot v_{ikljr} \geq 0 \quad \forall i,k,l,j,r \quad (9)$$

$$\sum_{l=1}^L \sum_{j \in M_{il}} x_{iklj} = 1 \quad \forall i,k \quad (10)$$

$$U_{ljr}^E \leq u_{ljr} \leq U_{ljr}^L \quad \forall l,j,r \quad (11)$$

$$c_{ik} \geq 0 \quad \forall i,k \quad (12)$$

$$u_{ljr} \geq 0 \quad \forall l,j,r \quad (13)$$

$$x_{iklj} \cdot y_{ikhq} \cdot v_{ikljr} \in \{0,1\} \quad \forall \{i,h \mid i \neq h\}, k,q,l,j,r \quad (14)$$

Equation (1), main objective Function of Mo-FOJSPM problem indicates minimizing weighted sum of three objectives resulting from relations (2) to (4) with definite coefficients $\alpha_1, \alpha_2, \alpha_3$. The desirable delay unit in each component delivery for whole jobs is 1 ($\beta_j = 1$). It is necessary to note that in actual production and manufacturing systems, the number of weighted coefficients should be determined by proficient of that industry where it is possible these weights be different from a manufacturing environment to another. In the present work suppose that all these objectives have the same priority in shop level. So, we will consider the number of $\alpha_1, \alpha_2, \alpha_3$ similar and equal to hypothetical number 1/3. Because each scheduling program becomes possible solutions of logic constraints all done operation from each definite job should be allocated resources with ordinal priority and without interference to each other. Equations (5) guarantee that sequence of jobs operation set doesn't have time interference, in other words the set of these in equations make each operation in a sequence begin if their pioneer operation were applied.

As we can't process two operations on a single machine simultaneously, each valid scheduling should be also a possible solution for constraints. Set of constraints (6) and (7) guarantee simultaneously that set of operation that employed on single machine, don't have time interference. All presented models for jobshop problem have at least two similar constraints that in case of violation of these two constraints problem exit from jobshop case. So, all shop problems contains as presented in equation 5, 6 and 7 in the proposed mathematical model.

In equation (8) and (9) demonstrate the coincident constraints doing Maintenance activities and several jobs operation on single machine. As well as set of these in equations of dynamism problem because of differential arrival time shop are considered. Equation (10) indicated that one should choose a machine from available machines set for doing each operation. Equation (11) explained that Maintenance activities should be completed in a specific time window.

In order to measure validity of the proposed mathematical model for Mo-FOJSPM problem, a solution approach for small size problems is implemented and tested by Lingo software. And their results also confirmed validity of the proposed model for Mo-FOJSPM problem considering Maintenance constraint.

Being NP- hard problem

Mo-FOJSPM problem is developed of the flexible production shop model (FOJSP14) considering flexibility and as a result will also bring all complexities and difficulties. Considering any flexibility of operation in these problems, even complexity of estimated optimal solution also increased remarkably [1]. As FOJSP with flexibility of operation is strongly NP- hard itself [34], but Mo-FOJSPM problem also considering flexibility resulted from parallel machines in dynamic production and manufacturing environment regarding Maintenance constraint will be NP- hard strongly.

Problem Solution Approach

For solving this kind of scheduling problem, it is required to employ an effective optimization strategy for considering complexities resulted from solution space for obtaining appropriate values as for objective functions. According to the literature, several approached can be used for solving flexible job shop problem. One is to divide the main problem to three sub problems of routing, assignment of sequence and Maintenance scheduling. In the present work a synthetic method based on GA is proposed for solving the problem. In this way that two sub problems are routing and assignment of sequence solved by GA and for solving sub problem of Maintenance scheduling an innovative algorithm is utilized. It is supposed that decision maker has adequate information about weight of objectives and their priority. The objective of Mo-FOJSPM problem is finding the best weighted sum of objectives.

Next, we present the GA structures including the presentation of chromosomes, tuning the parameters, and dynamic adjustment of parameters to solve the problem. Also, the innovative algorithm for solving Maintenance sub problem and the efficiency of the proposed synthetic method according to weighted sum of objective function is discussed through numerical experiments.

Proposed genetic algorithm

As Mo-FOJSPM is an NP- hard problem, an algorithm that can obtain the solutions in a logical time for large sized problems is of importance. So, in this section for solving the proposed problem, a GA will be presented so that to obtain the solutions in a logical time. The general procedure of the proposed GA is shown in Figure 1.

Presentation of chromosomes: The first step for designing a GA is representation of the problem as chromosomes [34]. In presenting chromosomes, usually parameters needing less space and time are considered since otherwise due to complexity infeasible chromosomes may be generated [34]. Mo-FOJSPM problem considering Maintenance constraint is a tradeoff between two sub problems of allocation and assignment of operation sequence. The proposed GA is designed in such a way to be able to solve both two above mentioned sub problems,

1. **Begin**
2. Counting index of population (t) is zero;
3. Design first dynamic two dimensional chromosome
4. **While** stop condition don't have, continue:
5. Produce offspring C(t) with implementation next compound from father P[t]
6. Calculate fitness degrees of offspring C(t)
7. Determine P(t+1), between offspring C(t) and father P(t)
8. Add one unit to t and test limit condition
9. **End** (condition4)
10. **End** (total algorithm)

Figure 1: Procedure of proposed GA maintenance algorithm.

simultaneously. For this purpose a two dimensional chromosomes is utilized. At this representation length of chromosome is equal to total numbers of operation of present jobs for scheduling and its width is equal to 3. In the proposed GA, the first, second and third strings of chromosome indicate job (work) station, the number of machine and allocated priority to each operation, respectively. So, each problem solution is shown as a method that of course because of exiting parallel machines, includes allocation string of job (work) station as well as allocation string of machine [33].

Main factors of GA: In this section we investigate how to adjust other main factors of the proposed GA including initial population, crossover operator, mutation operator, selection approach, fitness function, impact strategy with constraints and ending criterion, respectively. After assignment transformation method of each solution to chromosome, an initial population of chromosome should be produced [34]. In the proposed GA also in order to avoid premature convergence we use an innovative method for production of initial population (Pop-Size). In the proposed GA two crossover operators of axis place carried out allocation string of machine (first and second string of chromosome) and crossover operator $R^{\wedge}x$ on string of operation sequence (third string of chromosome) are used. Because of specific structure of the problem and the proposed chromosome, an innovative mutation operator that is based on linear inversion and presented by Ferantez is established on an innovative rule as this operator work at two different mutation rates and for allocation sub problem the innovative rule of mutation operator with mutation rate (P_{m1}) and for sub problem of assignment operation sequence from inversion operator with mutation rate (P_{m2}) are utilized. As well as for evaluation chromosomes we utilize a fitness function. This function use the weighted sum function as is common for evaluating fitness value in GA. Gao et al. believed that for solving multi criteria scheduling problem with GA due to different scales of alternations amplitude each of objective function (as result of alternations amplitude of other problem parameters) in calculating fitness function of multi objective problem considering weighted sum of these objectives it is better to normalize these values during population and then their weighted sum is calculated from normalized synthetic amount of this function [13].

In the proposed GA, chromosome design is so that to minimize the problem impact with constraint and production of infeasible solutions. So, in this structure allocation string (stage and machine) will be searched continuously. In operation sequence string, there is only non-search possibilities of constraint of sequence priority are between operations of a job that will solve using correctional method. The algorithm is repeated until a stopping criterion of max-gen is achieved.

Parameters adjustment dynamically: One of the disadvantages of classic GA is premature convergence priority [20]. In GA two operators

of classic Genetic competed on how problem convergence. Although utilizing mutation operator creates diversity in populating crossover operator of population and makes it to converge [8]. Regarding this fact in adjustment of GA parameters we try to find optimal adjustment for applying of mutation and crossover operators. To improve our GA from avoiding premature convergence we may use "change of transformation rates and mutation during implementation of GA" technique. This method is based on production of each new generation using statistical data and computing distance of fitness of population member towards fitness of its best member for all population members. At this condition it is related to crossover operator and increase the mutation possibility to avoid the algorithm from the local optimal points. In opposite, if many numbers of chromosomes have not been qualified as the best individual population, then the desirable improvement is not achieved, because of just few number of chromosome have good qualities (good fitness) and the presented crossover rate doesn't create good solution effectively. To remove this problem, we can reduce mutation rate and/or increase crossover rate. So according to fitness distance of strings, mutation and crossover operators' rates are adjusted during problem solving. In the proposed algorithm, during the use of elitism technique, if numbers of best chromosome first ordered population are same, it is possible to fall into local optimal point trap. In this case, in the design of GA it is tried to only for once double the number of mutation as well as becoming half the number of crossover. While this condition is violated, the number of mutation and crossover change to the same number of problem again. As mentioned for measuring the efficiency of the proposed GA for solving the problem considering Maintenance constraint, a synthetic method for solving problem utilizing developed version of the proposed GA in order to solve two routing sub problem and assignment sequence simultaneously and then by using the efficient innovative algorithm approach we solve the Maintenance scheduling problem.

Innovative algorithm for solving sub problem

In 2006, Gao and et al. presented an innovative algorithm for solving sub problem of Maintenance scheduling flexible job shop [4]. In the present work, their algorithm for solving sub problem of Maintenance scheduling and implemented on parallel machines in the case exiting more than one Maintenance activity for each machine are developed. Steps of this innovative algorithm as follows:

Step 1: schedule Maintenance activities of total station machines in their ending window as $U_{ujr} = U_{1jr}^L$.

Step 2: Allocate operation of all jobs according to priority in sequence, for first machine available in predetermined job station and continue this job until reach to nearest scheduling window of doing Maintenance activity each machine.

Step 3: In the time of doing Maintenance activity, if availability to machine and possibility of doing operation before Maintenance activity, allocate operation to considered machine and go step 5, otherwise alternate Maintenance activity as possible as, as follows, $U_{1jr} = \text{Max} \{t_{m1j} + t_{1jr}, U_{1jr}^E\}$ to left side and close to beginning time window.

Step 4: If availability to machine and possibility of doing operation after Maintenance activity, allocate operation to machine.

Step 5: If operation of total jobs are scheduling end to algorithm, otherwise return to step 2.

Numerical Experiments

Comparison

We analyze the performance of the proposed method is evaluated in two cases. As the flexible job shop problem is a simplified version of our research problem in which the unavailability periods for machines are zero ($R=0$). So in this case, the proposed GA is compared with RK6A Meta heuristic method developed by Kurtz and Askin. In the second case, our proposed problem considering 1, 2 and 3 maintenance activities upon each machine ($R=1, 2, 3$) are studied and its obtained results are compared with the first case ($R=0$). In this case, the efficiency of the proposed solution method is investigated with respect to an index used by Gao et al. [4].

Generation of the random data

In this section, in order to generation of random problem, the first six parameters of the problem are determined according to Table 1. Note that, the first 5 parameters are extracted from Kurtz and Askin and for process speed of similar machines regarding Nahavandi and Abbasian a uniform distribution $U(1, 3)$ is applied.

In addition to parameters in Table 1, number of other four cases are considered as follows, 0, 1, 2 and 3 Maintenance activity on each machine ($R=1, 2, 3$). In general case, all components of these levels will be tested for doing numerical experiments; there are 288 possible

Factors	Levels of factors			
The number of jobs	6		30 – 100	
Distribution of machines	Variable	Constant	Variable	Constant
The number of machines	$U(1,4) - U(1,6)$	2 – 6	$U(1,4) - U(1,10)$	2 – 10
The number of stage	2 – 4 – 8			
Process timing	$U(50,70) - U(20,100)$			
Processing speed of machines	$U(1,3)$			

Table 1: Levels of factors for implementation GA.

scenarios from which 10 examples are produced. Thus, there is 2880 implementation in numerical experiments. Proposed GA is coded in C++ language and the all the 288 data sets are implemented via two algorithms.

Parameter adjustment of proposed GA

Given the obtained results of simulation, most appropriate values for algorithm parameters, i.e. the number of chromosome in elitism technique (u'), crossover rate (p_c), possibility of mutation string (p_{m1}), the possibility of mutation for string of assignment sequence (p_{m2}) are 20%, 7%, 2% and 4%, respectively.

Results of numerical experiments

In the first step, the efficiency of the proposed GA in lack of Maintenance activity on machines ($R=0$) is compared with the innovative method RKGA. As shown in Table 2, the proposed method given index "mean of objective function" in small, average and large dimensions excel from the RKGA in average of 1.33%, 1.38% and 2.12%, respectively, also the presented GA improves the solutions according to index of minimum number of objective function with the average of 0.94%.

In Table 2 different degrees of average objective function for GA in ($R=0$) with innovative method and RKGA are shown for several problem sizes. In Tables 3-5 effect of implementation considering Maintenance constraint on performance of the proposed solution approach is investigated considering one, two and three Maintenance activities on each machine ($R=1, 2, 3$).

The three separate experiments are compared with the case without the maintenance activity. In this case, efficiency of proposed solution method with aid of "maximum time period of doing Maintenance activities on machines" index is investigate by Gao et al. [4] for solving flexible job shop problem considering period of unavailability for each machine. As observed in Table 3, the average amounts of increase in objective function is lower than the sum maximum time period of

Dimensions of problem	The number of job	The number of stages	Proposed GA (R=0)			RKGA method			The degree of improvement (%)		The number of improvement times		
			Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective	Minimum objective function	The mean of objective	
Small	6	2	119	207.33	0.1	122	211.45	0.04	2.46%	1.95%	66	65	
		4	244	319.73	0.25	236	318.17	0.04	-3.39%	-0.49%	48	60	
		8	479	562.7	0.99	491	577.38	0.05	2.44%	2.54%	74	71	
	The man of small dimension		280.67	363.25	0.45	283	369	0.04	0.50%	1.33%	63	65	
Average	30	2	223	807.3	1.31	218	809.31	0.28	-2.29%	0.25%	57	62	
		4	304	901.13	3.44	313	905.22	0.25	2.88%	0.45%	78	71	
		8	553	1183.16	18.23	556	1225.13	0.24	0.54%	3.43%	65	71	
	The mean of average dimension		360	963.86	7.66	362.33	979.89	0.26	0.37%	1.38%	67	68	
Large	100	2	605	2754.29	11.53	622	2762.17	1.67	2.73%	0.29%	78	59	
		4	701	2833.54	37.26	702	2935.18	2.53	0.14%	3.46%	60	75	
		8	963	3630.34	242.47	992	3727.26	2.93	2.92%	2.60%	69	75	
	The mean of large dimension		756.33	3072.72	97.09	772	3141.54	2.38	1.93%	2.12%	69	70	
The mean of all dimensions			465.67	1466.61	35.06	472.44	1496.81	0.89	0.94%	1.61%	66	68	
General number of improvement times												595	609
The percent of all improvement times												82.64%	84.58%

*In above negative numbers demonstrate priority of RKGA method.

Table 2: Comparison results of proposed GA experiments and RKGA method.

Dimensions of problem	The number of job	The number of stages	Proposed GA (R=0)			Proposed GA (R=1)			The degree of increase (%)		Efficiency index of proposed solution method	
			Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective	Increase rate of mean objective function	sum of maximum time period of doing Maintenance activities
Small	6	2	119	207.33	0.1	123	224.21	0.13	3.25%	7.53%	14.31	116
		4	244	319.73	0.25	270	344.17	0.36	9.63%	7.10%	27.76	119
		8	479	562.7	0.99	507	585.12	1.21	5.52%	3.83%	51.65	119
	The man of small dimension		280.67	363.25	0.45	300	384.5	0.57	6.13%	6.15%	31.24	118
Average	30	2	223	807.3	1.31	254	857.43	1.77	12.20%	5.85%	39.93	118
		4	304	901.13	3.44	383	962.88	3.63	20.63%	6.41%	78.71	119
		8	553	1183.16	18.23	623	1249.18	21.78	11.24%	5.28%	100.41	117
	The mean of average dimension		360	963.86	7.66	420	1023.16	9.06	14.69%	5.85%	73.02	118
Large	100	2	605	2754.29	11.53	715	2855.61	13.06	15.38%	3.55%	116.82	117
		4	701	2833.54	37.26	821	2890.68	41.21	14.62%	1.98%	119.44	119
		8	963	3630.34	242.47	1125	3650.31	252.45	14.40%	0.55%	119.95	120
	The mean of large dimension		756.33	3072.72	97.09	887	3132.2	102.24	14.80%	2.02%	118.74	119
The mean of all dimensions			465.67	1466.61	35.06	535.67	1513.29	37.29	11.87%	4.68%	74.33	118

Table 3: Comparison results of proposed GA experiments in R=0 and R=1.

Dimensions of problem	The number of job	The number of stages	Proposed GA (R=0)			Proposed GA (R=2)			The degree of increase (%)		Efficiency index of proposed solution method	
			Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective	Increase rate of mean objective function	sum of maximum time period of doing Maintenance activities
Small	6	2	119	207.33	0.1	124	225.1	0.13	4.03%	7.90%	44.78	236
		4	244	319.73	0.25	265	362.98	0.36	7.92%	11.91%	58.24	234
		8	479	562.7	0.99	516	608.15	1.22	7.17%	7.47%	40.25	233
	The man of small dimension		280.67	363.25	0.45	301.67	398.74	0.57	6.38%	9.09%	47.75	234
Average	30	2	223	807.3	1.31	312	912.8	1.74	28.53%	11.56%	131.3	237
		4	304	901.13	3.44	407	984.43	3.65	25.31%	8.46%	78.26	234
		8	553	1183.16	18.23	617	1376.08	21.92	10.37%	14.02%	207.31	240
	The mean of average dimension		360	963.86	7.66	445.33	1091.1	9.1	21.40%	11.35%	138.96	237
Large	100	2	605	2754.29	11.53	734	2994.43	13.01	17.57%	8.02%	237.64	237
		4	701	2833.54	37.26	797	3128.18	41.27	12.05%	9.42%	236.94	236
		8	963	3630.34	242.47	1120	3886.47	252.61	14.02%	6.59%	236.11	239
	The mean of large dimension		756.33	3072.72	97.09	883.67	3336.36	102.3	14.55%	8.01%	236.9	237
The mean of all dimensions			465.67	1466.61	35.06	543.56	1608.73	37.32	14.11%	9.48%	141.2	236

Table 4: Comparison results of proposed GA experiments in R=0 and R=2.

doing Maintenance activities in small, average and large sizes (Figure 2).

According to the computations, little increase of objective function is implied and therefore the combinatorial method is demonstrated to be efficient. Therefore, the proposed method according to mean index of objective function in small, average and large sizes with mean of 6.15%, 5.85% and 2.02% increases and generally with the mean of 4.68% increase have appropriate performance, respectively. Also considering mean indicator of solution time in comparison with lack of Maintenance activity don't observe any difference between Maintenance activities. In Figure 3, average amount of increase mean of the proposed GA with sum in (R=0) and (R=1) cases for several sizes of problem are compared. Also, the obtained solution time in comparison

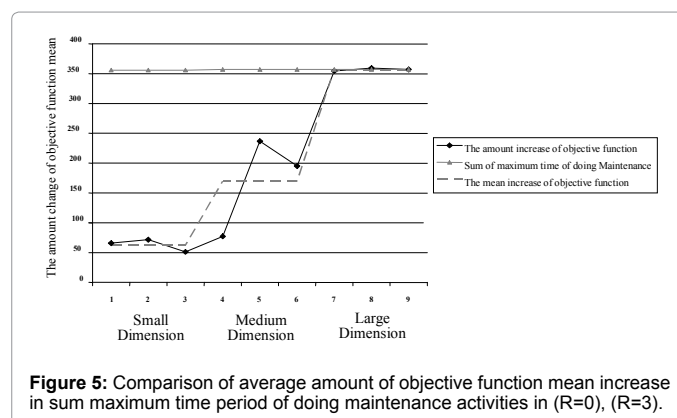
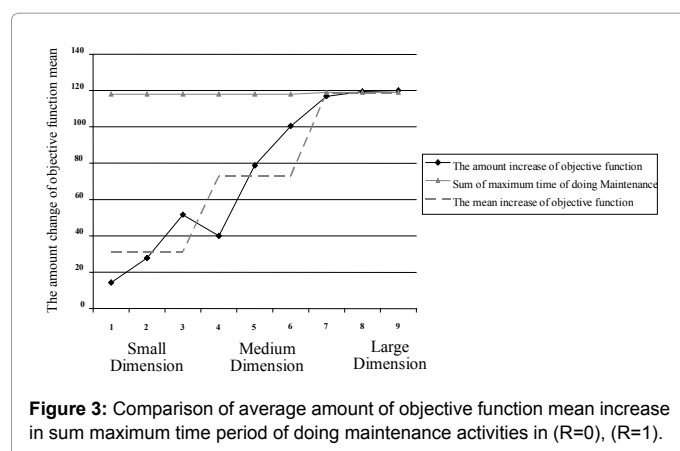
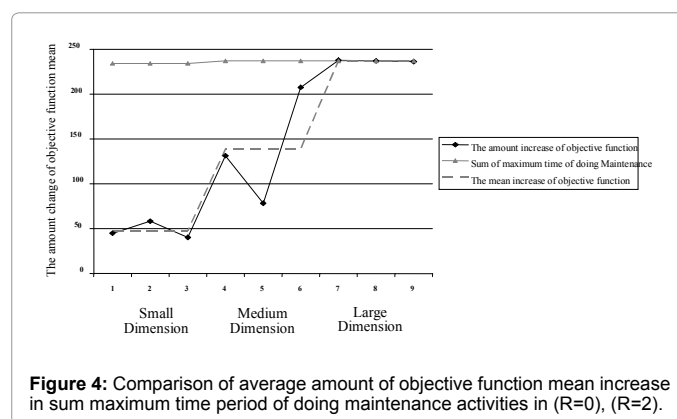
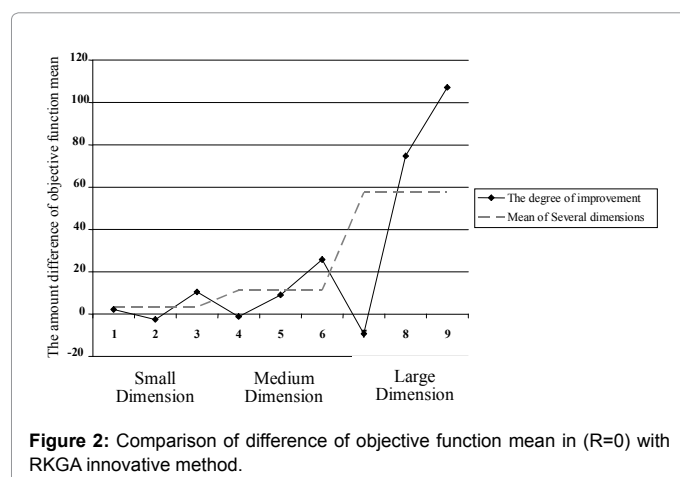
with lack Maintenance activities don't indicate any difference between two and three Maintenance activity. In Figures 4 and 5, average amount of increase mean with sum maximum time period of doing Maintenance activities upon machines are compared in cases R=0, R=2 and R=3, respectively. The obtained results for the proposed method according to index of objective function method in case one, two and three Maintenance activity with the mean of 4.68%, 9.48% and 11.75% has ideal performance, respectively.

The Effectiveness of the Proposed GA

For generating random problems with respect to 11 parameters given in Table 6, the data in Toy and Ho (2007) is employed for the

Dimensions of problem	The number of job	The number of stages	Proposed GA (R=0)			Proposed GA (R=3)			The degree of increase (%)		Efficiency index of proposed solution method	
			Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective function	The mean of solution time (s)	Minimum objective function	The mean of objective	Increase rate of mean objective function	sum of maximum time period of doing Maintenance activities
Small	6	2	119	207.33	0.1	124	236.23	0.13	4.03%	12.23%	0.81	1
		4	244	319.73	0.25	275	366.45	0.31	11.27%	12.75%	0.58	1
		8	479	562.7	0.99	509	619.15	1.04	5.89%	9.12%	0.9	1
	The man of small dimension		280.67	363.25	0.45	302.67	407.28	0.49	7.07%	11.37%	0.76	1
Average	30	2	223	807.3	1.31	316	888.75	1.49	29.43%	9.16%	0.69	1
		4	304	901.13	3.44	412	1102.93	3.93	26.21%	18.30%	0.96	1
		8	553	1183.16	18.23	625	1357.73	21.44	11.52%	12.86%	0.78	1
	The mean of average dimension		360	963.86	7.66	451	1116.47	8.95	22.39%	13.44%	0.81	1
Large	100	2	605	2754.29	11.53	727	3116.39	14.49	16.78%	11.62%	0.95	1
		4	701	2833.54	37.26	805	3230.75	40.46	12.92%	12.29%	0.74	1
		8	963	3630.34	242.47	1115	3921.23	253.01	13.63%	7.42%	0.85	1
	The mean of large dimension		756.33	3072.72	97.09	882.33	3422.79	102.65	14.44%	10.44%	0.85	1
The mean of all dimensions			465.67	1466.61	35.06	545.33	1648.84	37.37	14.63%	11.75%	0.81	1

Table 5: Comparison results of proposed GA experiments in R=0 and R=3.



first 8 parameters and for other parameters the data in Kurtz and Askin [21-23] are utilized. Delivery date of jobs relation is introduced by Broker [9]. Parameter C in above relation is tightness

index of delivery date that as much as the amount of this parameter are more for job, it will show weak related delivery date of job. Generally, all components at this level are tested. Given Table 6 there are 12 experimental scenarios having 10 data set generated for job, we implement the algorithms by the 120 data sets. Tay and Ho alternated the processing time P_{ij} to adapt this formula to Mo- FJSP problem with

its flexibility operation, and proposed the relation of Broker with $\overline{p_{i,j}}$ to compute relation (15). At this relation $n(f(o_{ij}))$ demonstrates the number of referential job stages for doing its job operations. We also utilize the following added relations to adapt this formula with the research problems MO-FDJSPM:

$$\overline{p_{ij}} = \frac{\sum_{n(f(o_{ij}))} p_{i,j,k}}{n(f(o_{ij}))} \quad (15)$$

$$\overline{p_{ij}} = \frac{\sum_{k=1}^m a_k^{(i,j)} \times [\sum_{pm=1}^{l_k} p_{i,j,k} / S_{k,pm}]}{\sum_{k=1}^m a_k^{(i,j)} \times l_k} \quad (16)$$

where $a_k^{(i,j)}$ is a binary variable that is 1 if in a processing stage a machine is a possible substitute for O_{ij} and otherwise it is zero. $S_{k,pm}$ and l_k demonstrate the speed of machine $M_{k,pm}$ and number of parallel machines for k^{th} job station. For more simplicity the penalty for delay is considered same for all parts.

Parameter	Values	States
Flexibility	100% (FJSP-100); 50% (FJSP-50); 20% (FJSP-20) [1 × m]; [0.5 × m]; [0.2 × m]	1
nJobs × nMachines	10 × 5,20 × 5,50 × 5,20 × 10,50 × 10,100 × 10,50 × 15,100 × 15 and 200 × 15	1
Processing time ($p_{i,j,k}$)	U[(nMachines)/2, (nMachines) × 2]	1
$dev(p_{i,j,k}, p_{i,j,l})$	5	1
# of Operation	U[m/2, m]	1
Release Date (r_i)	If nJobs ≥ 50: U[0,40]; Otherwise, U[0,20]	1
Tightness factor of Due Date(c)	1.2 (tight), 1.5 (moderate), 2 (loose)	3
Due Date (d)	$r_i + c \times \sum_{j=1}^{n_j} p_j$	1
Machine distribution	Constant Variable 2 -	2
# of machines (L_i)	n-U[1,4] - U[1, n]	2
Speed of machines	U[1,3]	1
Number of Scenarios		12

Table 6: Factors level of Mo-FOJSPM problem.

Parameters adjustment for the proposed GA

Here, procedure of parameter adjustment is utilized for adjusting four parameters entitled, (M'), (P_c), (P_{m1}), (P_{m2}). For this purpose, from each experimental scenario a data set and totally 72 data sets are generated randomly and then in procedure of parameters adjustment with production of each new parameter the GA is implemented upon 72 data sets. Also, the following parameters with the corresponding specifications are considered:

Percent selecting of best chromosome for next generation (M'): 5%, 10%, 15%, 20%, 25% and 30%, possible cross over (P_c): 3%, 4%, 5%, 6%, 7%, 8% and 9%, possible mutation (P_{m1}) and (P_{m2}): 5%, 1%, 15%, 2%, 25%, 3%, 4% and %5, the number of initial population (Pop- Size): 100, the number of repeats (Max- Gen): 125. The result of experiments are obtained to be 20%, 7%, 2% and 25%, for (M'), (P_c), (P_{m1}) and (P_{m2}), respectively.

Method of doing numerical experiments

The algorithm is encoded in C++ language and implemented using a computer with Pentium IV (CPU₃ GH_z 1GB of RAM) in Borland C++ 5/02 environment. Each of the algorithms (proposed GA and GP) are implemented using the same 120 generated data sets. To compare the proposed GA with the GP algorithm we utilize the numbers of weighted sum function and the collected parameters listed in Tables 7-9, briefly.

Results of GA experiments

The followings are the results of a survey of the algorithm:

1) The advantage of the proposed GA rather GP with respect to "minimum value of minimization objective function of maximum weighted sum" index in small, average and large sizes are reported with the mean of 2.65%, 3.16% and 5.90%, respectively, and totally with the mean 3.90% which is a noticeable improvement.

2) The "Mean of objective function" could be the best criterion for evaluating the algorithms. The proposed GA in small, average and large sizes with the mean of 4.34%, 3.64% and 5.34%, respectively, and totally with the mean of 4.90% has better performance than the GP algorithm.

3) The only strength of the GP algorithm against the proposed GA according to the results of Table 7 and with respect to "mean solution time" index in small, average and large sizes is revealed. Of course,

The number of machines			5				10				15				Mean of all dimensions
The number of job			10	20	50	Total	20	50	100	Total	50	100	200	Total	
Proposed Genetic algorithm GA	Objective function	best	12.53	20.4	43.34	12.53	105.68	300.32	549.04	105.68	542.72	1006.63	1754.17	542.72	220.3
		mean	16.9	21.24	56.09	31.41	115.66	305.47	561.87	327.67	542.5	1027.48	1804.64	1124.87	494.7
	mean of solution time (second)		0.35	0.88	8.61	3.28	0.71	12.09	69.21	27.33	28.18	170.89	521.18	240.08	90.23
Algorithm GP	Objective function	best	12.12	21.35	46.53	12.12	105.26	315.23	578.82	105.26	580.82	1067.48	1854.89	580.82	232.7
		mean	16.51	23.33	59.93	33.26	116.02	322.01	594.38	344.13	598.38	1087.47	1906.42	1197.42	524.9
	mean of solution time (second)		0.02	0.028	0.035	0.0277	0.02	0.0312	0.0412	0.0308	0.023	0.0342	0.0538	0.037	0.032
rate of improvement	Objective function	best	-3.35%	4.45%	6.84%	2.65%	-0.40%	4.73%	5.15%	3.16%	6.56%	5.70%	5.43%	5.90%	3.90%
		mean	-2.37%	8.98%	6.41%	4.34%	0.31%	5.14%	5.47%	3.64%	9.34%	5.52%	5.34%	6.73%	4.90%

Table 7: Comparison numerical findings resulted from implementation of proposed GA and GP for FJSP-20%.

The number of machines			5				10				15				Mean of all dimensions
The number of job			10	20	50	Total	20	50	100	Total	50	100	200	Total	
Proposed Genetic algorithm GA	Objective function	best	10.75	21.2	53.09	10.75	94.2	227.31	514.93	94.2	443.29	781.86	1627.9	443.29	182.75
		mean	12.46	23.64	54.61	30.24	97.92	243.04	525.47	288.8	454.33	829.03	1686.5	989.94	436.33
	mean of solution time (second)		0.36	0.9	8.78	3.34	0.72	12.33	70.59	27.88	28.74	174.31	532	244.88	92.04
Algorithm GP	Objective function	best	10.66	22.41	57.12	10.66	95.77	238.05	545.93	95.77	472.09	829.82	1731	472.09	192.84
		mean	12.32	26.05	59.79	32.72	100.84	257.73	559.23	305.9	484.95	865.01	1796.4	1048.8	462.48
	mean of solution time (second)		0.02	0.028	0.035	0.028	0.02	0.0312	0.0412	0.031	0.0262	0.0374	0.05	0.0391	0.0325
rate of improvement	Objective function	best	-0.81%	5.40%	7.06%	3.88%	1.64%	4.51%	5.68%	3.94%	6.10%	5.78%	###	5.94%	4.59%
		mean	-1.12%	9.24%	8.65%	5.59%	2.89%	5.70%	6.04%	4.88%	6.31%	4.16%	6.12%	5.53%	5.33%

Table 8: Comparison numerical findings resulted from implementation of proposed GA and GP for FJSP-50%.

The number of machines			5				10				15				Mean of all dimensions
The number of job			10	20	50	Total	20	50	100	Total	50	100	200	Total	
Proposed Genetic algorithm GA	Objective function	best	8.73	20.43	47.11	8.73	102.29	228.18	489.14	102.29	407.51	773.51	1529.05	407.51	172.84
		mean	12.99	22.34	51.05	28.79	105.05	243.42	511.18	286.55	412.27	803.21	1562.26	925.91	413.75
	mean of solution time (second)		0.36	0.91	8.9	3.39	0.73	12.49	71.51	28.24	29.11	175.57	538.52	247.73	93.12
Algorithm GP	Objective function	best	8.61	21.76	49.96	8.61	102.93	242	519.92	102.93	432.37	816.4	1650.6	432.37	181.3
		mean	12.78	23.69	55.08	30.52	104.88	260.11	545.39	303.46	442.46	844.51	1659.08	982.02	438.66
	mean of solution time (second)		0.02	0.031	0.0479	0.0328	0.02	0.0608	0.0866	0.0558	0.0083	0.09	0.117	0.0718	0.0535
rate of improvement	Objective function	best	-1.47%	6.11%	5.70%	3.45%	0.63%	5.71%	5.92%	4.09%	5.75%	5.25%	7.36%	6.12%	4.55%
		mean	-1.68%	5.70%	7.31%	3.78%	-0.17%	6.41%	6.27%	4.17%	6.82%	4.89%	5.84%	5.85%	4.60%

Table 9: Comparison numerical findings resulted from implementation of proposed GA and GP for FJSP-100%.

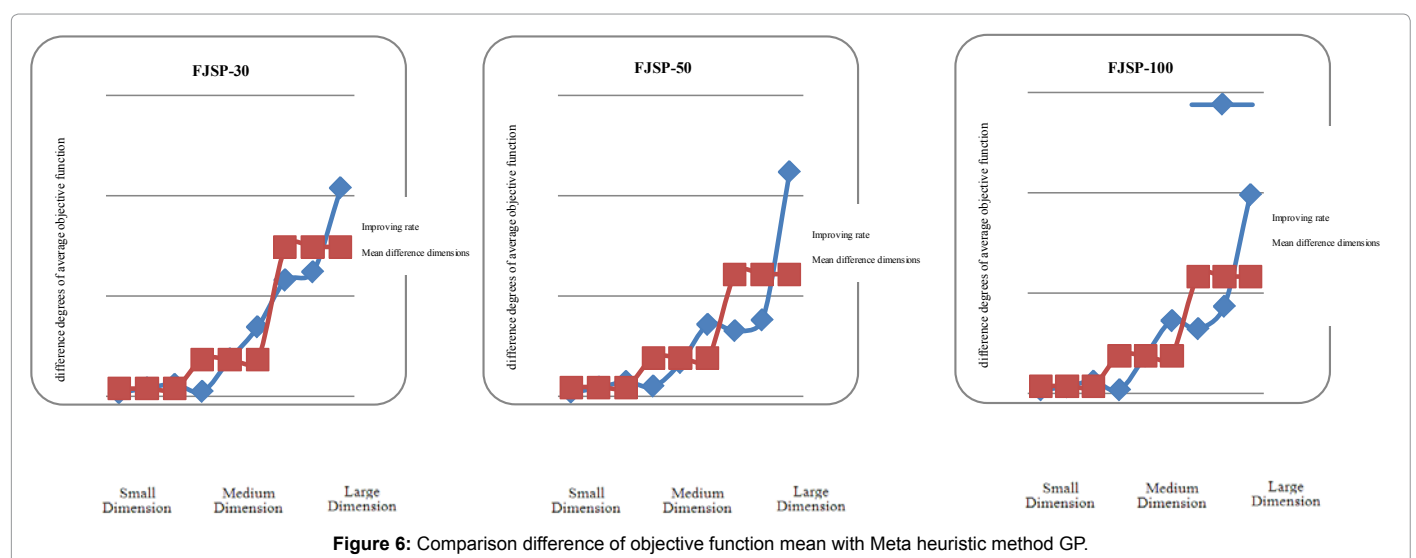


Figure 6: Comparison difference of objective function mean with Meta heuristic method GP.

the improvements done by GA justify the longer solution time of this algorithm against the GP (Figure 6).

Conclusions

In this paper, a problem of flexible multi-objective scheduling

job shop with parallel machines in dynamic job shop environment, Mo-FOJSPM considering maintenance constraint for machines is introduced. Given parameters of the problem, an analytical mathematical model was developed and it was proved that the problem is NP-hard due to conventional solution approaches disability for handling the problem. Thus, a genetic algorithm was proposed and adjusted for the proposed scheduling problem. Also, an innovative algorithm was designed for solving maintenance sub problem. Due to a classical problem in GA, premature convergence, a technique was utilized to employ the model knowledge not to trap in local optimum in an intelligent manner causing more efficiency of the searching process in the solution space and finding the optimal solutions. Several parameters adjustment and different data set generation was set to perform the numerical experiments. Also, a comprehensive numerical discussion for different cases of the problem depicting in tables and figures were presented. The numerical results and the performed comparisons verified the effectiveness of the proposed GA against GP in the proposed Mo-FOJSPM.

References

- Brandimarte P (1993) Routing and scheduling in a flexible job shop by taboo search. *Annals of Operations Research* 41: 157-183.
- Tay JC, Wibowo D (2004) An effective chromosome representation for evolving flexible job-shop scheduling. *Lecture Notes in Computer Science* 3103: 210-221.
- Kim YK, Park K, Ko J (2003) A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling. *Computers and Operations Research* 30: 1151-1171.
- Gao J, Gen M, Sun L (2006) Scheduling jobs and maintenances in flexible job shop with a hybrid genetic algorithm. *Journal of Intelligent Manufacturing* 17: 493-507.
- Adiri I, Bruno J, Frostig E, Rinnooy Kan AHG (1989) Single machine flow-time scheduling with a single breakdown. *Acta Informatica* 26: 679-696.
- Schmidt G (2000) Scheduling with limited machine availability. *European Journal of Operational Research* 121: 1-15.
- Xie J, Wang X (2002) Branch and bound algorithm for flexible flowshop with limited machine availability. *Asian Information Science Life* 1: 241-248.
- Cheng TCE, Liu Z (2003) Approximability of two-machine flow shop scheduling with availability constraints. *Operations Research Letters* 31: 319-322.
- Xie J, Wang X (2005) Complexity and algorithms for two-stage flexible flow shop scheduling with availability constraints. *Computers and Mathematics with Applications* 50: 1629-1638.
- Wu CC, Lee WC (2006) A note on single-machine scheduling with learning effect and an availability constraint. *Int J Advance Manufacturing Technology* 33: 540-544.
- Breit J (2007) Improved approximation for non-preemptive single machine flow-time scheduling with an availability constraint. *European Journal of Operational Research* 183: 516-524.
- Zegordi H, RahimiM (2010) Job shop scheduling with maintenance constraint. *Journal of Sharif* 46: 131-137.
- Brucker P, Jurisch B, Sievers B (1994) A branch and bound algorithm for the job-shop scheduling problem. *Discrete Applied Mathematics* 49: 107-127.
- Carlier J, Pinson E (1989) An algorithm for solving the job-shop problem. *Management Science* 35: 164-176.
- Tay JC, Ho NB (2008) Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems. *Computer and Industrial Engineering* 54: 453-473.
- Brandimarte P (1999) Theory and methodology exploiting process plan flexibility in production scheduling a multi-objective approach. *European Journal of Operational Research* 114: 59-71.
- Ghedjati F (1999) Genetic algorithms for the job-shop scheduling problem with unrelated parallel constraints heuristic mixing method machines and precedence. *Computer and Industrial Engineering* 37: 39-42.
- Kacem I, Hammadi S, Borne P (2002) Approach by localization and multi objective evolutionary optimization for flexible job-shop scheduling problems. *IEEE Transaction on Systems, Man and Cybernetics* 32: 1-13.
- Moreno-Torres JG, Llorca X, Goldberg DE (2009) Binary Representation in Gene Expression Programming: Towards a Better Scalability. *IlliGAL Report No. 2009003*.
- Gen M, Cheng R (2004) Genetic algorithm in search, optimization and machine learning. *Addition-Wesley, Reading, MA*.
- Kurz ME, Askin RG (2004) Scheduling flexible flow-lines with sequence-dependent setup times. *European Journal of Operational Research* 159: 66-82.
- Kurz ME, Askin RG (2003) Comparing scheduling rules for flexible flow-lines. *Int J Production Economics* 85: 371-388.
- Ho NB, Tay JC, Lai E (2007) An effective architecture for learning and evolving Flexible Job Shop schedules. *European Journal of Operational Research*, 179: 316-333.
- Reeves Coline R (1993) Modern Heuristic Techniques for Combinatorial Problems John Wiley and Sons.
- Dagli CH, Sittisathanchai S (1995) Genetic neurons-scheduler: a new approach for job shop scheduling. *International Journal of Production Economics* 41: 135-145.
- Abbasian M, Nahavandi N (2009) Minimization Flow Time in a Flexible Dynamic Job Shop with Parallel Machines. *Engineering Department of Industrial Engineering, Master of Science Thesis*.
- Abbasian M, Nahavandi N (2010) Minimization Flow Time in a Flexible Dynamic Job Shop with Parallel Machines. *Journal of Sharif*.
- Amiri M, Jamshidi SF, Sadeghiani JS (2009) A Genetic Algorithm Approach for Statistical Multi-Response Models Optimization: A Case Study. *Journal of Science and Technology* 49: 131-137.
- Abbasian M, Nahavandi N (2010) Solving Multi-Objective Flexible Dynamic Job-Shop Scheduling Problem with Parallel Machines. *International Journal of Industrial Engineering of Production Research* 21: 1-12.
- Verma A, Llorca X, Venkataraman S, Goldberg DE, Campbell RH (2010) Scaling eCGA Model Building via Data-Intensive Computing. *IlliGAL Report No. 2010001*.
- Kubzin MA, Strusevich VA (2005) Two-machine flow shop no-wait scheduling with machine maintenance. *A Quarterly Journal of Operations Research* 3: 303-313.
- Lee YH, Jeong CS, Moon C (2002) Advanced planning and scheduling with outsourcing in manufacturing supply chain. *Computer and Industrial Engineering* 43: 351-374.
- Jansen K (2001) Approximation Algorithms for Flexible Job Shop Problems. *International Journal of Foundations of Computer Science*: 68-77.
- Low C (2005) Simulated annealing heuristic for flow shop scheduling problem with unrelated parallel machines. *Computer and operation Research* 32: 2013-2025.