Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 9270802, 32 pages
https://doi.org/10.1155/2018/9270802

Review Article

Recent Research Trends in Genetic Algorithm Based Flexible Job Shop Scheduling Problems

Muhammad Kamal Amjad,[1] **Shahid Ikramullah Butt**,[1] **Rubeena Kousar**,[2] **Riaz Ahmad**,[3] **Mujtaba Hassan Agha**,[4] **Zhang Faping**,[5] **Naveed Anjum**,[1] **and Umer Asgher**[1]

[1]*School of Mechanical and Manufacturing Engineering, National University of Sciences and Technology, Islamabad, Pakistan*
[2]*Department of Mechanical Engineering, University of Engineering and Technology, Taxila, Pakistan*
[3]*Directorate of Quality Assurance, National University of Sciences and Technology, Islamabad, Pakistan*
[4]*Department of Mechanical Engineering, Capital University of Sciences & Technology, Islamabad, Pakistan*
[5]*Department of Mechanical Engineering, Beijing Institute of Technology, Beijing, China*

Correspondence should be addressed to Muhammad Kamal Amjad; kamal.amjad@smme.edu.pk

Received 9 May 2017; Revised 2 January 2018; Accepted 28 January 2018; Published 28 February 2018

Academic Editor: Thomas Hanne

Flexible Job Shop Scheduling Problem (FJSSP) is an extension of the classical Job Shop Scheduling Problem (JSSP). The FJSSP is known to be NP-hard problem with regard to optimization and it is very difficult to find reasonably accurate solutions of the problem instances in a rational time. Extensive research has been carried out in this area especially over the span of the last 20 years in which the hybrid approaches involving Genetic Algorithm (GA) have gained the most popularity. Keeping in view this aspect, this article presents a comprehensive literature review of the FJSSPs solved using the GA. The survey is further extended by the inclusion of the hybrid GA (hGA) techniques used in the solution of the problem. This review will give readers an insight into use of certain parameters in their future research along with future research directions.

1. Introduction

The share of manufacturing sector in the Gross Domestic Product (GDP) of the world is up to 18% thus making it extremely important to the worldwide economy [1]. Efficient manufacturing leads to improvement in profits, market share, and ultimately a competitive advantage in new product launch time [2]. Manufacturing needs to have efficient and optimal operations of the facility which were later termed as "scheduling." Owing to the importance of the subject, huge amount of research has been conducted to formulate techniques, separately for each shop type, which can effectively handle the complex problem of scheduling.

Genetic Algorithm has proven to be one of the most effective evolutionary techniques for solving Job Shop Scheduling Problem (JSSP) and consequently Flexible Job Shop Scheduling Problem (FJSSP). Çaliş and Bulkan [3] pointed out that 26.4% of the research studies for solution of JSSP have been conducted using GA. This is the highest percentage of any artificial intelligence based technique used for the solution of the said problem which became motivation for this review paper.

This paper critically analyzes the state-of-the-art Flexible Job Shop Scheduling Problem (FJSSP) solution techniques belonging to the GA class. In this review paper, Section 2 introduces the machine layouts and a classification scheme. FJSSP is then presented along with formulation and complexity along with scheduling algorithms. Section 3 gives an insight to the Genetic Algorithms (GA), basic elements, and their adaptation for the solution of FJSSP. Section 4 presents the schematic review of literature for obtaining solution of FJSSP with GA, advanced GA, and hybrid GA (hGA) approaches. Section 5 provides analysis and discussion and afterwards Section 6 presents the conclusion. Notations

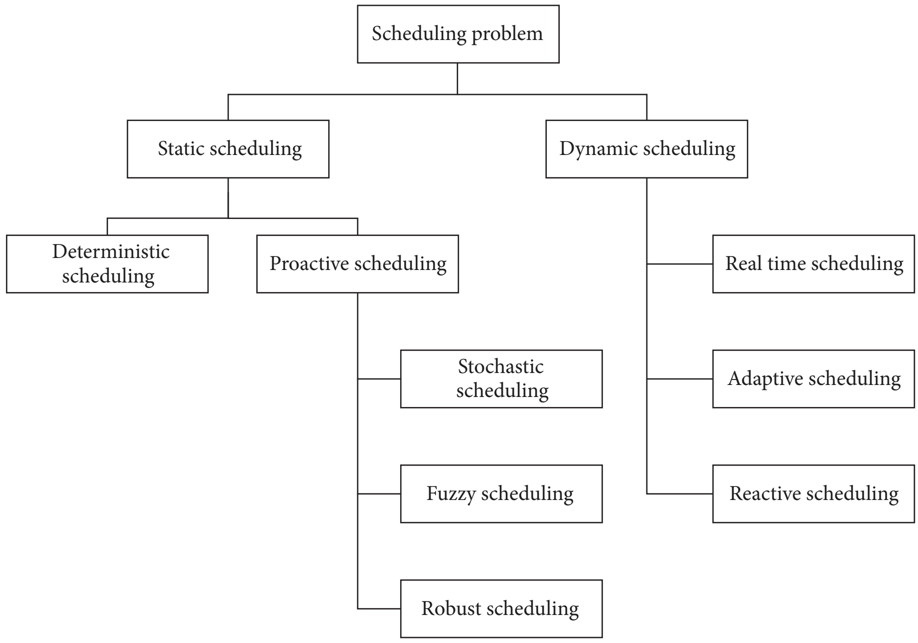

Figure 1: Classification of scheduling problem.

are widely used in this paper for clutter-free presentation of literature, which have been summarized in Notations. All other abbreviations are explained in paper where they appear for the very first time.

2. Manufacturing Scheduling

2.1. Scheduling. Scheduling refers to the allocation of tasks (e.g., jobs, parts, and operations) to resources (e.g., machines) in such a way that they can be processed and/or manufactured in an optimal manner [4]. The consumer wants to get the product delivered at required time and hence scheduling becomes a critical factor in meeting this demand [5] and plays a vital role in the operation of any manufacturing environment. The scheduling problem aims to formulate a processing order that can achieve a desired objective in an optimal manner which can be total time required for completing all operations, maximum lateness, maximum earliness, and so on. Therefore schedules can be generated to attain various performance measures of the shop floor. Scheduling can be of the following two types:

(i) Static: jobs arrive at an idle machine after a fixed time interval.

(ii) Dynamic: jobs arrive in random manner.

Dynamic scheduling is considered a situation when any disruption occurs in the manufacturing environment in contrast to the static scheduling. This may require necessary changes in the schedule so that it can remain optimal. Such problems are classified as job and/or recourse related [6]. Due to the importance of scheduling in manufacturing environments, handsome literature is published in this area. Some of the salient works on scheduling in a general context are included in references [7–12], whereas the classification of scheduling problem is presented in Figure 1 [13].

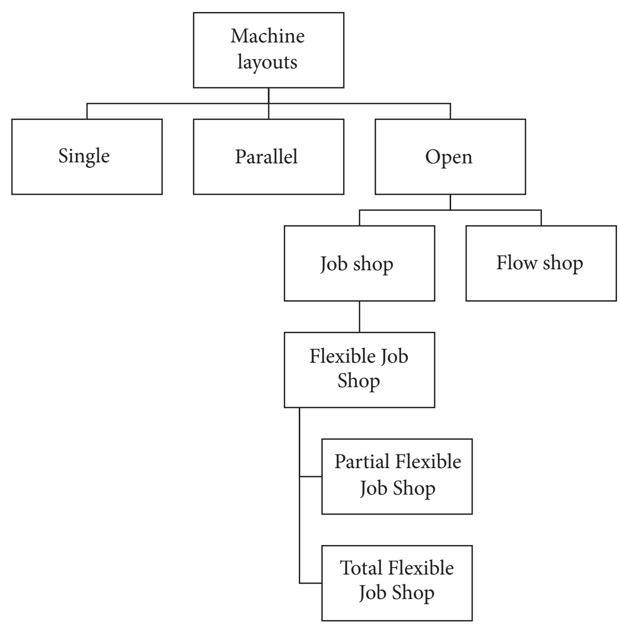

Figure 2: Classification of shop layouts.

2.2. Classification of Machine Layouts. Based upon the requirement of manufacturing process and product requirements, the machine shops have been classified in various layouts. Figure 2 presents a schematic classification of the machine layout with emphasis on the Job Shop. The JSSP is a classical combinatorial optimization problem which has been attracting research interest since 1950s [14, 15]. JSSP has the following salient features:

(i) It deals with the sequencing of a number of operations on fixed machines.
(ii) Every job can have a different processing time.
(iii) Each job must undergo a set of tasks performed in a given manner on different machines in order to be completed.

The FJSSP is a further extension of JSSP in which the operations can be performed on any machine which can be selected from a finite number of given set of machines in a flexible manufacturing cell. Thus the problem is intricate in a sense that it also involves machine assignment problem for each operation and thus it is subdivided into following two parts:

(i) Routing, through which the jobs should be processed on available set of machines
(ii) Sequencing, that is, the order in which the jobs should be processed on the selected machines.

Thus there is inherent "flexibility" in the FJSSP in contrast to the JSSP, which may be used as advantage for processing various types of parts, both through routing and sequencing. Flexibility has been introduced in the classical JSSP in some of the following ways:

(i) The idea of FJSSP was first adapted by Brucker and Schlie [16] as multipurpose machines equipped with different tools.
(ii) Barnes and Chambers [17] argued that a JSSP can be converted into FJSSP by incorporating multiple instances of a single machine where a bottleneck is encountered during the scheduling process. This concept is sometimes called parallel machine FJSSP.
(iii) Najid et al. [18] argued that flexibility is brought in the JSSP with the condition that one machine may be able to perform more than one type of operation.

Kacem et al. [19] classify the FJSSPs into the following types:

(i) Total FJSSP (T-FJSSP): in this type, required operation can be performed on any of the available identical machines in the machine cell; thus complete flexibility has been achieved.
(ii) Partial FJSSP (P-FJSSP): in this type, some operations can only be performed on specific machines and remaining operations can be executed on any of the machines in the machine cell.

According to Chan et al. [20], there are the following two types of FJSSP:

(i) Type I FJSSP: in this type, jobs under consideration have different operation sequences and identical/nonidentical machines for each operation. In this problem, the interest is to find the operation's sequence and job processing order.
(ii) Type II FJSSP: in this type, jobs under consideration have fixed operation sequences, but different identical or nonidentical machines for each operation. In this problem, the interest is to arrange jobs on machines according to their operation sequences.

2.3. Optimization. A schedule for any manufacturing product has to be optimum in order to obtain effectiveness. Optimization refers to obtaining the best solution in a solution space with respect to some predefined criteria [21, 22]. The criterion to be minimized or maximized is called objective function. For constrained optimization, the objective function is to be optimized keeping in view the constraints which govern the system. When viewed from a manufacturing system perspective, optimized process produces maximum output with minimum input, or vice versa, as desired. Figure 3 presents a flow of a generic optimization process.

A general optimization problem can be defined as follows:

$$\begin{aligned} \text{Minimize/maximize } (\text{objective function}) \quad & z = f(x) \\ \text{Subject to } (\text{constraints}) \quad & g_i(x) \le 0 \\ & h_i(x) = 0 \\ & x \ge 0; \\ & i = 1, 2, \dots, n, \end{aligned} \tag{1}$$

where x is the decision variable and g and h are inequality and equality constraints, respectively. The model presented above is for single objective optimization. The multiobjective optimization problem is formulated as follows:

$$\begin{aligned} \text{Minimize/maximize } (\text{objective function}) \quad & z = f(x) = (f_1(x), f_2(x), \dots, f_k(x)) \\ \text{Subject to } (\text{constraints}) \quad & g_i(x) \le 0; \quad i = 1, \dots, p \\ & h_j(x) = 0; \quad j = 1, \dots, q. \end{aligned} \tag{2}$$

The function $f(x)$ is a k-dimensional vector of objective functions, where k is the total number of objective functions ($k \ge 2$), p is the number of inequality constraints, and q is the number of equality constraints.

Multiobjective optimization is more complex than the single objective optimization due to the fact that simultaneous minimization of two or more functions can lead to a situation where decreasing one function further may cause

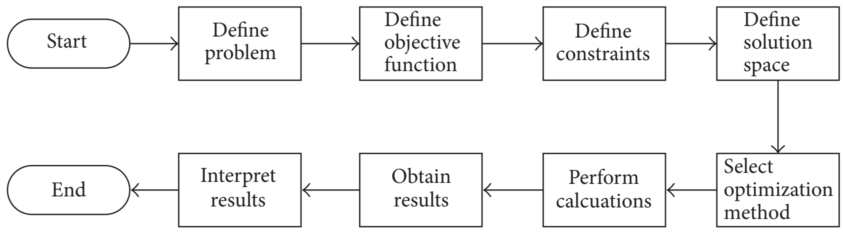

Figure 3: A generic optimization process.

the other function to increase. To address this optimization issue, the concept of Pareto optimality [23] is used. A Pareto optimal point is such a point in a feasible design space where further decreasing any function beyond that point will result in the increase of other functions. Another approach for multiobjective optimization is to assign weights to different objects and formulate a weighted single objective optimization problem.

2.4. FJSSP Formulation and Complexity. The classical JSSP can be formulated as follows [209]:

(i) A set of n jobs are available to be scheduled on m machines.

(ii) The set of jobs is denoted by J ($J = J_1, J_2, \ldots, J_n$).

(iii) The set of machines is denoted by M ($M = M_1, M_2, \ldots, M_m$).

(iv) Each job i consists of a sequence of n_i operations.

(v) Each operation $O_{i,j}$ of job i has to be processed on one machine, M_k out of the given set of machines, M ($i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, n_i$).

(vi) The processing time for each operation $O_{i,j}$ is predetermined as $t_{i,j,k}$ on each machine.

For FJSSP the following additional parameters are added [210]:

(i) Each operation can be processed on one M_k out of the available machines such that $M_k \in M_{i,j}$ and $M_{i,j} \subseteq M$.

(ii) For P-FJSSP, $M_{i,j} \subset M$.

(iii) For T-FJSSP, $M_{i,j} = M$.

It is generally assumed for FJSSP that all machines and jobs are available at time $t = 0$ and one machine can only process one operation at a time such that jobs are independent from each other; thus no priority restriction exists.

Initially, the Job Shop Scheduling Problem (JSSP) either was not solvable or could take excessive time period for obtaining solution. In context of computational complexity, the JSSP is NP-hard [211] and it belongs to one of the most difficult problems in this class [212]. This is due to the fact that, in a JSSP, every job can have a different and separate processing time; thus the complexity of the problem grows with the number of jobs.

Framinan et al. [12] have shown that it will take 1.68 billion years to evaluate all possible solutions for 30 jobs to be scheduled on a single machine with a fast running computer at 5 Picohertz (PHz). Similarly in a state-of-the-art survey of JSSP complexity, Brucker et al. [213] pointed out that JSSP can go up to binary NP-hard class. As FJSSP is a further extension of the classical JSSP, it is further complex. A schedule for JSSP with n jobs and m machines will have $(n!)^m$ possible sequences [214]. Therefore an exact solution to these problems cannot be found in a reasonable time keeping in view the manufacturing priorities. The computation time increases exponentially for NP-hard problems with a linear increase in size of problems [215].

2.5. Scheduling Algorithms. According to Cormen et al. [216], algorithms are a sequence of activities which can transform an input value to a desired output, hence serving as a tool for solving a specified computational problems. The origins of algorithms can be traced back to 8th century when Al-Khwarizmi defined steps for solution of quadratic equations [217]. With the immense increase in the computational power, more and more complex calculations can now be performed to address various issues and thus more advanced algorithms have been developed. Figure 4 presents a classification for the scheduling algorithms. This classification is not exhaustive and only contains a broad view of the algorithm classes.

Exact algorithms guarantee that there will be no better solution after a problem has been solved. However, as mentioned earlier, the complexity of the FJSSP is of extreme nature and there is very limited scope for the use of exact algorithms. In the modern era, approximate algorithms have gained extreme popularity due to the fact that problems have become more complex and the need to reach the solution in a reasonable time has become a prominent research area.

3. Genetic Algorithms

GA belongs to the evolutionary algorithms class and its development was inspired through the process of natural genetic evolution. The original work on natural evolution was contributed by Darwin [218] in which he claimed that natural populations evolve according to the process of natural selection on the basis of "survival of the fittest" rule. Initial work on GA was conducted by Holland [219] in 1975, which was then extended majorly by Goldberg [220].

Giraffes use their long necks to eat the leaves at higher parts of the plants. Thus as per the rule of the survival of the fittest, giraffes have evolved with generations having longer

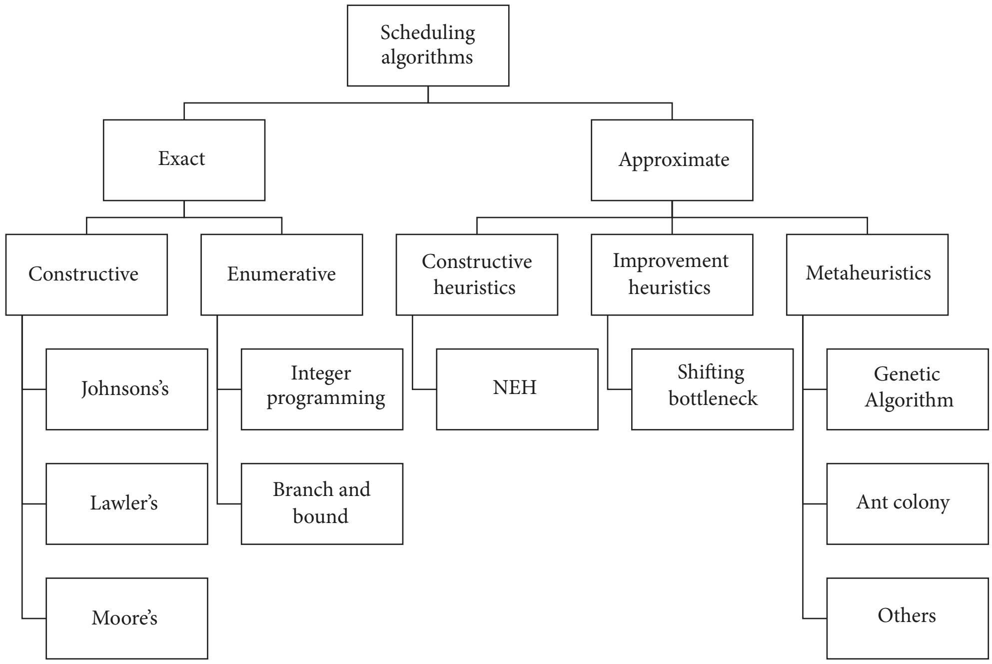

FIGURE 4: Scheduling algorithms.

necks. The GAs can be used to mimic this natural process of genetic evolution on the principal of survival of the fittest to obtain solutions to the engineering problems. The beauty of GA lies in its adaptive nature; that is, it can change/fit itself according to the changing environment. Next section explains basic working of GA.

3.1. Basic Elements of GA. The basic working element of GA is gene, a group of which constitutes a chromosome. The chromosomes contain the current state data coded in the form of binary digits 0 or 1 which is distinctively stored in a gene. This structure represents a candidate solution to the problem in consideration. GA works on these coded forms of the data instead of working on actual data elements. The chromosomes combine to form a population which in turn formulates a generation. GA is an iterative evolutionary process which formulates a generation after each iteration. Figure 5 represents the schematic representation of the relation of these elements.

3.2. Genetic Operators. Each generation is subjected to the genetic operators to obtain a new generation. The new generation is theoretically better than the previous generation, as the new generation is generated after implementing the principle of "survival of the fittest" and thus it replaces the older generation. During this process, either the whole population can be changed or only the worst chromosome can be replaced [221]. Obviously, these are two extreme methods and several strategies for new population can be formulated.

The iterations are guided in a way that they satisfy a fitness criterion and they are repeated to obtain an acceptable generation. The genetic operators are used to bring in the beauty of randomization in the algorithm. Standard GA operators are presented in the following.

3.2.1. Selection. Selection operator is used to select chromosomes in a generation based upon fitness. The chromosomes satisfying the fitness criteria are likely to be selected in each newer generation. Generally used selection criteria are as follows:

(i) Roulette wheel selection: the selection probability of a chromosome is directly proportional to its fitness as assessed by the fitness criteria. Thus a chromosome with higher fitness will have more probability to be selected; however, lower fitness chromosomes may also be selected.

(ii) Rank based fitness assignment: this method associates relative fitness between individual chromosomes, hence preventing a generation from containing an all-fit chromosome structure. The method is mainly used to maintain diversity in the population.

(iii) Tournament selection: *a set of chromosomes are selected randomly and then the fittest chromosomes are selected* for further operation. This method is completely random.

(iv) Elitism: the crux of this method is that it maintains a fixed number of fittest chromosomes and the rest of the population is generated by using any of the preferred selection methods. Thus this method not only ensures that the best solutions remain in the population, but also ensures the diversification of the population by selecting chromosomes from the entire solution space.

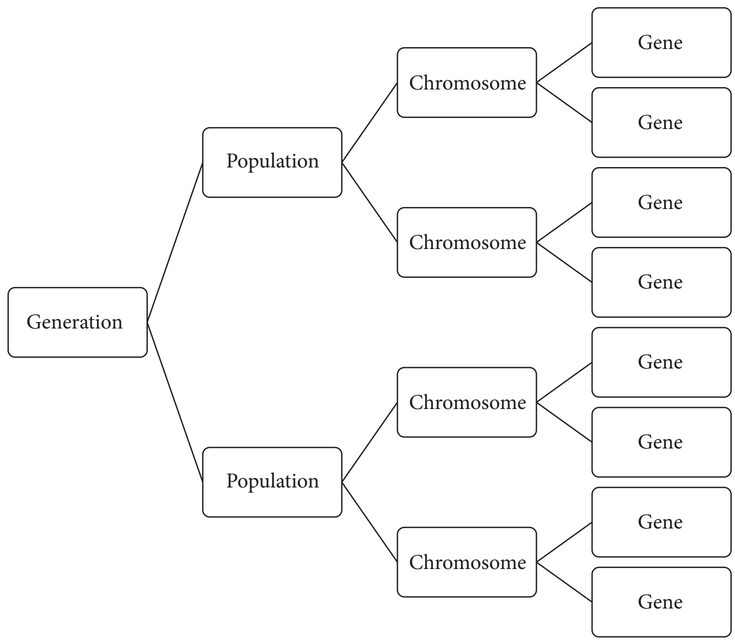

Figure 5: Basic elements of GA.

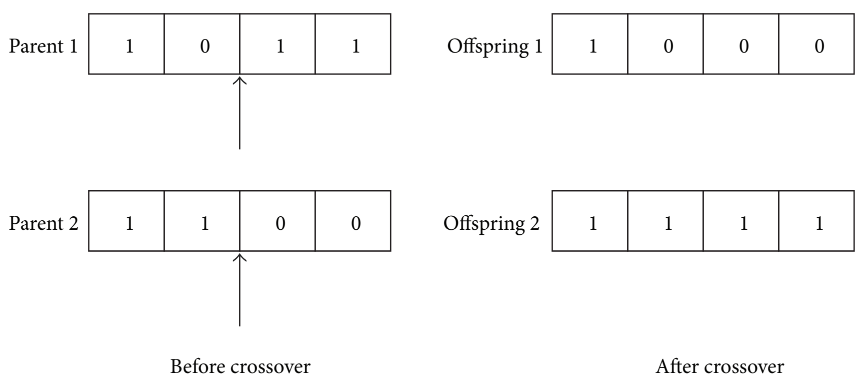

Figure 6: A typical crossover.

3.2.2. Crossover. The crossover operator is applied on the genes of two parent chromosomes to produce two offsprings which contain distinctiveness of the parent chromosomes. These offsprings have more probability of survival than their parents as they are fit as compared to their parents. Consider two parent chromosomes having 4 genes each. Crossover can be applied to these chromosomes at third gene (at pointed arrows) to obtain two offsprings as presented in Figure 6. This technique is known as single-point crossover. Many modified crossover techniques have been proposed in literature which will be identified in this review.

3.2.3. Mutation. Mutation operator is applied on a single chromosome for the purpose of changing a gene at its respective location. The gene 1011 can be mutated as 1111, as the gene at location 2 is flipped from 0 to 1. The mutation operator is used to change some information in a selected chromosome or diversify the solution space for further exploration. Many modified mutation techniques have been proposed in literature which will be identified in this review.

3.3. A Simple GA. First of all, the problem is coded in such a way that it can be represented in the form of binary numbers in a chromosome. As GA requires an initial candidate solution for its initiation, the initial solution is generated by randomization or diversification. The solution is then subjected to genetic operators (selection, crossover, and mutation) until the termination criteria are met. Algorithm 1 represents a typical GA.

3.4. General Approaches for FJSSP Solution Using GA. Keeping in view the combinatorial nature of the FJSSP, evolutionary algorithms have proven to be highly effective in providing acceptable solutions. Mesghouni et al. [222] were the first to use GA for the solution of FJSSP by proposing parallel job and parallel machine representation. In literature, the approaches for FJSSP solution can be classified as follows:

(i) Hierarchical approach: this approach aims to solve the FJSSP by decomposing into two parts and solving them separately according to its structure, that is,

Start
Encode initial solutions in chromosomes
Randomly generate an initial population of chromosomes
Compute fitness for each chromosome in the population
Repeat the following until number of offsprings <= number of chromosomes,
(i) Select a pair of parent chromosomes using selection method
(ii) Crossover the selected pair with the crossover probability at randomly chosen point to form two offsprings
(iii) Mutate the offsprings with mutation probability at all locations
Obtain new set of chromosomes
Replace the current population with new population using replacement strategy
Compute fitness
Generate new population until the fitness criteria is met
End

Algorithm 1: A simple GA.

machine selection problem and operation sequencing problem. Examples include the classical work of Brandimarte [223].

(ii) Integrated approach: this approach solves the two subproblems of FJSSP simultaneously instead of dealing with them in a separate way. Examples include state-of-the-art works of Dauzère-Pérès and Paulli [224], Hurink et al. [225], and Mastrolilli and Gambardella [226].

The scheduling problem cannot be solved without efficient solution aids due its difficult nature. Therefore, scheduling modules/systems have been designed to handle the problem. These types of systems help in performing experiments and also prove very helpful in debugging and validation of the scheduling algorithm. A modular and schematic representation of such scheduling system architecture with GA is presented in Figure 7.

4. FJSSPs Involving GA

Many different approaches have been applied to solve the problem due to its difficult nature. Some of the very recent approaches include biogeography based optimization [227], firefly algorithm [228], heuristics [229], invasive weed optimization [230], and differential evolution [231]. However, GA remains the most used algorithm for the FJSSP [3, 232]. This section presents the literature survey of the FJSSP solved using GA. First the methodology and scope are defined and then the literature survey is presented in following three areas:

(i) FJSSP solved using only GA and NSGA
(ii) FJSSP solved using advanced forms of GA
(iii) FJSSP solved using hGA.

4.1. Methodology and Scope. For the purpose of literature review, databases of Elsevier, Springer, Taylor and Francis, IEEE, and Hindawi are searched with the phrases "Flexible Job Shop Scheduling" and "Genetic Algorithm". Both conference and journal papers have been reviewed; however, emphasis has been laid on the journal publications. Book sections, thesis, and technical reports have not been included. The publications occurring after 2001 have been considered in this review. Data has been collected manually from selected publications using EndNote®.

4.2. Available Reviews. JSSP is a classical optimization problem, so the reviews of this problem can be traced back to 1966 [214]. However, the review papers aiming at the survey of FJSSP have appeared after 2000. Some of the salient features of reviews are outlined below.

(i) Gen and Lin [233] have presented the survey of multiobjective evolutionary algorithms for JSSP. They have reviewed FJSSP in this paper along with other shop layouts and identified various evolutionary strategies for achieving the solution of the said problem.

(ii) Vincent and Durai [234] have presented a survey of optimization techniques for multiobjective FJSSP. They have compared five algorithms and their performance results have been summarized.

(iii) Çaliş and Bulkan [3] have reviewed the artificial intelligence based approaches for JSSP. They have also included some instances of FJSSP in their survey.

(iv) Chaudhry and Khan [232] have presented a survey on all available solution strategies for FJSSP. They have segregated the literature based upon the solution techniques and provided insight to the research directions in FJSSP.

(v) Genova et al. [210] have also presented the solution approaches for multiobjective FJSSP.

It can be concluded from the data presented above that there is a need to assess the application and implementation of GA based approaches as they have not been addressed in a separate manner.

4.3. Objective Functions of FJSSP. The aim of solving the FJSSP is to satisfy a predefined performance criterion in order to obtain an optimal schedule. Therefore the FJSSP

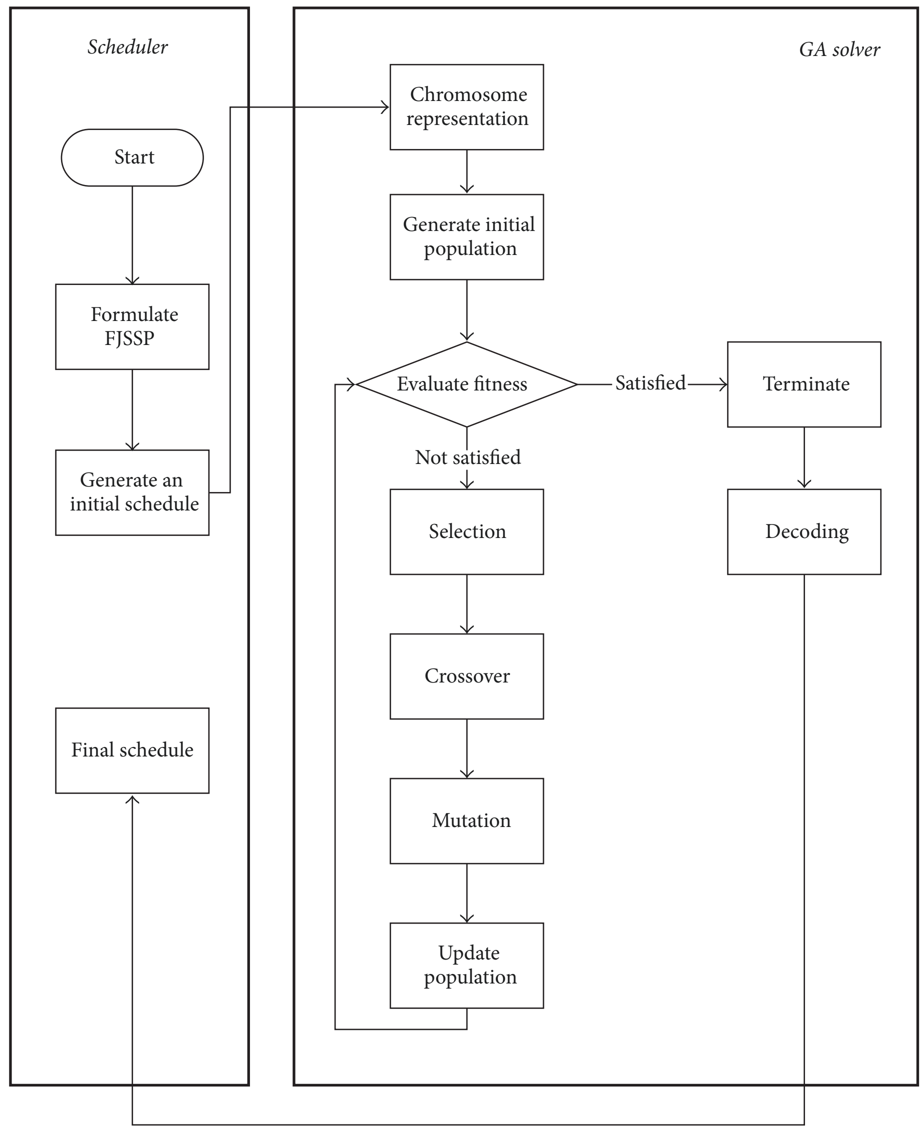

Figure 7: An architecture for FJSSP scheduler.

is essentially an optimization problem with a cost function which is required to be either minimized or maximized. Several optimization criteria have been formulated as a result and researchers have carried out single objective and multiobjective optimization with these criteria.

Table 1 presents a summary of commonly used objective functions in FJSSP along with their impact and applicability with respect to the production environment. Obviously, this list is not exhaustive and many other objective functions can be found in the literature.

4.4. Benchmark Problems. A number of benchmark problems have been formulated for FJSSP in order to compare the performance of new scheduling algorithms. The validation of a newly developed scheduling algorithm is done by the stated comparison. Various benchmark problems/data sets for FJSSP have been published. However this article reviews the benchmark data published by Fisher and Thompson [235], Lawrence [236], Tillard [237], Brandimarte [223], Hurink et al. [225], Lee and DiCesare [238], Barnes and Chambers [17], Dauzère-Pérès and Paulli [224], Kacem et al. [19, 133], and Fattahi et al. [239]. A detailed benchmark instances data has been presented by Dennis and Geiger [240].

4.5. FJSSP with GA and NSGA. GA has been used for solution of JSSP for above thirty years now; for example, Lawrence [241] has used GA for the solution of JSSP in 1985. However, the implementations of GA in FJSSP started after 1990 when Brucker and Schlie [16] presented their study in this area. Since then, there has been an immense increase in the research interest in this area. Table 2 presents the year-wise literature review. The single objective functions solved using GA have been included. Furthermore, the algorithms for multiobjective optimization are also included in this section.

Table 1: Commonly used FJSSP objective functions.

Measure	Symbol	Formula	Meaning	Impact/applicability
Makespan	$C_{\max}$	$\max_{1\le j\le n} C_j$	The time taken to complete all jobs	Minimizing makespan will directly minimize the production cost
Mean completion time	$\overline{C}$	$\frac{\sum_{j=1}^{n} C_j}{n}$	Average time required for completion of a single job	Minimizing this will directly reduce the production cost
Maximum Flowtime	F_j	$\max_{1\le j\le n} F_j$	The time that a job j spends in a shop while the processing takes place or while waiting	The longer the time a job spends on the production floor, the bigger its cost
Total tardiness	T	$\sum_{j=1}^{n} T_j$	The positive difference between the completion time and due date of all jobs	Applicable when early jobs do not give a reward but late jobs are penalized
Average tardiness	$\overline{T}$	$\frac{\sum_{j=1}^{n} T_j}{n}$	Average difference between the completion time and due date of a single job	Applicable when overall production is required to be completed in a stipulated time
Total weighted tardiness	T_{wt}	$\sum_{i=1}^{n} \alpha_i T_i$	Sum of weighted difference between the completion time and due date of a job	Applicable when some jobs are more important than others
Maximum lateness	$L_{\max}$	$\max_{1\le j\le n} L_j$	The maximum slack of a job with respect to its due date	Applicable when early jobs give a reward
Number of tardy jobs	n_T	$\sum_{j=1}^{n} U_j$	Number of jobs that are late	Directly affects the production cost and machine availability
Total workload of machines	W_T	$\sum_{j=1}^{n} W_j$	The total working time on all machines	Ensures maximum utilization of machines

These problems are primarily solved with Nondominated Sorting Genetic Algorithm (NSGA) and similar approaches.

4.6. FJSSP with Advanced Forms of GA. With the advancement in computing power and artificial intelligence techniques, various advances have been made in the original GA by incorporation of innovative ideas, majorly learning based evolution. Table 3 presents the year-wise literature in this area.

4.7. FJSSP with hGA. Although better results have been obtained with the techniques presented in Section 4.6, other standalone optimization techniques have also been proposed for the solution of FJSSP. However, researchers have amalgamated some standalone techniques with GA to obtain better solution times and results. These techniques have primarily been used to further improve the solution of a stated GA iteration before starting the new iteration. In this way, optimum solution is reached in a more effective manner. Table 4 presents year-wise literature in this class.

5. Analysis and Discussion

As obvious from the data presented in Section 4, FJSSP is an important research area which is highly published and which has been attended to with continuity over the last twenty years. This is due to the fact that the exact solution of this optimization problem has not been found yet and efforts are still being made to attain good solutions in a reasonable time and with reasonable computational resources.

We have reviewed a total of 190 research articles published from 2001 to December, 2017. These articles were narrowed down from a total of 384 articles found on the FJSSP. The articles have strictly been selected if they are on optimization of FJSSP and solved using a variant of GA. Furthermore, data also depicts the use of various types of GA operators (crossover, mutation, and selection) used by the researchers. The following facts have been revealed by this survey.

5.1. Source-Wise Distribution. Source-wise distribution of this survey is presented in Table 5. We have emphasized the number of journal articles over conference publications. It is evident from Figure 8 presenting the patch-wise distribution that 41% articles have been collected from 2009 to 2012 while 38% articles have been collected from 2013 to 2017. The combined percentage of articles published during years 2009–2017 comes out to be 79% of the total published research. Thus, a major chunk has been published in the last seven years.

5.2. Year-Wise Distribution. Year-wise distribution of these articles (journal and conference) is presented in Figure 9. There has been an increasing trend in the publications in this area from 2009 to 2012 while a constant and healthy trend has been observed in years 2013–2017.

5.3. Most Published Journals. The journals covering the subject of FJSSP are presented in Table 6. A total of 113 journals have given coverage to FJSSP related articles, while the journals publishing more than 2 papers are presented here. The International Journal of Production Research has published most research articles in this area.

Table 2: FJSSP with GA and NSGA.

Ref	Year	Article type	Algorithm details	Objective	GA parameters Mutation	Crossover	Selection	Benchmark
[24]	2001	C	GA	$C_{\max}$, W_M, W_T	Genetic mutation	Sequencing crossover, sequencing and ACX	-	KA
[25]	2001	J	GA	T	Operator for routing selection, operator for machine selection, operator for operation processing sequence	Operators for routing selection, operators for machine selection, operators for operations processing sequence	Fittest	Other
[26]	2002	C	NSGA II	$C_{\max}$, W_M, W_T	Directed mutation	OPX	Elitism	Other
[27]	2003	J	GA	T	Bit mutation	TPX	Fittest	LA
[28]	2003	C	GA	$C_{\max}$, W_M, W_T	AssM, IM	ACX, POX	Fittest	KA
[29]	2004	C	GA	$C_{\max}$	Two-part mutation	TPX	-	KA
[30]	2005	CP	GA	$C_{\max}$, T_{wq}	Genes pair mutation, specific gene mutation	Dominant gene crossover	Elitism	LD, other
[31]	2005	C	GA	$C_{\max}$	Reverse mutation	SPX, TPX, SXX	NA	KA
[32]	2006	C	GA	$C_{\max}$, W_M, W_T	PPS	POX	Elitism	KA
[33]	2008	J	GA	$C_{\max}$	AssM, reordering mutation	POX, ACX	Roulette wheel	FH
[34]	2008	J	GA	$C_{\max}$	PPS, AssM, assignment IM	POX, ACX	Binary tournament, linear ranking	BR, DP, BC, HU
[35]	2008	C	GA	$C_{\max}$	Random	Modified OPX	Random	Other
[36]	2008	C	GA	$C_{\max}$	Two-point EM	Linear order crossover	Roulette wheel, elitism	Other
[37]	2009	J	GA	$C_{\max}$	SM	Edge crossover	Roulette wheel	BR
[38]	2009	C	GA	$C_{\max}$	PPS	POX	Roulette wheel	Other
[39]	2009	J	GA	S_T, J_w	EM	SPX, TPX	-	Industry
[40]	2009	C	GA	$C_{\max}$, W_T	Weak link effect based mutation	Select mechanism crossover	-	KA
[41]	2009	C	GA	$C_{\max}$, W_M, W_T	SM	Linear order crossover	Roulette wheel	KA
[42]	2010	J	GA	$C_{\max}$	Local mutation, global mutation	TPX	Linear ranking	FT, HU, LA
[43]	2010	J	GA	$E_{\min}$, S_s	Random	POX, ACX	Tournament selection	Other

Table 2: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA parameters: Mutation	GA parameters: Crossover	GA parameters: Selection	Benchmark
[44]	2010	C	GA	$C_{\max}$, W_M, W_T	Individual mutation, AllM, SM	TPX	Random	KA
[45]	2010	C	GA	Max average agreement index, Fuzzy $C_{\max}$, W_T	InsM, replace mutation	Precedence operation crossover (POX), multipoint crossover	Elitism, tournament selection	KA
[46]	2010	J	GA	$C_{\max}$, minimum total load of machines, minimize the maximum load of machines	SM	MPPX, MGOX, MGPMX1, MGPMX2	Tournament selection	KA
[47]	2010	C	GA	$C_{\max}$	Random	TPX	Tournament selection	LA
[48]	2010	C	GA	$C_{\max}$, W_M, W_T	Random	TPX	-	KA
[49]	2010	J	GA	$C_{\max}$	Random	TPX	Elitism	KA
[50]	2011	J	GA	$C_{\max}$	MBM, modified PBM	Modified POX	Roulette wheel, tournament	KA, BR
[51]	2011	J	NSGA	$C_{\max}$, W_M, W_T, T	Preference mutation, machine mutation	Preference crossover, machine crossover	-	KA
[52]	2011	C	GA	$C_{\max}$, C_p	SM	MPPX, MGOX, MGPMX1, MGPMX2	Tournament	KA
[53]	2011	J	GA	$C_{\max}$	Random	PMX	-	Industry
[54]	2011	C	GA	$C_{\max}$	Random	Modified crossover	Binary tournament selection	LA
[55]	2011	J	NSGA-II	$C_{\max}$, min of the system unavailability	Random	TPX, RX	Roulette wheel, tournament selection	BC, BR, DP, HU
[56]	2011	J	NSGA-II	$C_{\max}$, W_M, W_T	Frame shift mutation, translocation mutation, inversion mutation	Uniform order-based crossover, precedence preservative crossover	Tournament selection	FT
[57]	2011	C	GA	$C_{\max}$, mean tardiness, mean flow time	Reciprocal EM	TPX	Elitism	Other
[58]	2011	C	NSGA	$C_{\max}$, W_M, W_T	Self-adaptive mutation	Precedence crossover, machine crossover	Niche selection	Other
[59]	2011	J	GA	$C_{\max}$	Random	TPX, UX, POX	Roulette wheel	BR, BC, DP
[60]	2012	C	GA	$C_{\max}$	Inverted mutation, random	TPX, random	-	Other
[61]	2012	C	GA	$C_{\max}$	SM	TPX	Linear ranking	KA
[62]	2012	J	GA	$C_{\max}$, W_M, W_T	SM, EM	TPX, POX	Random selection	FT
[63]	2012	J	NSGA-II, NRGA, MOGA, PAES	$C_{\max}$, C_p	SM, reversion mutation, InsM	OPX	Tournament selection, roulette wheel	BR, other

Table 2: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA parameters: Mutation	Crossover	Selection	Benchmark
[64]	2012	J	GA	$C_{\max}$	AssM, reordering mutation	POX, ACX	Roulette wheel	Other
[65]	2012	J	GA	$C_{\max}$	Modified SM	Hierarchical clustering based crossover	Fuzzy roulette wheel selection	BR
[66]	2012	J	GA	Max of total profit	SM	SPX	Tournament selection	Industry
[67]	2013	C	NSGA-II, SPEA-2	$C_{\max},\ E,\ T$	-	-	-	HU, KA, other
[68]	2013	CP	GA	$C_{\max}$	SM, random	OX, UX	-	KA
[69]	2013	J	NSGA, NRGA	$C_{\max},\ W_M,\ W_T$	Random, SM	IPOX, multipoint preservative crossover	Binary tournament	DP, BR
[70]	2013	J	GA	$C_{\max}$	SM	TPX	-	BR
[71]	2013	C	GA	$C_{\max}$	Modified mutation	Modified crossover	Roulette wheel	Industry
[72]	2014	C	GA	$C_{\max}$	Scramble mutation	Active schedule constructive crossover, GOX	High low fit selection	FT
[73]	2014	J	GA	$C_{\max}$	SM	TPX, POX	Tournament selection	FH, BC
[74]	2014	J	GA	$C_{\max},\ W_M$	-	-	Tournament	BR
[75]	2014	CP	GA	$C_{\max}$	SM	TPX	Roulette wheel	BR
[76]	2014	CP	GA	$C_{\max}$	SM, IM	POX, UX	Elitism	BR
[77]	2014	J	GA	T	SM	UX	Tournament selection	Other
[78]	2014	C	NSGA-II	$C_{\max}$, total production energy costs, total energy costs of maintenance	InsM, SM	SPX, MPX	-	Other
[79]	2014	J	GA	$C_{\max}$	SM	TPX, POX	Roulette wheel	Industry
[80]	2014	J	GA	Min of due date mean squared deviation	Shift mutation	TPX	-	Other
[81]	2015	J	GA	$C_{\max}$	SM	Position based crossover, OX, PMX	Roulette wheel, tournament selection	LD
[82]	2015	J	GA	$C_{\max}$	Random selection, neighborhood search	TPX, UX	Roulette wheel	BR
[83]	2015	J	GA	$\overline{T}$	Shift mutation, EM	TPX, POX	-	BR
[84]	2015	J	GA	$C_{\max}$	Values mutation	UX, POX	Roulette wheel	BR
[85]	2015	J	GA	$C_{\max}$	Inversion mutation, random	UX, POX		BR
[86]	2015	CP	GA	$C_{\max}$	SM, inversion mutation	TPX, POX	Tournament selection	KA
[87]	2015	J	GA	$C_{\max}$	SM	OPX		Other
[88]	2015	J	GA	$C_{\max}$	SM	TPX	Tournament selection	Other
[89]	2015	J	GA	$C_{\max}$	EM	TPX	Linear ranking	Other
[90]	2015	CP	GA	$C_{\max}$	Random	Integer crossover	Roulette wheel	KA
[91]	2016	J	NSGA-II, NRGA	$C_{\max}$ and stability objectives	Modified PBM, MBM	POX	-	KA, BR

Table 2: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA parameters Mutation	Crossover	Selection	Benchmark
[92]	2016	J	NSGA-II	C_{max}, total energy consumption	Deviation-based mutation, reciprocal EM	Intermediate recombination, line recombination	Elitism	Other
[93]	2017	CP	GA	C_{max}	RM, SM	POX, MPX	Tournament, elite reservation	Other
[94]	2017	J	GA	C_{max}	RM	MPX	Ranking, stochastic universal sampling	Other
[95]	2017	CP	GA	C_{max}, lead time	RM	MPX	Roulette wheel	Other
[96]	2017	J	GA	C_{max}	SM	POX	Elitism, population diversity strategy	Other
[97]	2017	CP	GA	C_{max}	RM	MPX	Roulette wheel	Industry
[98]	2017	CP	GA	C_{max}	RM, SM	POX	Tournament selection	BC
[99]	2017	J	GA	C_{max}	RM	TPX	Elitism	BR, other
[100]	2017	J	NSGA-II	C_{max}, W_M, W_T	RM, SM, reverse mutation, multipoint mutation	SPX, MPX, POX, JBX	Tournament	KA, DP, BR, BC
[101]	2017	J	GA	C_{max}, workload of each machine, W_T	-	-	-	KA
[102]	2017	CP	GA	C_{max}, W_T	-	POX, MPX	Elitism	Industry

Table 3: FJSSP with advanced forms of GA.

Ref	Year	Article type	Algorithm details	Objective	GA parameters Mutation	Crossover	Selection	Benchmark
[19]	2002	J	Controlled GA	$C_{\max}$, sum of machine workloads	Artificial mutation	Modified crossover	Elitism	KA
[103]	2005	J	Multistage operation based GA	$C_{\max}$, W_M, W_T	Local-search mutation	One-cut point crossover	Random	KA
[104]	2005	C	LEGA	$C_{\max}$	-	-	-	KA
[20]	2006	J	Iterative GA	$C_{\max}$	Random	SPX, job-based order crossover	Roulette wheel	FT, LA
[105]	2006	C	GA with Choquet integral	$C_{\max}$, W_M, W_T	-	-	-	KA
[106]	2007	J	Learning GA	$C_{\max}$	Random, algorithm based	TPX, random	Linear scaling, stochastic universal sampling	Mesghouni, BR
[107]	2008	C	TPGA	Min of fuzzy $C_{\max}$	Random	GOX, generalization of PPX	Tournament selection	KA
[108]	2008	J	GA with Choquet integral	$C_{\max}$, W_M, W_T, sum of weighted earliness and weighted tardiness, sum of production cost	-	-	-	KA
[109]	2009	C	Course grained parallel GA based on island model parallelization technique	$C_{\max}$	Sublot step mutation, Sublot swap mutation, random operation AssM, intelligent operations AssM, operations sequence shift mutation	SPX-, SPX-2 (SPC-2), operation to machine ACX, job level operations sequence crossover, subplot level operations sequence crossover	Tournament selection	Other
[110]	2009	C	Adaptive GA	$C_{\max}$	Adaptive	Adaptive, precedence operation crossover	Roulette wheel	KA
[111]	2010	C	GA with learning by injection of sequences	$C_{\max}$	Mutation by direct exchange, mutation by random exchange, mutation by inversion, Mutation by close exchange, mutation by gap of all the elements	SPX	Random	FT, other
[112]	2010	C	Cooperative coevolutionary GA	$C_{\max}$	Random, SM	Row crossover, column crossover, precedence order crossover	Roulette wheel	BR

Table 3: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA parameters Mutation	Crossover	Selection	Benchmark
[113]	2010	J	Parallel GA	$C_{\max}$	PPS, AssM, IM	ACX, POX	Tournament selection	Other
[114]	2010	C	Matrix coded GA	$C_{\max}$	Machine dimension mutation, operation dimension mutation	Machine dimension crossover, operation dimension crossover	Tournament	Other
[115]	2010	J	Decomposition integration GA	$C_{\max}$	Swap operator	Generalized position crossover, generalization of PPX	Tournament selection	KA
[116]	2010	J	LEGA	$C_{\max}$	Random, operational memory based mutation	TPX, random	Tournament	LA
[117]	2010	C	Adaptive GA	$C_{\max}$	Random	OPX, three-point crossover	League selection	Industry
[118]	2010	C	Coevolutionary genetic algorithm	Fuzzy $C_{\max}$	SM Random	TPX, discrete crossover	Tournament selection	KA
[119]	2010	C	GA based on immune and entropy principle	$C_{\max}$, W_T	Random	IPOX, MPX	-	KA
[120]	2011	J	GA with heuristics	$C_{\max}$	-	-	Elitism, tournament selection	BR, other
[121]	2011	C	Adaptive GA	$C_{\max}$	Random	OPX	Elitism	KA
[122]	2011	C	MSCEA	$C_{\max}$	Neighborhood mutation	TPX	Random	BR
[123]	2012	J	Multiobjective GA	$C_{\max}$, total machining time	SM	SPX	-	Other
[124]	2012	C	GA with learning	$C_{\max}$	Random	OPX	Elitist	LD, BR
[125]	2012	J	Coevolutionary GA	Fuzzy $C_{\max}$	SM	TPX, extension of PPX	Modified tournament selection	Other
[126]	2012	J	Jumping genes GA	$C_{\max}$, flow time of products with AGV, completion of the products	EM	SPX	Tournament selection	Other
[127]	2013	J	Real coded GA	$C_{\max}$	Random	Extended intermediate crossover, OPX	Roulette wheel, binary tournament, elitism, replacement	BR
[128]	2014	J	NSGA-II based on blood variation	$C_{\max}$, processing cost, energy consumption, cost-weighted processing quality	Modified mutation	Blood relation based crossover	Modified quick sorting ranking	Industry
[129]	2016	CP	Immune GA	Maximization of due time satisfaction, minimize the total processing costs	Random	SPX	Roulette wheel	FT
[130]	2016	J	Extended GA	Maximize satisfaction degree	SM	TPX, POX	Tournament selection	Other
[131]	2016	CP	GA with comprehensive search	$C_{\max}$	Random	Operation-based crossover	Roulette wheel	Other

Table 4: FJSSP with hGA.

Ref	Year	Article type	Algorithm details	Objective	GA Parameters Mutation	Crossover	Selection	Benchmark
[132]	2001	C	GA + MILP	$C_{\max}$, Min of T	SM	Multipoint crossover	Tournament selection	Other
[133]	2002	J	Controlled GA + fuzzy logic	$C_{\max}$, W_M, W_T	Artificial mutation	Modified crossover	Fittest	KA
[134]	2004	J	GA + priority dispatching rules	Min ratio of tardy jobs, variance of the flow time, amount of mold changes, max efficiency of machines	Priority dispatching rule based mutation	Two-point conical	Tournament selection, elitism	Other
[135]	2004	C	GA + LS + heuristic	$C_{\max}$, W_M, W_T	-	POX	Fittest	Other
[136]	2006	C	GA + TS	T	PPS, random	POX, GOX	-	Other
[137]	2006	J	GA + LS	$C_{\max}$, W_M, W_T	Phenotype based mutation	Phenotype based crossover	Ranking selection	KA
[138]	2006	J	GA + scheduling rules	$C_{\max}$	Random	SPX	Roulette wheel	FT
[139]	2006	C	GA + heuristic	$C_{\max}$	InsM, SM	OPX with priority list, TPX with priority list, selective machine sequences crossover	-	LA, FT
[140]	2006	C	GA + LS procedure based on shifting bottleneck	$C_{\max}$, W_M, W_T	AllM, ImmM	Exchange crossover, EOX	-	KA
[141]	2006	C	GA + TS	$C_{\max}$, maximum lateness	Random (dynamic)	Random (Boolean matrix)	-	LA
[142]	2006	C	GA + heuristic	T	Random sequencing mutation, random AssM, IM	GOX, sequencing and ACX, ACX	-	BR, other
[143]	2007	J	GA + LS procedure based on shifting bottleneck	$C_{\max}$, W_M, W_T	AllM	EOX	-	KA, BR
[144]	2007	J	GA + LS	$C_{\max}$, W_M, W_T	PPS	Preserving order based crossover	Fittest	BR, DP,
[145]	2008	J	GA + variable neighborhood descent	$C_{\max}$, W_M, W_T	AllM, ImmM	EOX, UX	Ranking selection	KA, BR, BC, DP, FT, LA, other

TABLE 4: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA Parameters Mutation	Crossover	Selection	Benchmark
[146]	2008	J	GA + guided LS	$C_{\max}$, W_M, W_T	Random	TPX	Ranking selection	KA, other
[147]	2008	J	GA + MILP	Mean flow time, $C_{\max}$, maximum lateness, total absolute deviation from the due dates	Random	OPX, partial matched crossover	Elitist	Other
[148]	2008	C	GA + TS	$C_{\max}$	Random	TPX, improved POX	Tournament selection	BR
[149]	2009	J	GA + simulation	$C_{\max}$, min of mean tardiness	AssM, sequencing mutation	TPX	Roulette wheel	BR
[150]	2009	C	GA + LS	$C_{\max}$	-	-	-	BR
[151]	2010	J	NSGAII + SA	$C_{\max}$, C_p	Reciprocal swap	UX	Elitism	Other
[152]	2010	J	GA + immune mechanism + SA	$C_{\max}$, C_p	Adaptive crossover	Adaptive crossover	Fittest	Other
[153]	2010	J	VNS + GA	$C_{\max}$, W_M, W_T	Random	TPX	Fittest	KA
[154]	2010	C	GA + chaotic LS	$C_{\max}$	IM, random	GOX, generalized PMX	Binary tournament	BR
[155]	2010	J	GA + hill climbing	$C_{\max}$, W_M, W_T	Random	TPX	Tournament	KA
[156]	2010	J	GA + immune + entropy principle	$C_{\max}$, W_M, W_T	Random	IPOX, multipoint preservative crossover	Tournament selection	KA, BR, DP
[157]	2010	C	PSO + GA	$C_{\max}$	Random	SPX	-	Other
[158]	2010	C	GA + VNS	$C_{\max}$, W_M, W_T	Random, swap	TPX, POX	Tournament selection	KA
[159]	2010	C	GA + TS	$C_{\max}$, W_M, W_T	Random	TPX, POX	-	KA
[160]	2011	J	GA + LS	$C_{\max}$	PBM, MBM	POX	Roulette wheel, ranking	KA, Mesghouni, LD, BR, BC, DP, HU
[161]	2011	C	TS + SA + GA	$C_{\max}$, W_M, W_T	Random	Combined order and position-based crossover	-	KA, BR
[162]	2011	C	GA + AIS	$C_{\max}$	SM	MPPX, MGOX, MGPMX1, MGPMX2	Elitism	Other
[163]	2011	J	GA	Min of maximum workload	SM	UX	Search rate survival	BR, LA
[164]	2011	J	GA + SA	$C_{\max}$, W_T	Dynamic mutation	Dynamic crossover	Roulette wheel	KA
[165]	2011	C	GA + TS	Min time, min cost, equipment utilization rate	Random	MPPX, MGOX, MGPMX1, MGPMX2	Elitism	Other

Table 4: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA Parameters Mutation	Crossover	Selection	Benchmark
[166]	2011	C	GA + PSO	C_{max}	Random, balance load mutate	POX, MPX	-	KA, BR
[167]	2011	J	GA + fuzzy set theory	Optimization of cost, quality and time	Neighborhood mutation	Neighborhood crossover	-	Industry
[168]	2011	J	GA + ACO	C_{max}	Inverted mutation, operation assignment machine knowledge	TPX, modified crossover	Linear scaling, stochastic universal sampling	KA, BR
[169]	2012	J	GA + grouping GA	T, total machine idle time, C_{max}	-	-	-	Industry
[170]	2012	C	GA + LS	C_{max}	Machine replacement	POX	Elitism	Industry
[171]	2012	J	GA + Petri nets	C_{max}, total expense, workload of machines	InvM	-	Elitism	Other
[172]	2012	C	GA + LS + TS	C_{max}, W_M, W_T	SM, random	UX, IPOX	Elitism	KA
[173]	2012	C	hGA	Min the total earliness, min of tardiness penalties	SM, SA	POX, job-based machine crossover	Roulette wheel	FT
[174]	2012	J	GA + TS	C_{max}, min of mean flow time	AllM	PMX, OX	Tournament selection	Other
[13]	2012	C	GA + PSO	T	-	OPX	Roulette wheel	KA
[175]	2012	C	GA + TS + modified shifting bottleneck procedure	C_{max}	SM	SPX	Elitism	Other
[176]	2012	J	Duplicate GA + LS	C_{max}, min of total idleness	SM	UX	Roulette wheel	KA
[177]	2012	J	GA + LS based on critical path theory	C_{max}, W_M, W_T	ImmM, modified AssM	POX, TPX	-	KA, BR
[178]	2012	J	GA + TS	C_{max}	AssM	SPX	Roulette wheel	Other
[179]	2013	J	GA + VNS with affinity function	C_{max}	SM	UX, OPX	Tournament selection	Other
[180]	2013	J	GA + simulation	Total of average flow times	SM	Two-stage crossover	Tournament	Other
[181]	2013	J	GA + SA	Min the total cost including delay costs, setup costs, and holding costs	Intelligent AssM, random AssM, intelligent sequencing mutation 1, intelligent sequencing mutation 2, randomly sequencing mutation	POX, random crossover	Linear ranking	Other

Table 4: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA Parameters Mutation	Crossover	Selection	Benchmark
[182]	2013	C	PSO + GA with Cauchy distribution	$C_{\max}$	InsM	MPX	-	Other
[183]	2013	J	GA + SA	$C_{\max}$, W_M, W_T	New mutation	New crossover	Elitism	KA
[179]	2013	J	GA + VNS	$C_{\max}$	InsM, SM	TPX, modified crossover	Tournament selection	Other
[184]	2013	J	NGSA + knowledge based Algorithm	$C_{\max}$, robustness	MBM, ImmM	TPX	-	KA, other
[185]	2013	C	NSGA-II + LS	$C_{\max}$, W_M, W_T	Random	Modified crossover	-	KA, BR
[186]	2013	J	GA + SA	$C_{\max}$, sum of std deviation of processing workload for all working centers	InvM	OX	Ranking selection	Other
[187]	2014	J	GA + shifting bottleneck	$C_{\max}$, W_M	SM	EOX	Elitism	Other
[188]	2014	J	GA + population improvement	$C_{\max}$	SM	POX, MPX	Binary tournament selection	BR
[189]	2014	CP	GA + TS	$C_{\max}$	SM, alternative mutation	POX, PMX	Tournament selection	KA
[190]	2014	CP	GA + LS	$C_{\max}$	Random	POX	-	BR, HU, DP
[191]	2015	CP	GA + TS	$C_{\max}$	IM	OX	Elitism	BR, HU
[192]	2015	J	Neighborhood-based GA + TS + LS	$C_{\max}$	SM, InvM	UX, IPOX	Fitness neighborhood selection operator	BR, HU
[193]	2015	J	GA + TS	$C_{\max}$	-	Job order crossover	Tournament selection	BR, BC, DP, Other
[194]	2015	J	GA + PSO	Minimize sum of holding, setup, production, overtime costs	SM	MPX	Tournament selection	Other
[195]	2015	J	GA + heuristics	$C_{\max}$, overtime costs of machines	SM	OX	Ranking	Other
[196]	2015	J	GA + VNS	T	AssM, SM	UX, modified POX	Linear ranking	BR, HU, other
[197]	2016	J	GA + heuristics	Mean tardiness	SM	SPX	-	Other
[198]	2016	J	GA + TS	$C_{\max}$	SM	PBX	Tournament	FT, LA
[199]	2016	J	GA + TS	$C_{\max}$	SM, neighborhood mutation	POX, JBX, TPX	Elitism, tournament selection	KA, FH, BR, BC, HU, DP
[200]	2016	J	GA + TS	Weighted tardiness, balancing the setup workers load, min the work-in-process	SM	TPX	Ranking	Other
[201]	2016	J	Neighborhood GA + TS	$C_{\max}$	SM, InsM	UX, IPOX	Fitness neighborhood selection operator	Other
[202]	2016	CP	GA + LS	$C_{\max}$	Uniform mutation, InsM, SM	UX, TPX, POX	Average hamming distance	KA, BR

Table 4: Continued.

Ref	Year	Article type	Algorithm details	Objective	GA Parameters Mutation	Crossover	Selection	Benchmark
[203]	2016	J	GA + LS	$C_{\max}$	SM	JBX	Elitism, tournament	BR
[204]	2017	J	GA + SA	$C_{\max}$, maximizing the total availability of the system, minimizing total energy cost of both production and maintenance operations	RM, SM	UX, POX	Roulette wheel	Other
[205]	2017	CP	GA + VNS	$C_{\max}$	IntM	MPX	Elitism	HU
[206]	2017	J	GA + Taguchi	$C_{\max}$	RM, IntM	TPX, POX, UX	Tournament, other	BR, other
[207]	2017	J	GA + VNS	$C_{\max}$, mean tardiness	RM	POX	Tournament selection	HU
[208]	2017	J	GA + LS	$C_{\max}$	SM, RM	POX	Fitness-neighborhood selection	KA, BR, HU, BC

Table 5: Distribution of article types.

Article type	Quantity
Journal article	108
Conference paper	64
Conference proceedings	18
Total	190

Table 6: Paper distribution in journals.

Journal name	Number of publications
International Journal of Production Research	13
Computers & Operations Research	6
International Journal of Advanced Manufacturing Technology	5
Expert Systems with Applications	5
Journal of Intelligent Manufacturing	5
Computers & Industrial Engineering	4
International Journal of Production Economics	3

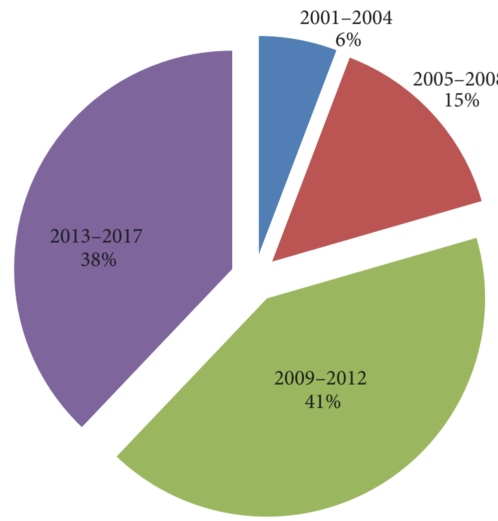

Figure 8: Percentage of publications in time—patches.

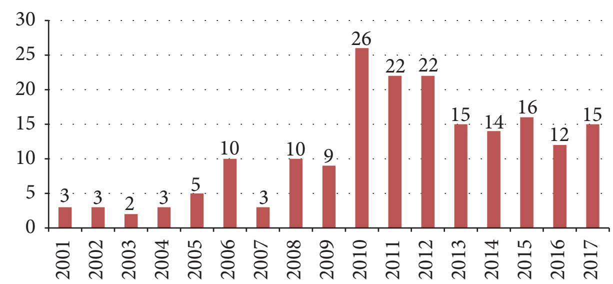

Figure 9: Year-wise distribution of research.

5.4. Country-Wise Publication Data. Figure 10 presents country-wise publication data. A total of 184 countries have contributed in this area, out of which China has published 43.53% of publications while Iran, France, and Japan have published 11.18%, 10.59%, and 7.06% publications, respectively. Other notable countries are India, Turkey, and Taiwan.

5.5. Techniques Used for FJSSP. There are 78 different technique combinations used in the selected papers, out of which only 10 techniques constitute 119 papers (62.63%). A distribution of techniques having at least 3 publications is presented in Table 7. It is evident that 70 publications have used GA as a sole technique for solution of FJSSP and GA + TS is the most used hybrid technique. A group-wise division of the whole techniques in Table 8 shows that hybrid techniques constitute a 37.5% of our study, while pure GA based publications amount to 39.5%. It is also evident that GA has been hybridized majorly with local search approaches like TS, SA, and VNS. This technique improves the initial solution of GA routine. There is a need to explore the possibility of hybridizing various other standalone optimization algorithms with GA.

Table 7: Algorithms used for FJSSP having at least 3-publication count.

Algorithm details	Publication count	Percentage
GA	70	54.69
GA + TS	13	10.16
NSGA-II	11	8.59
GA + local search	10	7.81
GA + heuristic	5	3.91
GA + SA	5	3.91
GA + VNS	5	3.91
Adaptive GA	3	2.34
GA + PSO	3	2.34
GA with learning	3	2.34

Table 8: Group-wise publication count.

Group	Publication count	Percentage
GA	79	39.5
Hybrid	75	37.5
Advanced GA	31	15.5
NSGA	15	7.5

5.6. Most Used Objective Functions. A total of 62 objective functions have been optimized in single/multiobjective manner. Table 9 summarizes the occurrences and percentages of the objective functions giving at least 02 occurrences. It is evident that makespan is the most sought after objective. Figure 11 shows that 46 different multiobjective functions have been addressed in contrast with 13 different single objective functions.

Table 10 indicates that makespan has been addressed the most as a single objective function, while makespan, workload of most loaded machine, and total workload of machines have been addressed the most as a multiobjective function.

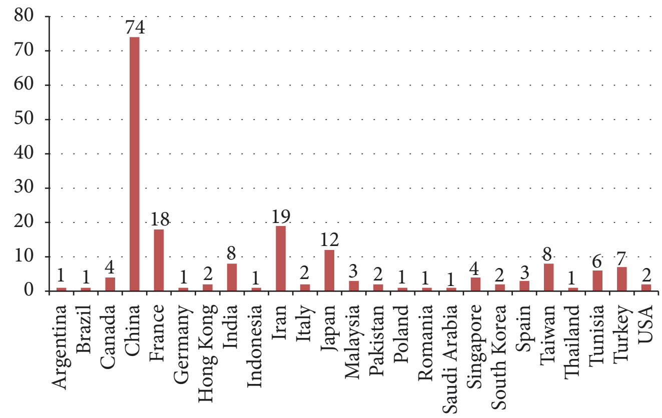

Figure 10: Country-wise publication data.

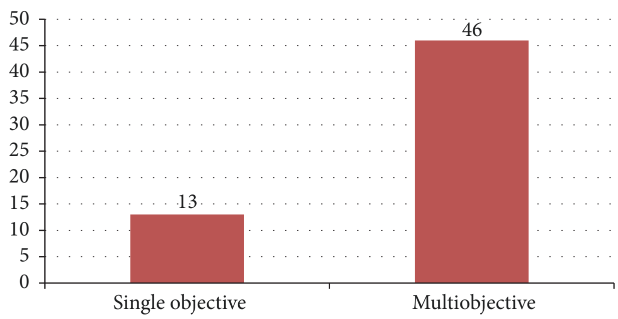

Figure 11: Division of different single and multiobjective cost functions.

Table 9: Most addressed objective functions.

Objective function	Number of occurrences	Percentage
Makespan	162	71.37
Total workload of machines	32	14.10
Workload of most loaded machine	14	6.17
Total tardiness	7	3.08
Max lateness	2	0.88
Mean flow time	2	0.88
Mean tardiness	2	0.88
Min of fuzzy makespan	2	0.88
Tardiness	2	0.88
Weighted tardiness	2	0.88

5.7. Most Benchmark Problems Attempted. The benchmark problems attempted the most are tabulated in Figure 12. The benchmark problems have been addressed 262 times. It is pertinent to mention here that there is a tendency in literature, especially conference papers, to attempt using the selected data sets. Thus if an author has attempted to solve only one of the ten problems of Brandimarte, we have counted it as one instance. The problems of Kacem have been attempted the most with Brandimarte at the 2nd priority. The industrial problems have been addressed only 5% of the time and other than that the research has been inclined towards algorithm development and comparison with benchmark instances.

Table 10: Occurrences of objective functions.

Objective functions	Nature	Occurrences
Makespan	Single	92
Makespan, workload of most loaded machine, total workload of machines	Multi	28
Total tardiness	Single	4
Makespan, production costs	Multi	4
Makespan, total workload of machines	Multi	4
Min of fuzzy makespan	Single	3

Table 11: Use of software tools for FJSSP solution.

Software tool	Times used
C++	32
MATLAB	28
JAVA	10
C#	7
Visual Basic	5
C	4
Visual C++	3

5.8. Software Tools Used. This survey shows that 25 software tools have been used for the competent solution of FJSSP. Table 11 depicts that C++ has been the most popular language for programming the problem, with MATLAB being the second most popular.

5.9. Special Cases of FJSSP. Although there are many different cases of FJSSP studied in literature, the following cases have

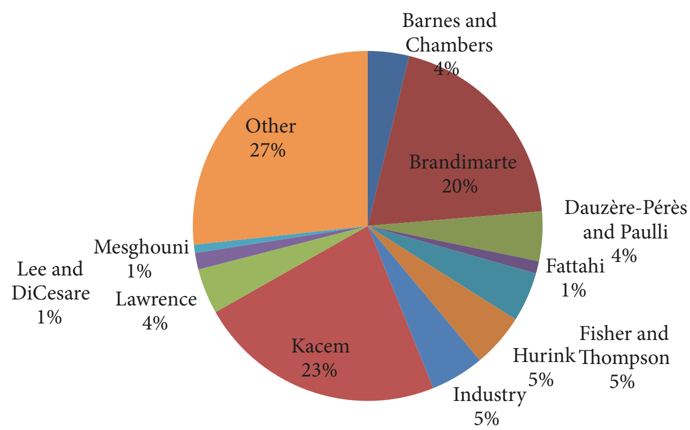

Figure 12: Distribution of benchmark problems attempted.

Table 12: Distribution of mutation schemes.

Mutation type	Times used
Swap mutation	60
Random	53
Inverse mutation	9
Assignment mutation	8
Insertion mutation	7
Precedence preserving shift mutation	7
Intelligent mutation	7
Allele mutation	5
Immigration mutation	4
Neighborhood mutation	4
Exchange mutation	3

received special attention as they have been studied more often than others.

(i) Dual-resource constrained FJSSP, for example, [52, 162]

(ii) Sequence dependent setup times, for example, [88, 207]

(iii) Distributed and flexible JSSP, for example, [79, 99]

(iv) Just-In-Time dynamic scheduling, for example, [80, 83]

(v) Overlapping in operations, for example, [73, 98]

(vi) Random machine breakdowns, for example, [91, 184]

(vii) Dynamic FJSSP, for example, [94, 96].

5.10. GA Parameters. It is evident from the literature review presented in Tables 2, 3, and 4 that various GA parameters have been used to address the FJSSP. Table 12 presents the major types of mutation and their frequency of use. Similarly, Tables 13 and 14 present the frequency of crossover and selection operators.

Table 13: Distribution of crossover schemes.

Crossover type	Times used
Two-point crossover	49
Precedence preserving order-based crossover	44
Uniform crossover	20
Single-point crossover	15
One-point crossover	13
Multipoint crossover	12
Assignment crossover	7
Modified crossover	7
Random	7
Generalized order crossover	5
Order crossover	5
Partially mapped crossover	5
Enhanced order crossover	4
Improved precedence preserving order-based crossover	5

Table 14: Distribution of selection schemes.

Selection type	Times used
Tournament	50
Roulette wheel	33
Elitism	30
Fittest	9
Linear ranking	6
Random	6
Ranking selection	8

6. Conclusions

This paper has presented the review of GA based techniques for solution of FJSSP with the help of literature published in the conference and journal papers in the time frame of 2001–2017. The study presents a comprehensive insight into the research trends in this area.

The contribution of this work is twofold. Firstly, it addresses the application of GA specifically to the FJSSP and provides a startup for researchers who want to excel in this area by providing recent research trends. Secondly, the parameters that have been used the most are also identified which can be mapped with references for advanced studies. Furthermore, the special cases of FJSSP have also been identified.

The study has surveyed the implementation of GA for FJSSP in detail and the trends for use of GA parameters have also been presented, along with the benchmark studies conducted with each approach. It is obvious that GA is the most popular technique for the solution of FJSSP. The researchers have made no claim that any of the methods is the best, but the trend is to compare the solutions with the standard benchmarks. The study has pointed out the mostly used parameters of GA in the literature. It was also observed that hybrid GA is even more popular than the pure GA. Furthermore, due to the known phenomena of local minima trap in GA routine, local search techniques have mostly been integrated with the GA. Consequently, there is a need to explore options for integration of more advanced artificial intelligence based algorithms with GA.

Notations

ACX:	Assignment crossover
AllM:	Allele mutation
AssM:	Assignment mutation
BC:	Barnes and Chambers
BR:	Brandimarte
C:	Conference paper
$C_{\max}$:	Makespan
CP:	Conference proceedings
C_p:	Production costs
DP:	Dauzère-Pérès and Paulli
E:	Earliness
EM:	Exchange mutation
$E_{\min}$:	Minimum of efficiency
EOX:	Enhanced order crossover
$\overline{F}$:	Mean flow time
FH:	Fattahi
FT:	Fisher and Thompson
Fuzzy $C_{\max}$:	Min of fuzzy makespan
GA:	Genetic Algorithm
GOX:	Generalized order crossover
HU:	Hurink
ImmM:	Immigration mutation
InsM:	Insertion mutation
IntM:	Intelligent mutation
InvM:	Inverse mutation
IPOX:	Improved precedence preserving order-based crossover
J:	Journal article
J_w:	Waiting time of jobs
KA:	Kacem
LA:	Lawrence
LD:	Lee and DiCesare
LEGA:	LEarnable Genetic Architecture
$L_{\max}$:	Max lateness
LS:	Local search
MBM:	Machine based mutation
MGOX:	Modified generalized order crossover
MGPMX1:	Modified generalized partially mapped crossover 1
MGPMX2:	Modified generalized partially mapped crossover 2
MOGA:	Multiobjective Genetic Algorithm
MPPX:	Modified precedence preserving crossover
MPX:	Multipoint crossover
MX:	Modified crossover
NM:	Neighborhood mutation
NRGA:	Nondominated ranked Genetic Algorithm
NSGA:	Nondominated sorting Genetic Algorithm
OPX:	One-point crossover
OX:	Order crossover
PAES:	Pareto archive evolutionary strategy
PBM:	Position based mutation
PMX:	Partially mapped crossover
POX:	Precedence preserving order-based crossover
PPS:	Precedence preserving shift mutation
PPX:	Precedence preserving crossover
PSO:	Particle swarm optimization
RM:	Random mutation
RX:	Random crossover
SA:	Simulated annealing
SM:	Swap mutation
SPEA:	Strength Pareto evolutionary algorithm
SPX:	Single-point crossover
S_s:	Stability of schedules
SSX:	Subsequence exchange crossover
T:	Total tardiness
$\overline{T}$:	Average tardiness
TI:	Tillard
TPGA:	Two-population Genetic Algorithm
TPX:	Two-point crossover
TS:	Tabu search
T_{wt}:	Weighted tardiness
UX:	Uniform crossover
VNS:	Variable neighborhood search
W_M:	Workload of most loaded machine
W_T:	Total workload of machines
MSCEA:	Multi-swarm collaborative evolutionary algorithm
MILP:	Mixed integer linear programming
ACO:	Ant colony optimization
hGA:	Hybrid Genetic Algorithm
JBX:	Job based crossover.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The authors acknowledge NUST for partially financing these studies.

References

[1] WorldBank, *Manufacturing, Value Added (% of GDP)*, 2016, http://data.worldbank.org/indicator/NV.IND.MANF.ZS/countries?display=graph.

[2] M. M. Andreasen and H. Lars, *Integrated Product Development*, IFS, Denmark, 2000.

[3] B. Çaliş and S. Bulkan, "A research survey: review of AI solution strategies of job shop scheduling problem," *Journal of Intelligent Manufacturing*, vol. 26, no. 5, pp. 961–973, 2015.

[4] H. Emmons and G. Vairaktarakis, *Flow Shop Scheduling: Theoretical Results, Algorithms, And Applications*, vol. 182, Springer Science & Business Media, 2012.

[5] J. Grobler, A. P. Engelbrecht, S. Kok, and S. Yadavalli, "Metaheuristics for the multi-objective FJSP with sequence-dependent set-up times, auxiliary resources and machine down time," *Annals of Operations Research*, vol. 180, no. 1, pp. 165–196, 2010.

[6] P. Cowling and M. Johansson, "Using real time information for effective dynamic scheduling," *European Journal of Operational Research*, vol. 139, no. 2, pp. 230–244, 2002.

[7] R. W. Conway, W. L. Maxwell, and L. W. Miller, *Theory of Scheduling*, Addison-Wesley, Reading, Mass, USA, 1967.

[8] S. Dauzère-Pères and J. B. Lasserre, *An integrated approach in production planning and scheduling*, vol. 411 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, Berlin, 1994.

[9] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, Springer, New York, NY, USA, 2012.

[10] K. R. Baker and D. Trietsch, *Principles of Sequencing and Scheduling*, John Wiley & Sons, New York, NY, USA, 2009.

[11] J. Blazewicz, K. H. Ecker, P. Erwin, G. Schmidt, and J. Weglarz, *Scheduling Computer and Manufacturing Processes*, Springer Science & Business Media, 2013.

[12] J. M. Framinan, R. Leisten, and R. R. García, "Manufacturing scheduling systems: An integrated view on models, methods and tools," *Manufacturing Scheduling Systems: An Integrated View on Models, Methods and Tools*, vol. 9781447162728, pp. 1–400, 2014.

[13] L. Lin, M. Gen, Y. Liang, and K. Ohno, "A hybrid EA for reactive flexible job-shop scheduling," in *Proceedings of the 2012 Complex Adaptive Systems Conference*, pp. 110–115, USA, November 2012.

[14] R. T. Nelson, *Shop Scheduling: An Application of Linear Programming*, Defense Technical Information Center, 1954.

[15] C. S. Chong, A. I. Sivakumar, M. Y. H. Low, and K. L. Gay, "A bee colony optimization algorithm to job shop scheduling," in *Proceedings of the 2006 Winter Simulation Conference, WSC*, pp. 1954–1961, USA, December 2006.

[16] P. Brucker and R. Schlie, "Job-shop scheduling with multi-purpose machines," *Computing*, vol. 45, no. 4, pp. 369–375, 1990.

[17] J. W. Barnes and J. B. Chambers, "Flexible Job Shop Scheduling by Tabu Search," in *Flexible Job Shop Scheduling by Tabu Search T.R.S. Graduate Program in Operations Research and Industrial Engineering, ORP96-09*, University of Texas at Austin, Tex, USA, 1996.

[18] N. M. Najid, S. Dauzere-Pérès, and A. Zaidat, "A modified simulated annealing method for flexible job shop scheduling problem," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 5, pp. 89–94, 2002.

[19] I. Kacem, S. Hammadi, and P. Borne, "Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 32, no. 1, pp. 1–13, 2002.

[20] F. T. S. Chan, T. C. Wong, and L. Y. Chan, "Flexible job-shop scheduling problem under resource constraints," *International Journal of Production Research*, vol. 44, no. 11, pp. 125–143, 2006.

[21] A. Jasbir, *Introduction to Optimum Design*, Academic Press, 2004.

[22] M. A. Bhatti, *Practical Optimization Methods: With Mathematica® Applications*, Springer Science & Business Media, 2012.

[23] P. Vilfredo, "Manuale di economia politica," *Societa Editrice*, vol. 13, 1906.

[24] I. Kacem, S. Hammadi, and P. Borne, "Approach by localization and genetic manipulation algorithm for flexible job-shop scheduling problem," in *Proceedings of the 2001 IEEE International Conference on Systems, Man and Cybernetics*, pp. 2599–2604, Ariz, USA, 2001.

[25] C. Zhao and Z. Wu, "A genetic algorithm approach to the scheduling of FMSs with multiple routes," *International Journal of Flexible Manufacturing Systems*, vol. 13, no. 1, pp. 71–88, 2001.

[26] T. Hsu, R. Dupas, D. Jolly, and G. Goncalves, "Evaluation of mutation heuristics for the solving of multiobjective flexible job shop by an evolutionary algorithm," in *Proceedings of the 2002 IEEE International Conference on Systems, Man and Cybernetics*, A. E. K. K. M. P. Borne, Ed., vol. 6, 2002.

[27] Y.-J. Jang, K.-D. Kim, S.-Y. Jang, and J. Park, "Flexible job shop scheduling with multi-level job structures," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 46, no. 1, pp. 33–38, 2003.

[28] I. Kacem, "Genetic algorithm for the flexible job-shop scheduling problem," in *Proceedings of the System Security and Assurance*, pp. 3464–3469, 2003.

[29] N. B. Ho and J. C. Tay, "GENACE: An efficient cultural algorithm for solving the flexible job-shop problem," in *Proceedings of the 2004 Congress on Evolutionary Computation, CEC2004*, vol. 2, pp. 1759–1766, 2004.

[30] F. T. S. Chan, S. H. Chung, and P. L. Y. Chan, "An introduction of dominant genes in genetic algorithm for scheduling of FMS," in *Proceedings of the 20th IEEE International Symposium on Intelligent Control, ISIC '05 and the 13th Mediterranean Conference on Control and Automation, MED '05*, pp. 1429–1434, Cyprus, June 2005.

[31] E. Murat, "A genetic algorithm approach for solving flexible job shop scheduling problems," in *Proceedings of the in 35th International Conference on Computers & Industrial Engineering*, M. B. D. C. K. M. I. D. G. A. S. S. Eid, Ed., pp. 245–250, Istanbul Technical University, Istanbul, Turkey, 2005.

[32] K. F. Guimarães and M. A. Fernandes, "An approach for flexible Job-Shop Scheduling with separable sequence-dependent setup time," in *Proceedings of the 2006 IEEE International Conference on Systems, Man and Cybernetics*, pp. 3727–3731, Taiwan, October 2006.

[33] M. Zandieh, I. Mahdavi, and A. Bagheri, "Solving the flexible job-shop scheduling problem by a genetic algorithm," *Journal of Applied Sciences*, vol. 8, no. 24, pp. 4650–4655, 2008.

[34] F. Pezzella, G. Morganti, and G. Ciaschetti, "A genetic algorithm for the flexible job-shop scheduling problem," *Computers & Operations Research*, vol. 35, no. 10, pp. 3202–3212, 2008.

[35] T. Yu, J. Zhou, J. Fang, Y. Gong, and W. Wang, "Dynamic scheduling of flexible job shop based on genetic algorithm," in *Proceedings of the IEEE International Conference on Automation and Logistics, ICAL 2008*, pp. 2014–2019, China, September 2008.

[36] X. Du, Z. Li, and W. Xiong, "Flexible Job Shop scheduling problem solving based on genetic algorithm with model constraints," in *Proceedings of the 2008 IEEE International Conference on Industrial Engineering and Engineering Management, IEEM 2008*, pp. 1239–1243, Singapore, December 2008.

[37] B. S. Girish and N. Jawahar, "Scheduling job shop associated with multiple routings with genetic and ant colony heuristics," *International Journal of Production Research*, vol. 47, no. 14, pp. 3891–3917, 2009.

[38] H. Qiu, W. Zhou, and H. Wang, "A genetic algorithm-based approach to flexible job-shop scheduling problem," in *Proceedings of the 5th International Conference on Natural Computation, ICNC 2009*, pp. 81–85, China, August 2009.

[39] L. Li and J.-Z. Huo, "Multi-objective flexible Job-shop scheduling problem in steel tubes production," *Xitong Gongcheng Lilun yu Shijian/System Engineering Theory and Practice*, vol. 29, no. 8, pp. 117–126, 2009.

[40] X. Yang, J. Zeng, and J. Liang, "Apply MGA to multi-objective flexible job shop scheduling problem," in *Proceedings of the 2009 International Conference on Information Management, Innovation Management and Industrial Engineering, ICIII 2009*, pp. 436–439, China, December 2009.

[41] X.-L. Wu and S.-J. Li, "Mass variety and small batch scheduling in the flexible job shop," in *Proceedings of the 2009 2nd International Conference on Biomedical Engineering and Informatics, BMEI 2009*, China, October 2009.

[42] L. De Giovanni and F. Pezzella, "An improved genetic algorithm for the distributed and flexible job-shop scheduling problem," *European Journal of Operational Research*, vol. 200, no. 2, pp. 395–408, 2010.

[43] P. Fattahi and A. Fallahi, "Dynamic scheduling in flexible job shop systems by considering simultaneously efficiency and stability," *CIRP Journal of Manufacturing Science and Technology*, vol. 2, no. 2, pp. 114–123, 2010.

[44] P. Unachak and E. Goodman, "Solving multiobjective flexible job-shop scheduling using an adaptive representation," in *Proceedings of the 12th Annual Genetic and Evolutionary Computation Conference, GECCO-2010*, pp. 737–742, USA, July 2010.

[45] J. Liu, C. Zhang, L. Gao, and X. Wang, "Research on flexible job-shop scheduling problem under uncertainty based on genetic algorithm," in *Proceedings of the 2010 6th International Conference on Natural Computation, ICNC'10*, pp. 2462–2467, China, August 2010.

[46] X. Liu, C. Liu, and Z. Tao, "Study on scheduling optimization for flexible job shop," *Applied Mechanics and Materials*, vol. 26-28, pp. 821–825, 2010.

[47] M. Wan, X. Fan, F. Zhang, and C. Bai, "An integrated genetic algorithm for flexible jobshop scheduling problem," in *Proceedings of the 2010 International Conference on Computational Intelligence and Software Engineering, CiSE 2010*, China, December 2010.

[48] S.-T. Hsieh, S.-Y. Chiu, and S.-J. Yen, "An improved multi-objective genetic algorithm for solving flexible job shop problem," in *Proceedings of the IET International Conference on Frontier Computing. Theory, Technologies and Applications*, pp. 427–431, Taiwan, August 2010.

[49] W. Sun, Y. Pan, X. Lu, and Q. Ma, "Research on flexible job-shop scheduling problem based on a modified genetic algorithm," *Journal of Mechanical Science and Technology*, vol. 24, no. 10, pp. 2119–2125, 2010.

[50] N. Al-Hinai and T. Y. Elmekkawy, "Robust and stable flexible job shop scheduling with random machine breakdowns using a hybrid genetic algorithm," *International Journal of Production Economics*, vol. 132, no. 2, pp. 279–291, 2011.

[51] L. Ju, J. Yang, and B. Liu, "The optimization of flexible job shop scheduling problem based on improved dual coding non-dominated sorting genetic algorithm," *Advanced Materials Research*, vol. 291-294, pp. 2537–2540, 2011.

[52] X. X. Liu, C. B. Liu, and Z. Tao, "Research on bi-objective scheduling of dual-resource constrained flexible job shop," *Advanced Materials Research*, vol. 211-212, pp. 1091–1095, 2011.

[53] Y. Mati, C. Lahlou, and S. Dauzère-Pérès, "Modelling and solving a practical flexible job-shop scheduling problem with blocking constraints," *International Journal of Production Research*, vol. 49, no. 8, pp. 2169–2182, 2011.

[54] M. Wan, X. Xu, and J. Nan, "Flexible job-shop scheduling with integrated genetic algorithm," in *Proceedings of the 4th International Workshop on Advanced Computational Intelligence, IWACI 2011*, pp. 13–16, China, October 2011.

[55] E. Moradi, S. M. T. Fatemi Ghomi, and M. Zandieh, "Bi-objective optimization research on integrated fixed time interval preventive maintenance and production for scheduling flexible job-shop problem," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7169–7178, 2011.

[56] E. S. Nicoară, F. G. Filip, and N. Paraschiv, "Simulation-based optimization using genetic algorithms for multi-objective flexible JSSP," *Studies in Informatics and Control*, vol. 20, no. 4, pp. 333–344, 2011.

[57] R. K. Phanden, A. Jain, and R. Verma, "A Genetic Algorithm-based approach for flexible job shop scheduling," *Applied Mechanics and Materials*, vol. 110-116, pp. 3930–3937, 2012.

[58] J. J. Yang, L. Y. Ju, and B. Y. Liu, "The improved genetic algorithm for multi-objective flexible job shop scheduling problem," *Applied Mechanics and Materials*, vol. 66-68, pp. 870–875, 2011.

[59] G. Zhang, L. Gao, and Y. Shi, "An effective genetic algorithm for the flexible job-shop scheduling problem," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3563–3573, 2011.

[60] B. Ak and E. Koc, "A guide for genetic algorithm based on parallel machine scheduling and flexible job-shop scheduling," *Procedia - Social and Behavioral Sciences*, vol. 62, pp. 817–823, 2012.

[61] S. Fan and J. Wang, "Scheduling for the flexible job-shop problem based on genetic algorithm (GA)," *Advanced Materials Research*, vol. 457-458, pp. 616–619, 2012.

[62] M. Li and M. Yao, "Research on improved genetic algorithm solving flexible job-shop problem," *Advanced Materials Research*, vol. 479-481, pp. 1918–1921, 2012.

[63] M. Rabiee, M. Zandieh, and P. Ramezani, "Bi-objective partial flexible job shop scheduling problem: NSGA-II, NRGA, MOGA and PAES approaches," *International Journal of Production Research*, vol. 50, no. 24, pp. 7327–7342, 2012.

[64] M. Saravanan, E. Nagamani, and D. Chandrasekaran, "Single objective evolutionary algorithm for flexible job-shop scheduling problem," *International Journal of Mathematics Trends and Technology*, vol. 3, no. 2, pp. 78–81, 2012.

[65] W. Teekeng and A. Thammano, "Modified genetic algorithm for flexible job-shop scheduling problems," *Procedia Computer Science*, vol. 12, no. 12, pp. 122–128, 2012.

[66] S. Vaghefinezhad and K. Y. Wong, "A genetic algorithm approach for solving group technology problem with process plan flexibility," in *Proceedings of the 2012 14th International Conference on Modelling and Simulation, UKSim 2012*, pp. 52–58, UK, March 2012.

[67] J. Autuori, F. Hnaien, F. Yalaoui, A. Hamzaoui, and N. Essounbouli, "Comparison of solution space exploration by NSGA2 and SPEA2 for Flexible Job Shop Problem," in *Proceedings of the 2013 International Conference on Control, Decision and Information Technologies, CoDIT 2013*, pp. 750–755, Tunisia, May 2013.

[68] K. I. Chaudhry and A. A. Khan, "A genetic algorithm for flexible job shop scheduling," in *Proceedings of the in Proceedings of the World Congress on Engineering*, London, UK, 2013.

[69] S. H. A. Rahmati, M. Zandieh, and M. Yazdani, "Developing two multi-objective evolutionary algorithms for the multi-objective flexible job shop scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 64, no. 5–8, pp. 915–932, 2013.

[70] Y. M. Wang, H. L. Yin, and K. D. Qin, "A novel genetic algorithm for flexible job shop scheduling problems with machine disruptions," *The International Journal of Advanced Manufacturing Technology*, vol. 68, no. 5-8, pp. 1317–1326, 2013.

[71] Z. Tao and O. Rose, "Scheduling in a flexible job shop with continuous operations at the last stage," in *Proceedings of the 15th ASIM Simulation in Production and Logistics*, pp. 611–620, Paderborn, Germany, 2013.

[72] M. Cwiek and J. Nalepa, "A fast genetic algorithm for the flexible job shop scheduling problem," in *Proceedings of the 16th Genetic and Evolutionary Computation Conference, GECCO 2014*, pp. 1449-1450, Canada, July 2014.

[73] Y. Demir and S. K. İşleyen, "An effective genetic algorithm for flexible job-shop scheduling with overlapping in operations," *International Journal of Production Research*, vol. 52, no. 13, pp. 3905–3921, 2014.

[74] C. Gutierrez, "Overlap algorithms in flexible job-shop scheduling," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 2, no. 6, p. 41, 2014.

[75] H.-C. Chang, H.-T. Tsai, and T.-K. Liu, "Application of genetic algorithm to optimize unrelated parallel machines of flexible job-shop scheduling problem," in *Proceedings of the 11th IEEE International Conference on Control and Automation, IEEE ICCA 2014*, pp. 596–599, twn, June 2014.

[76] A. M. Moghadam, K. Y. Wong, and H. Piroozfard, "An efficient genetic algorithm for flexible job-shop scheduling problem," in *Proceedings of the 2014 IEEE International Conference on Industrial Engineering and Engineering Management, IEEM 2014*, pp. 1409–1413, Malaysia, December 2014.

[77] H. Na and J. Park, "Multi-level job scheduling in a flexible job shop environment," *International Journal of Production Research*, vol. 52, no. 13, pp. 3877–3887, 2014.

[78] W. J. Song, C. Y. Zhang, W. W. Lin, and X. Y. Shao, "Flexible job-shop scheduling problem with maintenance activities considering energy consumption," *Applied Mechanics and Materials*, vol. 521, pp. 707–713, 2014.

[79] T.-K. Liu, Y.-P. Chen, and J.-H. Chou, "Solving Distributed and Flexible Job-Shop Scheduling Problems for a Real-World Fastener Manufacturer," *IEEE Access*, vol. 2, pp. 1598–1606, 2014.

[80] G. Zambrano Rey, A. Bekrar, V. Prabhu, and D. Trentesaux, "Coupling a genetic algorithm with the distributed arrival-time control for the JIT dynamic scheduling of flexible job-shops," *International Journal of Production Research*, vol. 52, no. 12, pp. 3688–3709, 2014.

[81] G. Candan and H. R. Yazgan, "Genetic algorithm parameter optimisation using Taguchi method for a flexible manufacturing system scheduling problem," *International Journal of Production Research*, vol. 53, no. 3, pp. 897–915, 2015.

[82] H.-C. Chang, Y.-P. Chen, T.-K. Liu, and J.-H. Chou, "Solving the Flexible Job Shop Scheduling Problem with Makespan Optimization by Using a Hybrid Taguchi-Genetic Algorithm," *IEEE Access*, vol. 3, pp. 1740–1754, 2015.

[83] J. Chen, W. Weng, G. Rong, and S. Fujimura, "Integrating Genetic Algorithm with Time Control for Just-In-Time Scheduling Problems," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 893–897, 2015.

[84] I. Driss, K. N. Mouss, and A. Laggoun, "A new genetic algorithm for flexible job-shop scheduling problems," *Journal of Mechanical Science and Technology*, vol. 29, no. 3, pp. 1273–1281, 2015.

[85] S. Ishikawa, R. Kubota, and K. Horio, "Effective hierarchical optimization by a hierarchical multi-space competitive genetic algorithm for the flexible job-shop scheduling problem," *Expert Systems with Applications*, vol. 42, no. 24, pp. 9434–9440, 2015.

[86] L. Jiang and Z. Du, "An improved genetic algorithm for flexible job shop scheduling problem," in *Proceedings of the 2015 2nd International Conference on Information Science and Control Engineering, ICISCE 2015*, pp. 127–131, China, April 2015.

[87] P.-H. Lu, M.-C. Wu, H. Tan, Y.-H. Peng, and C.-F. Chen, "A genetic algorithm embedded with a concise chromosome representation for distributed and flexible job-shop scheduling problems," *Journal of Intelligent Manufacturing*, 2015.

[88] S. K. Parjapati and J. Ajai, "Optimization of flexible job shop scheduling problem with sequence dependent setup times using genetic algorithm approach," *International Journal of Mathematical, Computational, Natural and Physical Engineering*, vol. 9, pp. 41–47, 2015.

[89] R. K. Phanden and A. Jain, "Assessment of makespan performance for flexible process plans in job shop scheduling," *IFAC-PapersOnLine*, vol. 28, no. 3, pp. 1948–1953, 2015.

[90] X. Yang, J. Wang, M. Hou, and X. Fan, "Job shop scheduling based on genetic algorithm using Matlab," in *Proceedings of the IEEE Advanced Information Technology, Electronic and Automation Control Conference, IAEAC 2015*, pp. 772–775, China, December 2015.

[91] E. Ahmadi, M. Zandieh, M. Farrokh, and S. M. Emami, "A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms," *Computers & Operations Research*, vol. 73, pp. 56–66, 2016.

[92] X. Yang, Z. Zeng, R. Wang, and X. Sun, "Bi-objective flexible job-shop scheduling problem considering energy consumption under stochastic processing times," *PLoS ONE*, vol. 11, no. 12, article e0167427, 2016.

[93] M. Chen and J.-L. Li, "Genetic Algorithm Combined with Gradient Information for Flexible Job-shop Scheduling Problem with Different Varieties and Small Batches," in *Proceedings of the 3rd International Conference on Mechatronics and Mechanical Engineering, ICMME 2016*, China, October 2016.

[94] A. R. Elgendy, H. Mohammed, and A. Elhakeem, "Optimizing Dynamic Flexible Job Shop Scheduling Problem Based on Genetic Algorithm," *International Journal of Current Engineering and Technology*, 2017.

[95] L. Qiao, Z. Zhang, and M. K. Nawaz, "Genetic algorithm based novel methodology of multi-constraint job scheduling," in *Proceedings of the 2017 5th International Conference on Enterprise Systems (ES)*, 2017.

[96] L. Wang, C. Luo, and J. Cai, "A variable interval rescheduling strategy for dynamic flexible job shop scheduling problem by improved genetic algorithm," *Journal of Advanced Transportation*, vol. 2017, Article ID 1527858, 12 pages, 2017.

[97] E. Zhou, J. Zhu, and L. Deng, "Flexible job-shop scheduling based on genetic algorithm and simulation validation," in *Proceedings of the 13th Global Congress on Manufacturing and Management, GCMM 2016*, China, November 2016.

[98] Y. He, W. Weng, and S. Fujimura, "Improvements to genetic algorithm for flexible job shop scheduling with overlapping in operations," in *Proceedings of the 16th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2017*, pp. 791–796, China, May 2017.

[99] M.-C. Wu, C.-S. Lin, C.-H. Lin, and C.-F. Chen, "Effects of different chromosome representations in developing genetic algorithms to solve DFJS scheduling problems," *Computers & Operations Research*, vol. 80, pp. 101–112, 2017.

[100] Q. Deng, G. Gong, X. Gong, L. Zhang, W. Liu, and Q. Ren, "A Bee Evolutionary Guiding Nondominated Sorting Genetic Algorithm II for Multiobjective Flexible Job-Shop Scheduling," *Computational Intelligence and Neuroscience*, vol. 2017, Article ID 5232518, 2017.

[101] W. Zhang, J. B. Wen, Y. C. Zhu, and Y. Hu, "Multi-objective scheduling simulation of flexible job-shop based on multi-population genetic algorithm," *International Journal of Simulation Modelling*, vol. 16, no. 2, pp. 313–321, 2017.

[102] Z. Jing, H. Jin, and Y. Zhu, "Multi-objective integrated optimization problem of preventive maintenance planning and flexible job-shop scheduling," in *Proceedings of the 23rd International Conference on Industrial Engineering and Engineering Management*, Springer, 2016.

[103] H. Zhang and M. Gen, "Multistage-based genetic algorithm for flexible job-shop scheduling problem," *Complexity International*, vol. 11, pp. 223–232, 2005.

[104] N. B. Ho and J. C. Tay, "LEGA: an architecture for learning and evolving flexible job-shop schedules," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, vol. 2, pp. 1380–1387, September 2005.

[105] I. Saad, S. Hammadi, P. Borne, and M. Benrejeb, "Aggregative approach for the multiobjective optimization flexible job-shop scheduling problems," in *Proceedings of the ICSSSM'06: 2006 International Conference on Service Systems and Service Management*, pp. 889–894, France, October 2006.

[106] N. B. Ho, J. C. Tay, and E. M. K. Lai, "An effective architecture for learning and evolving flexible job-shop schedules," *European Journal of Operational Research*, vol. 179, no. 2, pp. 316–333, 2007.

[107] D.-M. Lei and X.-P. Guo, "Solving fuzzy flexible job shop scheduling problems using genetic algorithm," in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics, ICMLC*, pp. 1014–1019, China, July 2008.

[108] I. Saad, S. Hammadi, M. Benrejeb, and P. Borne, "Choquet integral for criteria aggregation in the flexible job-shop scheduling problems," *Mathematics and Computers in Simulation*, vol. 76, no. 5-6, pp. 447–462, 2008.

[109] F. M. Defersha and M. Chen, "A coarse-grain parallel genetic algorithm for flexible job-shop scheduling with lot streaming," in *Proceedings of the 2009 International Conference on Computational Science and Engineering, CSE '09*, pp. 201–208, Canada, August 2009.

[110] W. Qiao and Q. Li, "Solving the flexible job shop scheduling problems based on the adaptive genetic algorithm," in *Proceedings of the 2009 International Forum on Computer Science-Technology and Applications, IFCSTA 2009*, pp. 97–100, China, December 2009.

[111] F. Asma, J. Olfa, B. H.-A. Atidel, and B. N. Ibtissem, "The performance of the AGAIS (II) algorithm for the resolution of the Flexible Job-shop Scheduling Problem," in *Proceedings of the 2010 2nd International Conference on Engineering System Management and Applications, ICESMA 2010*, are, April 2010.

[112] H. Asmuni and Y. R. Lee, "A Study of Cooperative Co-evolutionary Genetic Algorithm for Solving Flexible Job Shop Scheduling Problem," in *World Academy of Science, Engineering and Technology*, pp. 348–354, 2010.

[113] F. M. Defersha and M. Chen, "A parallel genetic algorithm for a flexible job-shop scheduling problem with sequence dependent setups," *The International Journal of Advanced Manufacturing Technology*, vol. 49, no. 1-4, pp. 263–279, 2010.

[114] G. Zhang, "Using matrix-coded genetic algorithm for solving the flexible job-shop scheduling," in *Proceedings of the 2010 International Conference on Computational Intelligence and Software Engineering, CiSE 2010*, China, December 2010.

[115] D. Lei, "A genetic algorithm for flexible job shop scheduling with fuzzy processing time," *International Journal of Production Research*, vol. 48, no. 10, pp. 2995–3013, 2010.

[116] E. Moradi, S. M. T. Fatemi Ghomi, and M. Zandieh, "An efficient architecture for scheduling flexible job-shop with machine availability constraints," *The International Journal of Advanced Manufacturing Technology*, vol. 51, no. 1-4, pp. 325–339, 2010.

[117] X. Yaoqun and W. Wenjian, "Modified Adaptive Genetic Algorithm for flexible job-shop scheduling problem," in *Proceedings of the 2010 International Conference on Future Information Technology and Management Engineering, FITME 2010*, pp. 52–55, China, October 2010.

[118] Y.-L. Zheng, Y.-X. Li, D.-M. Lei, and C.-X. Ma, "Scheduling flexible job shop with fuzzy processing time using co-evolutionary genetic algorithm," in *Proceedings of the 2010 2nd WRI Global Congress on Intelligent Systems, GCIS 2010*, pp. 331–334, China, December 2010.

[119] C. Zhang, X. Wang, and L. Gao, "An improved genetic algorithm for multi-objective flexible job-shop scheduling problem," *Advanced Materials Research*, vol. 97-101, pp. 2449–2454, 2010.

[120] C. Gutiérrez and I. García-Magariño, "Modular design of a hybrid genetic algorithm for a flexible job-shop scheduling problem," *Knowledge-Based Systems*, vol. 24, no. 1, pp. 102–112, 2011.

[121] Y. Pan, W. X. Zhang, T. Y. Gao, Q. Y. Ma, and D. J. Xue, "An adaptive Genetic Algorithm for the Flexible Job-shop Scheduling Problem," in *Proceedings of the 2011 IEEE International Conference on Computer Science and Automation Engineering, CSAE 2011*, pp. 405–409, China, June 2011.

[122] X. Li, L. Gao, L. Zhang, and W. Li, "An effective multi-swarm collaborative evolutionary algorithm for flexible job shop scheduling problem," in *Proceedings of the 2011 15th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2011*, pp. 281–284, Switzerland, June 2011.

[123] V. Jain, R. Agrawal, L. Pattanaik, and S. Kumar, "Scheduling of a flexible job-shop using a multi-objective genetic algorithm," *Journal of Advances in Management Research*, vol. 9, no. 2, pp. 178–188, 2012.

[124] N. Al-Hinai and T. Y. ElMekkawy, "Solving the flexible job shop scheduling problem with uniform processing time uncertainty," *World Academy of Science, Engineering and Technology*, vol. 6, no. 4, pp. 848–853, 2012.

[125] D. Lei, "Co-evolutionary genetic algorithm for fuzzy flexible job shop scheduling," *Applied Soft Computing*, vol. 12, no. 8, pp. 2237–2245, 2012.

[126] P. Paul Pandian, S. Saravana Sankar, S. G. Ponnambalam, and M. Victor Raj, "Scheduling of automated guided vehicle and flexible jobshop using jumping genes Genetic algorithm," *American Journal of Applied Sciences*, vol. 9, no. 10, pp. 1706–1720, 2012.

[127] W. F. Mahmudy, R. M. Marian, and L. H. S. Luong, "Real coded genetic algorithms for solving flexible job-shop scheduling problem- Part II: Optimization," *Advanced Materials Research*, vol. 701, pp. 364–369, 2013.

[128] Z. Jiang, L. Zuo, and E. Mingcheng, "Study on multi-objective flexible job-shop scheduling problem considering energy consumption," *Journal of Industrial Engineering and Management*, vol. 7, no. 3, pp. 589–604, 2014.

[129] H. Ren, H. Xu, and S. Sun, "Immune genetic algorithm for multi-objective flexible job-shop scheduling problem," in *Proceedings of the 28th Chinese Control and Decision Conference, CCDC 2016*, pp. 2167–2171, China, May 2016.

[130] J. Zhang and J. Yang, "Flexible job-shop scheduling with flexible workdays, preemption, overlapping in operations and satisfaction criteria: an industrial application," *International Journal of Production Research*, vol. 54, no. 16, pp. 4894–4918, 2016.

[131] M. Zhang and K. Wu, "An improved genetic algorithm for the re-entrant and flexible job-shop scheduling problem," in *Proceedings of the 28th Chinese Control and Decision Conference, CCDC 2016*, pp. 3399–3404, China, May 2016.

[132] H. Tamaki, T. Ono, H. Murao, and S. Kitamura, "Modeling and genetic solution of a class of flexible job shop scheduling problems," in *Proceedings of the 8th International Conference on Emerging Technologies and Factory Automation (ETFA 2001)*, pp. 343–350, France, 2001.

[133] I. Kacem, S. Hammadi, and P. Borne, "Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic," *Mathematics and Computers in Simulation*, vol. 60, no. 3–5, pp. 245–276, 2002.

[134] I. T. Tanev, T. Uozumi, and Y. Morotome, "Hybrid evolutionary algorithm-based real-world flexible job shop scheduling problem: Application service provider approach," *Applied Soft Computing*, vol. 5, no. 1, pp. 87–100, 2004.

[135] N. Zribi, I. Kacem, A. El Kamel, and P. Borne, "Optimization by phases for the flexible job-shop scheduling problem," in *Proceedings of the 2004 5th Asian Control Conference*, pp. 1889–1895, aus, July 2004.

[136] N. Zribi, A. Kacem, and P. El-Kamel, "Minimizing the total tardiness in a flexible job-shop," in *Proceedings of the International Symposium on Intelligent Automation and Control, World Automation Congress*, Budapest, Hungary, 2006.

[137] J. Gao, M. Gen, and L. Sun, "Scheduling jobs and maintenances in flexible job shop with a hybrid genetic algorithm," *Journal of Intelligent Manufacturing*, vol. 17, no. 4, pp. 493–507, 2006.

[138] S. I. Butt and H.-F. Sun, "Application of genetic algorithms and rules in the scheduling of flexible job shops," *Journal of Applied Sciences*, vol. 6, no. 7, pp. 1586–1590, 2006.

[139] F. Neifar, M. Gzara, A. Moukrim, and T. Loukil, "Hybrid evolutionary algorithm with insertion heuristics for the flexible job shop problem," in *Proceedings of the ICSSSM'06: 2006 International Conference on Service Systems and Service Management*, pp. 1211–1216, France, October 2006.

[140] J. Gao, M. Gen, and L. Sun, "A hybrid of genetic algorithm and bottleneck shifting for flexible job shop scheduling problem," in *Proceedings of the 8th Annual Genetic and Evolutionary Computation Conference 2006*, pp. 1157–1164, usa, July 2006.

[141] G. Vilcot, J.-C. Billaut, and C. Esswein, "A genetic algorithm for A bicriteria flexible job shop scheduling problem," in *Proceedings of the ICSSSM'06: 2006 International Conference on Service Systems and Service Management*, pp. 1240–1244, France, October 2006.

[142] N. Zribi, A. El Kamel, and P. Borne, "Total tardiness in a flexible job-shop," in *Proceedings of the IMACS Multiconference on "Computational Engineering in Systems Applications", CESA*, pp. 1543–1549, chn, October 2006.

[143] J. Gao, M. Gen, L. Sun, and X. Zhao, "A hybrid of genetic algorithm and bottleneck shifting for multiobjective flexible job shop scheduling problems," *Computers & Industrial Engineering*, vol. 53, no. 1, pp. 149–162, 2007.

[144] N. Zribi, I. Kacem, A. El Kamel, and P. Borne, "Assignment and scheduling in flexible job-shops by hierarchical optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 4, pp. 652–661, 2007.

[145] J. Gao, L. Sun, and M. Gen, "A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems," *Computers & Operations Research*, vol. 35, no. 9, pp. 2892–2907, 2008.

[146] N. B. Ho and J. C. Tay, "Solving multiple-objective flexible job shop problems by evolution and local search," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 38, no. 5, pp. 674–685, 2008.

[147] I. Moon, S. Lee, and H. Bae, "Genetic algorithms for job shop scheduling problems with alternative routings," *International Journal of Production Research*, vol. 46, no. 10, pp. 2695–2705, 2008.

[148] G. Zhang, S. Yang, and L. Gao, "A genetic algorithm and tabu search for solving flexible job shop schedules," in *Proceedings of the 2008 International Symposium on Computational Intelligence and Design, ISCID 2008*, pp. 369–372, China, October 2008.

[149] M. Gholami and M. Zandieh, "Integrating simulation and genetic algorithm to schedule a dynamic flexible job shop," *Journal of Intelligent Manufacturing*, vol. 20, no. 4, pp. 481–498, 2009.

[150] S. Yang, Z. Guohui, G. Liang, and Y. Kun, "A novel initialization method for solving flexible job-shop schedulingproblem," in *Proceedings of the 2009 International Conference on Computers and Industrial Engineering, CIE 2009*, pp. 68–73, fra, July 2009.

[151] M. Frutos, A. C. Olivera, and F. Tohmé, "A memetic algorithm based on a NSGAII scheme for the flexible job-shop scheduling problem," *Annals of Operations Research*, vol. 181, pp. 745–765, 2010.

[152] M. Lan, T.-R. Xu, and L. Peng, "Solving flexible multi-objective JSP problem using a improved genetic algorithm," *Journal of Software* , vol. 5, no. 10, pp. 1107–1113, 2010.

[153] J.-Q. Li, Q.-K. Pan, and S.-X. Xie, "A hybrid variable neighborhood search algorithm for solving multi-objective flexible job

shop problems," *Computer Science and Information Systems*, vol. 7, no. 4, pp. 908–930, 2010.

[154] L. Song and X. Xu, "Flexible job shop scheduling problem solving based on genetic algorithm with chaotic local search," in *Proceedings of the 2010 6th International Conference on Natural Computation, ICNC'10*, pp. 2356–2360, China, August 2010.

[155] A. Motaghedi-larijani, K. Sabri-laghaie, and M. Heydari, "Solving flexible job shop scheduling with multi objective approach," *International Journal of Industiral Engineering & Production Research*, vol. 21, no. 4, pp. 197–209, 2010.

[156] X. Wang, L. Gao, C. Zhang, and X. Shao, "A multi-objective genetic algorithm based on immune and entropy principle for flexible job-shop scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 51, no. 5-8, pp. 757–767, 2010.

[157] X.-H. Xu, L.-L. Zeng, and Y.-W. Fu, "Hybrid particle swarm optimization for flexible job-shop scheduling problem and its implementation?" in *Proceedings of the 2010 IEEE International Conference on Information and Automation, ICIA 2010*, pp. 1155–1159, China, June 2010.

[158] G. Zhang, L. Gao, and Y. Shi, "A novel variable neighborhood genetic algorithm for multi-objective flexible job-shop scheduling problems," *Advanced Materials Research*, vol. 118-120, pp. 369–373, 2010.

[159] G. Zhang, L. Gao, and Y. Shi, "A genetic algorithm and tabu search for multi objective flexible job shop scheduling problems," in *Proceedings of the 1st International Conference on Computing Control and Industrial Engineering, CCIE 2010*, pp. 251–254, China, June 2010.

[160] N. Al-Hinai and T. Y. Elmekkawy, "An efficient hybridized genetic algorithm architecture for the flexible job shop scheduling problem," *Flexible Services and Manufacturing Journal*, vol. 23, no. 1, pp. 64–85, 2011.

[161] E. B. Azardoost, "A hybrid algorithm for multi objective flexible job shop scheduling problem," in *Proceedings of the 2nd International Conference on Industrial Engineering and Operations Management (IEOM*, pp. 795–801, Kuala Lumpur, Malaysia, 2011.

[162] C. Xianzhou and Y. Zhenhe, "An improved genetic algorithm for dual-resource constrained flexible job shop scheduling," in *Proceedings of the 2011 4th International Conference on Intelligent Computation Technology and Automation, ICICTA 2011*, pp. 42–45, China, March 2011.

[163] K. Ida and K. Oka, "Flexible job-shop scheduling problem by genetic algorithm," *Electrical Engineering in Japan (English translation of Denki Gakkai Ronbunshi)*, vol. 177, no. 3, pp. 28–35, 2011.

[164] J. Jiang, M. Wen, K. Ma, X. Long, and J. Li, "Hybrid genetic algorithm for flexible job-shop scheduling with multi-objective," *Journal of Information and Computational Science*, vol. 8, no. 11, pp. 2197–2205, 2011.

[165] L. Di and T. Ze, "A genetic algorithm with Tabu Search for multi-objective scheduling constrained flexible job shop," in *Proceedings of the 2011 Cross Strait Quad-Regional Radio Science and Wireless Technology Conference, CSQRWC 2011*, pp. 1662–1665, China, July 2011.

[166] J. Tang, G. Zhang, B. Lin, and B. Zhang, "A hybrid algorithm for flexible job-shop scheduling problem," in *Proceedings of the 2011 International Conference on Advanced in Control Engineering and Information Science, CEIS 2011*, pp. 3678–3683, China, August 2011.

[167] W. Wei, Y. Feng, J. Tan, and I. Hagiwara, "Flexible job shop scheduling multi-objective optimization based on improved strength pareto evolutionary algorithm," *Advanced Materials Research*, vol. 186, pp. 546–551, 2011.

[168] L. Xing, Y. Chen, and K. Yang, "Multi-population interactive coevolutionary algorithm for flexible job shop scheduling problems," *Computational Optimization and Applications*, vol. 48, no. 1, pp. 139–155, 2011.

[169] J. C. Chen, C.-C. Wu, C.-W. Chen, and K.-H. Chen, "Flexible job shop scheduling with parallel machines using Genetic Algorithm and Grouping Genetic Algorithm," *Expert Systems with Applications*, vol. 39, no. 11, pp. 10016–10021, 2012.

[170] A. R. Fard, B. Y. Yegane, and N. Khanlarzade, "Hybrid genetic algorithm for flexible job shop scheduling with overlapping in operations," *Advanced Materials Research*, vol. 433-440, pp. 1499–1505, 2012.

[171] H. Hui, "Approach for multi-objective flexible job shop scheduling," *Advanced Materials Research*, vol. 542-543, pp. 407–410, 2012.

[172] R. Javadi and M. Hasanzadeh, "A new method for hybridizing metaheuristics for multi-objective flexible job shop scheduling," in *Proceedings of the 2012 2nd International eConference on Computer and Knowledge Engineering, ICCKE 2012*, pp. 105–110, Iran, October 2012.

[173] J. Song, "Integrated optimal algorithm for flexible job shop scheduling with due-date window," in *Proceedings of the 10th World Congress on Intelligent Control and Automation, WCICA 2012*, pp. 638–642, China, July 2012.

[174] S. Lee, I. Moon, H. Bae, and J. Kim, "Flexible job-shop scheduling problems with 'AND'/'OR' precedence constraints," *International Journal of Production Research*, vol. 50, no. 7, pp. 1979–2001, 2012.

[175] Q. Zhang, H. Manier, and M.-A. Manier, "A hybrid metaheuristic algorithm for flexible job-shop scheduling problems with transportation constraints," in *Proceedings of the 14th International Conference on Genetic and Evolutionary Computation, GECCO'12*, pp. 441–448, USA, July 2012.

[176] R. Tavakkoli-Moghaddam, N. ShahsavariPour, H. Mohammadi-Andargoli, and M. H. Abolhasani-Ashkezari, "Duplicate genetic algorithm for scheduling a bi-objective flexible job shop problem," *International Journal of Research in Industrial Engineering*, vol. 1, no. 2, pp. 10–26, 2012.

[177] J. Xiong, X. Tan, K. Yang, L. Xing, and Y. Chen, "A Hybrid Multiobjective Evolutionary Approach for Flexible Job-Shop Scheduling Problems," *Mathematical Problems in Engineering*, vol. 2012, pp. 1–27, 2012.

[178] Q. Zhang, H. Manier, and M.-A. Manier, "A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times," *Computers & Operations Research*, vol. 39, no. 7, pp. 1713–1723, 2012.

[179] M. E. T. Araghi, F. Jolai, and M. Rabiee, "Incorporating learning effect and deterioration for solving a SDST flexible job-shop scheduling problem with a hybrid meta-heuristic approach," *International Journal of Computer Integrated Manufacturing*, vol. 27, no. 8, pp. 733–746, 2014.

[180] F. Geyik and A. T. Dosdoğru, "Process plan and part routing optimization in a dynamic flexible job shop scheduling environment: An optimization via simulation approach," *Neural Computing and Applications*, vol. 23, no. 6, pp. 1631–1641, 2013.

[181] A. Jalilvand-Nejad and P. Fattahi, "A mathematical model and genetic algorithm to cyclic flexible job shop scheduling

problem," *Journal of Intelligent Manufacturing*, vol. 26, no. 6, pp. 1085–1098, 2015.

[182] T. Jamrus, C.-F. Chein, M. Gen et al., "Hybrid particle swarm optimization combined with genetic operators and cauchy distribution for flexible job-shop scheduling problem," p. 12, Asia Pacific Industrial Engineering and Management System Conference, Cebu, Philippines, 2013.

[183] N. Shahsavari-Pour and B. Ghasemishabankareh, "A novel hybrid meta-heuristic algorithm for solving multi objective flexible job shop scheduling," *Journal of Manufacturing Systems*, vol. 32, no. 4, pp. 771–780, 2013.

[184] J. Xiong, L.-N. Xing, and Y.-W. Chen, "Robust scheduling for multi-objective flexible job-shop problems with random machine breakdowns," *International Journal of Production Economics*, vol. 141, no. 1, pp. 112–126, 2013.

[185] Y. Yuan and H. Xu, "A memetic algorithm for the multi-objective flexible job shop scheduling problem," in *Proceedings of the 2013 15th Genetic and Evolutionary Computation Conference, GECCO 2013*, pp. 559–566, Netherlands, July 2013.

[186] D. Zhou and L. Zeng, "A flexible job-shop scheduling method based on hybrid genetic annealing algorithm," *Journal of Information and Computational Science*, vol. 10, no. 17, pp. 5541–5549, 2013.

[187] N. Selvaraj, "An effective hybrid heuristic to solve multiobjective flexible job shop scheduling problems," *Academic Journal of Science*, vol. 3, no. 1, pp. 61–73, 2014.

[188] S.-X. Wang, C.-Y. Zhang, and L.-L. Jin, "A hybrid genetic algorithm for flexible job-shop scheduling problem," *Advanced Materials Research*, vol. 889-890, pp. 1179–1184, 2014.

[189] Y. Li and G. Zhou, "The flexible job shop scheduling based on ATC and GATS hybrid algorithm," in *Proceedings of the 2014 IEEE International Conference on Information and Automation, ICIA 2014*, pp. 860–864, China, July 2014.

[190] S. Yokoyama, H. Iizuka, and M. Yamamoto, "Hybrid genetic algorithm with priority rule-based reconstruction for flexible job-shop scheduling," in *Proceedings of the 2014 Joint 7th International Conference on Soft Computing and Intelligent Systems, SCIS 2014 and 15th International Symposium on Advanced Intelligent Systems, ISIS 2014*, pp. 749–754, Japan, December 2014.

[191] A. Azzouz, M. Ennigrou, and B. Jlifi, "Diversifying TS using GA in multi-agent system for solving flexible job shop problem," in *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2015*, pp. 94–101, fra, July 2015.

[192] H. E. Nouri, O. B. Driss, and K. Ghédira, "Hybrid metaheuristics within a holonic multiagent model for the Flexible Job Shop Problem," in *Proceedings of the 19th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES 2015*, pp. 83–92, Singapore, September 2015.

[193] J. J. Palacios, M. A. González, C. R. Vela, I. González-Rodríguez, and J. Puente, "Genetic tabu search for the fuzzy flexible job shop problem," *Computers & Operations Research*, vol. 54, pp. 74–89, 2015.

[194] M. Rohaninejad, A. Kheirkhah, and P. Fattahi, "Simultaneous lot-sizing and scheduling in flexible job shop problems," *The International Journal of Advanced Manufacturing Technology*, 2014.

[195] M. Rohaninejad, A. Kheirkhah, P. Fattahi, and B. Vahedi-Nouri, "A hybrid multi-objective genetic algorithm based on the ELECTRE method for a capacitated flexible job shop scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 77, no. 1–4, pp. 51–66, 2015.

[196] A. Türkylmaz and S. Bulkan, "A hybrid algorithm for total tardiness minimisation in flexible job shop: Genetic algorithm with parallel VNS execution," *International Journal of Production Research*, vol. 53, no. 6, pp. 1832–1848, 2015.

[197] P. Kaweegitbundit and T. Eguchi, "Flexible job shop scheduling using genetic algorithm and heuristic rules," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 10, no. 1, 2016.

[198] N. Kundakci and O. Kulak, "Hybrid genetic algorithms for minimizing makespan in dynamic job shop scheduling problem," *Computers & Industrial Engineering*, vol. 96, pp. 31–51, 2016.

[199] X. Li and L. Gao, "An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem," *International Journal of Production Economics*, vol. 174, pp. 93–110, 2016.

[200] M. Sano, M. Nagao, and Y. Morinaga, "Balancing setup workers load of flexible job shop scheduling using hybrid genetic algorithm with tabu search strategy," *International Journal of Decision Support Systems*, vol. 2, no. 1, pp. 71–90, 2016.

[201] H. E. Nouri, O. Belkahla Driss, and K. Ghédira, "Simultaneous scheduling of machines and transport robots in flexible job shop environment using hybrid metaheuristics based on clustered holonic multiagent model," *Computers & Industrial Engineering*, vol. 102, pp. 488–501, 2016.

[202] C. Wang, N. Tian, C. Ji Zhi, and Y. Wang, "A hybrid evolutionary algorithm for flexible job shop scheduling problems," in *Proceedings of the 35th Chinese Control Conference, CCC 2016*, pp. 2690–2696, China, July 2016.

[203] W. Yi, X. Li, and B. Pan, "Solving flexible job shop scheduling using an effective memetic algorithm," *International Journal of Computer Applications in Technology*, vol. 53, no. 2, pp. 157–163, 2016.

[204] H. Mokhtari and A. Hasani, "An energy-efficient multi-objective optimization for flexible job-shop scheduling problem," *Computers & Chemical Engineering*, vol. 104, pp. 339–352, 2017.

[205] A. Azzouz, M. Ennigrou, and L. B. Said, "A self-adaptive evolutionary algorithm for solving flexible job-shop problem with sequence dependent setup time and learning effects," in *Proceedings of the 2017 IEEE Congress on Evolutionary Computation, CEC 2017*, pp. 1827–1834, Spain, June 2017.

[206] H.-C. Chang and T.-K. Liu, "Optimisation of distributed manufacturing flexible job shop scheduling by using hybrid genetic algorithms," *Journal of Intelligent Manufacturing*, 2015.

[207] A. Azzouz, M. Ennigrou, and L. Ben Said, "A hybrid algorithm for flexible job-shop scheduling problem with setup times," *International Journal of Production Management and Engineering*, vol. 5, no. 1, p. 23, 2017.

[208] H. E. Nouri, O. Belkahla Driss, and K. Ghédira, "Solving the flexible job shop problem by hybrid metaheuristics-based multiagent model," *Journal of Industrial Engineering International*, pp. 1–14, 2017.

[209] F. Geyik and I. H. Cedimoglu, "The strategies and parameters of tabu search for job-shop scheduling," *Journal of Intelligent Manufacturing*, vol. 15, no. 4, pp. 439–448, 2004.

[210] K. Genova, L. Kirilov, and V. Guliashki, "A survey of solving approaches for multiple objective flexible job shop scheduling problems," *Cybernetics and Information Technologies*, vol. 15, no. 2, pp. 3–22, 2015.

[211] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, Calif, USA, 1979.

[212] Y. Takeshi and N. Ryohei, "Job shop scheduling," *IEE Control Engineering Series*, pp. 134-134, 1997.

[213] P. Brucker, Y. N. Sotskov, and F. Werner, "Complexity of shop-scheduling problems with fixed number of jobs: a survey," *Mathematical Methods of Operations Research*, vol. 65, no. 3, pp. 461–481, 2007.

[214] P. Mellor, "A review of job shop scheduling," *OR*, vol. 17, no. 2, pp. 161–171, 1966.

[215] N. Bhatt and N. R. Chauhan, "Genetic algorithm applications on Job Shop Scheduling Problem: A review," in *Proceedings of the 2015 International Conference on Soft Computing Techniques and Implementations, ICSCTI 2015*, pp. 7–14, India, October 2015.

[216] T. H. Cormen, C. E. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, The MIT Press, 2009.

[217] E. Barbin, J. Borowczyk, J.-L. Chabert et al., *A history of algorithms: from the pebble to the microchip*, Springer Science & Business Media, 2012.

[218] C. Darwin, *On the Origin of Species by Means of Natural Selection*, J. Carroll, Ed., Broadview, Toronto, Canada, 2003.

[219] J. H. Holland, *Adaptation in Natural And Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.

[220] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Longman Publishing Co., Inc., 1989.

[221] T. El-Ghazali, *Metaheuristics: From Design to Implementation*, vol. 74, John Wiley & Sons, 2009.

[222] K. Mesghouni, S. Hammadi, and P. Borne, "Evolution programs for job-shop scheduling," in *Systems, Man, and Cybernetics*, Computational Cybernetics and Simulation, IEEE International Conference, 1997.

[223] P. Brandimarte, "Routing and scheduling in a flexible job shop by tabu search," *Annals of Operations Research*, vol. 41, no. 3, pp. 157–183, 1993.

[224] S. Dauzère-Pérès and J. Paulli, "An integrated approach for modeling and solving the general multiprocessor job-shop scheduling problem using tabu search," *Annals of Operations Research*, vol. 70, pp. 281–306, 1997.

[225] J. Hurink, B. Jurisch, and M. Thole, "Tabu search for the job-shop scheduling problem with multi-purpose machines," *OR Spectrum*, vol. 15, no. 4, pp. 205–215, 1994.

[226] M. Mastrolilli and L. M. Gambardella, "Effective neighbourhood functions for the flexible job shop problem," *Journal of Scheduling*, vol. 3, no. 1, pp. 3–20, 2000.

[227] J. Lin, "A hybrid biogeography-based optimization for the fuzzy flexible job-shop scheduling problem," *Knowledge-Based Systems*, vol. 78, no. 1, pp. 59–74, 2015.

[228] S. Karthikeyan, P. Asokan, S. Nickolas, and T. Page, "A hybrid discrete firefly algorithm for solving multi-objective flexible job shop scheduling problems," *International Journal of Bio-Inspired Computation*, vol. 7, no. 6, pp. 386–401, 2015.

[229] M. Ziaee, "A heuristic algorithm for the distributed and flexible job-shop scheduling problem," *The Journal of Supercomputing*, vol. 67, no. 1, pp. 69–83, 2014.

[230] M. Souad and C. B. Fayech, "A modified invasive weed optimization algorithm for multiobjective flexible job shop scheduling problems," *International Journal of Computer Science & Information Technology*, vol. 6, no. 6, pp. 51–60, 2014.

[231] Y. Yuan and H. Xu, "Flexible job shop scheduling using hybrid differential evolution algorithms," *Computers & Industrial Engineering*, vol. 65, no. 2, pp. 246–260, 2013.

[232] I. A. Chaudhry and A. A. Khan, "A research survey: review of flexible job shop scheduling techniques," *International Transactions in Operational Research*, vol. 23, no. 3, pp. 551–591, 2016.

[233] M. Gen and L. Lin, "Multiobjective evolutionary algorithm for manufacturing scheduling problems: state-of-the-art survey," *Journal of Intelligent Manufacturing*, vol. 25, no. 5, pp. 849–866, 2014.

[234] L. Vincent and C. Durai, "A survey on various optimization techniques with respect to flexible job shop scheduling," *International Journal of Scientific and Research Publications*, vol. 4, no. 3, 7 pages, 2014.

[235] H. Fisher and G. L. Thompson, "Probabilistic learning combinations of local job-shop scheduling rules," *Industrial scheduling*, vol. 3, no. 2, pp. 225–251, 1963.

[236] S. Lawrence, "Resource constrained project scheduling: an experimental investigation of heuristic scheduling techniques (supplement)," Graduate School of Industrial Administration, 1984.

[237] E. Taillard, "Benchmarks for basic scheduling problems," *European Journal of Operational Research*, vol. 64, no. 2, pp. 278–285, 1993.

[238] D. Y. Lee and F. DiCesare, "Scheduling Flexible Manufacturing Systems using Petri Nets and Heuristic Search," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 2, pp. 123–132, 1994.

[239] P. Fattahi, F. Jolai, and J. Arkat, "Flexible job shop scheduling with overlapping in operations," *Applied Mathematical Modelling*, vol. 33, no. 7, pp. 3076–3087, 2009.

[240] B. Dennis and M. J. Geiger, *Test Instances for the Flexible Job Shop Scheduling Problem with Work Centers*, Universitätsbibliothek der Helmut-Schmidt-Universität, Hamburg, Germany, 2012.

[241] D. Lawrence, "Job shop scheduling with genetic algorithms," in *Proceedings of the an international conference on genetic algorithms and their applications*, 1985.