RESEARCH ON IMPROVED FLOWER POLLINATION ALGORITHM FOR MULTI-OBJECTIVE GREEN FLEXIBLE JOB SHOP SCHEDULING

Jie ZHAO[1], Jinyu REN[2,*]

Addressing the multi-objective green flexible job shop scheduling problem (MGFJSP) considering green indicators, this paper analyzes the characteristics of the problem and considers relevant constraints. It establishes a mathematical model that minimizes total energy consumption and maximum completion time. An improved Pollen-based Pollination Algorithm (IFPA) is designed to solve the multi-objective discretized problem, which defines the discretized operation of the Pollen-based Pollination Algorithm (FPA). The main improvements include designing relevant encoding and decoding schemes, introducing an adaptive adjustment method in the global pollination phase to enhance the algorithm's global search capability, and introducing a new variation mechanism in the local pollination phase to increase the algorithm's local search capability. Experimental research and analysis results demonstrate that the improved Pollen-based Pollination Algorithm is effective in solving the studied MGFJSP problem.

Keywords: Multi-Objective Optimization, Flexible Job Shop Scheduling, Flower Pollination Algorithm, green manufacturing, adaptive probability

1. Introduction

Under the environment of global energy efficiency, greenhouse gas reduction, and green manufacturing, reducing energy consumption and achieving green manufacturing has gradually become a hot issue of concern to manufacturing. Green workshop scheduling is an important part of achieving green manufacturing and has broad development prospects [1]. By rationally optimizing the sequence of process operations and the allocation of equipment and other resources, in order to achieve economic benefits, the goal is to minimize energy usage during the production process as much as possible. Based on the classic shop scheduling problem, Brucker et al. [2] removed the unique constraint on the processing equipment during the production process and analyzed the situation of multi-objective collaborative optimization. Tutumlua et al. [3] introduced a Hybrid Genetic Algorithm (HGA) and a Local Search Algorithm (LSA) into the Genetic Algorithm (GA) to optimize sub-lot sizes and enhance efficiency. Chiang et al. [4]

[1] Associate Prof., School of Naval Architecture and Ocean Engineering, Wuhan Technical College of Communications, China, e-mail: zhaojieyujun@163.com
[2] Prof., School of Naval Architecture and Ocean Engineering, Wuhan Technical College of Communications, China, e-mail: zr10162023@163.com

introduced an evolutionary algorithm to address the multi-objective flexible job shop scheduling problem (MFJSP) with three optimization objectives. Gao et al. [5] used minimization of fuzzy maximum completion time and minimization of maximum machine load as the optimization goals of MFJSP, which were solved by artificial bee colony algorithm. The work of Gu et al. [6] focused on developing an advanced competitive particle swarm optimization strategy for addressing the complexities inherent in many-objective optimization problems. For the MGFJSP problem, many researchers, both domestic and international, have undertaken comprehensive studies on this particular topic. Karabulu et al. [7] proposed an evolutionary strategy method to address the distributed blocking flowshop scheduling issue. San Jose-Revuelta et al. [8] addressed a novel flower pollination algorithm (FPA) to achieve equalization in synchronous DS/CDMA multiuser communication systems. An innovative backtracking search algorithm was put forward by Caldeira et al. [9] to address the complexities of multi-objective scheduling optimization, incorporating considerations of new job arrivals and energy consumption. Mansouri [10] examined the correlation between energy consumption and maximum completion time in relation to MFJSP. Destouet et al. [11] introduced the low-carbon strategy for tackling the problem. Researchers have made significant progress in the field of green manufacturing and workshop scheduling optimization, providing important theoretical and methodological support for achieving green manufacturing and energy efficiency. These research achievements offer valuable guidance and insights for the sustainable development of the manufacturing industry in the global energy environment.

The MGFJSP presents a higher level of complexity compared to the classic MFJSP. Finding a solution for this problem is more challenging, and the research in this area holds greater practical significance and application value. However, after thoroughly analyzing the current research status and shortcomings in multi-objective scheduling for flexible job shops, we propose the low-carbon strategy of equipment status - energy consumption curve, as well as multi-objective flexibility including energy consumption, maximum completion time, processing cost and cost-weighted processing quality. The job scheduling problem in the model is a critical issue that must be urgently solved. Based on the MFJSP, this paper takes the minimization of total energy consumption as green indicator and minimization of the maximum completion time as the economic indicator and develops the MGFJSP that considers the green indicator and the economic indicator collaboratively. According to the characteristics of the MGFJSP problem, the development of an Improved Flower Pollination Algorithm (IFPA) aims to solve this problem, and its feasibility and effectiveness are verified through dedicated experimental investigations.

2. Problem description and mathematical model

In the context of this research, MGFJSP under investigation can be described in the following manner. Supposing there are n workpieces and m processing machines, each workpiece has multiple processing procedures, and each procedure can select a legal machine for processing, and the speed of the machine processing can be selected. It is assumed that the following constraints must also be met during processing:

(1) Once the sequence of processing steps of each workpiece is determined, it cannot be changed.

(2) Each process can select any machine within the legal range for processing.

(3) Each machine is restricted to processing a single task at a time.

(4) Once the processing speed of the machine is determined, it cannot be changed temporarily.

(5) After the process starts, no interruption is allowed.

In the actual processing process, choosing a higher machine speed led to reduced processing time at the expense of increased energy consumption. The duration of processing time and the amount of energy consumption will directly impact the economic benefits of the company. In this paper, the primary objectives are to minimize the maximum completion time and reduce the machine total energy consumption. On the premise of satisfying the relevant constraints, the optimization model is formulated as follows:

The problem's objective function is:

$$f = \min\,(C_{max}, Q) \tag{1}$$

where $C_{\max}$ is the maximum completion time, and Q is the total energy consumption of machine processing.

$$C_{max} = \max_{i=1}^{\mathrm{n}} E_{in_i} \tag{2}$$

Equation (2) presents the calculation method of the maximum completion time C_{max}. Here, i represents the workpiece number, n_i stands for the total number of processes of workpiece i, and $\boldsymbol{E_{in_i}}$ is the processing end time of the n_i-th process of workpiece i, which is also the processing completion time of workpiece i. The maximum completion time C_{max} is the maximum value among the processing completion times of all workpieces.

$$Q = Q_s + Q_o \tag{3}$$

$$Q_s = \sum_{k=1}^{m} C_k\, p_k^s \tag{4}$$

$$Q_o = \sum_{k=1}^{m} \sum_{i=1}^{n} \sum_{j=1}^{n_1} Q_{kv}^{ij} \tag{5}$$

Equation (3) indicates that the total energy consumption of the machine is composed of the machine's total standby energy consumption Q_s and the machine's total processing energy consumption Q_o.

Equation (4) is the calculation method of the total standby energy consumption, where C_k is the completion time of machine k's processing, p_k^s is the standby energy consumption of machine k.

Meanwhile, Equation (5) presents the calculation method of the machine's total processing energy consumption. Here, Q_{kv}^{ij} represents the processing energy consumption of machine j when processing the j-th process of workpiece i at speed v, and Qo is the sum of the energy consumption of each process during the machine's processing.

3. IFPA for MGFJSP

To solve the MFJSP mainly through the intelligent optimization algorithm, GA is the intelligent optimization algorithm that is most frequently utilized. However, GA suffers from the drawbacks of slow convergence and premature convergence. As a result, it is imperative to identify an alternative approach for addressing MGFJSP more effectively.

The FPA is an innovative meta-heuristic swarm intelligence optimization algorithm that replicates the process of flower pollination [12]. This algorithm possesses the benefits of a straightforward structure and a robust algorithmic search capability. However, the conventional FPA is primarily employed for addressing continuous optimization problems, whereas this article tackles MGFJSP, which is a representative discrete optimization problem. Considering its advantages, this paper designed the IFPA for solving MGFJSP.

First, the algorithm is transformed into a discrete form, and an encoding technique suitable for the characteristics of MGFJSP is provided. The relevant operations within the algorithm are then defined discretely, enabling the algorithm to effectively handle discrete optimization problems; second, a design is developed using a hybrid initialization strategy aimed at enhancing the quality of the initial solution. An adaptive adjustment method is integrated into the global pollination stage to prevent premature convergence to local optima. To further promote population diversity, the local pollination process of the algorithm incorporates a unique mutation mechanism. Finally, the fixed switching probability in the classic FPA is designed as an adaptive switching probability, which effectively balances and improves between global search and local search capabilities. The main steps of IFPA shown in Fig 1 are as follows:

Step 1: Perform parameter settings related to the algorithm, initialize the pollen population, and evaluate the initialized individuals;

Step 2: Randomly generate a random number rand greater than 0 and less than 1, compare the current switching probability p with rand, if p >rand, continue to step 3; if not, proceed to step 4;

Step 3: Perform global pollination operations;

Step 4: Perform local pollination operations;

Step 5: Re-evaluate the population and increase the current iteration number by one;

Step 6: Verify whether the current iteration count has reached the maximum limit. If the maximum iteration limit is achieved, continue to step 7; if not, proceed to step 2;

Step 7: Output the global optimal solution;

Step 8: End.

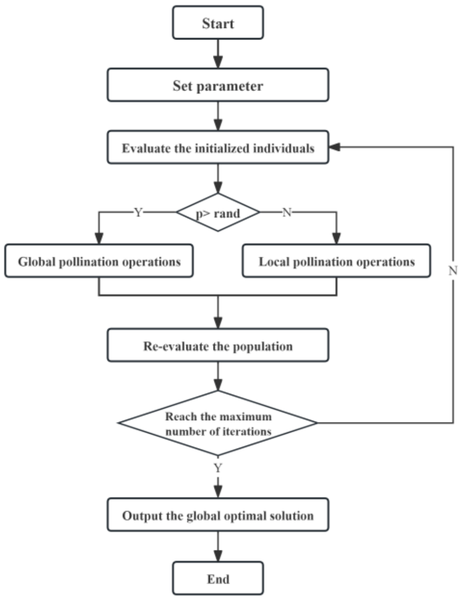

Fig. 1. The flowchart of the IFPA.

3.1. Encoding and decoding.

Taking into account the attributes of the MGFJSP addressed in this study, corresponding to the three subproblems of process selection, machine selection, and speed selection, the three-layer coding method is used.

As shown in Fig 2, OV represents the process scheduling vector. In this vector, the corresponding number represents the corresponding index of the workpiece number, and the frequency of the index number indicates the amount of operations performed on the workpiece. For instance, the fourth number "1" in the OV vector is the second occurrence, indicating that it corresponds to the second operation of the workpiece 1, named as O_{12}.

MV represents the machine arrangement vector. In this vector, the corresponding number represents the index number of the corresponding machine for the process.

SV represents the speed arrangement vector. In this vector, the corresponding number is the index number of the speed gear selected by the machine corresponding to the process.

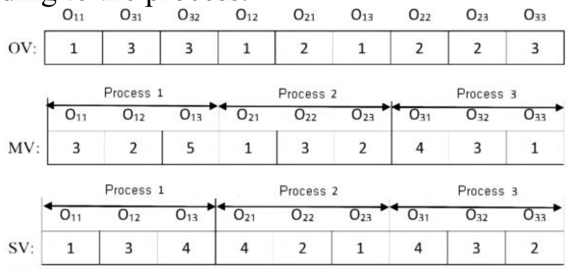

Fig. 2. Coding diagram.

3.2. Initial pollen.

Within this research, the method of strategy selection is employed to generate the initial solution, that is, the workpieces are arranged randomly based on the optimization objective. When selecting the legal machine and its corresponding processing speed, taking into account the processing time factor, the machine with the shortest processing time is given priority in selection as much as possible.

3.3. IFPA Discrete Definition.

Considering that the MGFJSP solved in this paper is a typical discrete optimization problem, the traditional global pollination and local pollination operations are not applicable [13]. Therefore, the relevant operations need to be redefined to make it suitable for the discrete problem-solving process. In this paper, the definition of addition and subtraction in classic FPA is discretized as follows:

a) Addition operation

This article uses two-point intersection to redefine the addition operation [14], that is, that is, two points are randomly selected in the parent's *OV* vector for intersection, but the selected machine gear and machine speed gear under the corresponding process will not change to get the individual child. If the child is better than the parent, replace the parent with the child, as shown in Fig 3.

b) Subtraction operation

The subtraction operation in this paper adopts the crossover method of preferential process crossing [15]. The specific operation method is described as follows:

First, the workpiece set is randomly divided into two non-empty complementary sets *A1* and *A2*. Then, select the index numbers included in the set *A1* from parent 1, and copy them to child 1 in the order of the original position.

Next, select the index numbers included in the set *A2* from parent 2, keep the original position of the process, and copy the content to child 2. After that, insert the process index numbers of the set *A2* in parent 2 into the blanks of child 1 in order, so that a complete child 1 is obtained.

Similarly, insert the content of set *A1* included in parent 1 into the blanks of child 2 in order to get a new child 2. If the child is better than the parent, substitute the parent with the child. The subtraction operation is visually depicted in a schematic diagram provided in this paper shown in Fig. 4.

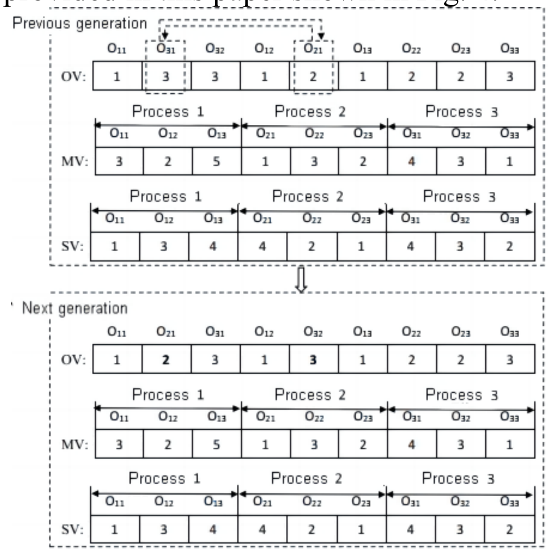

Fig. 3. Addition operation.

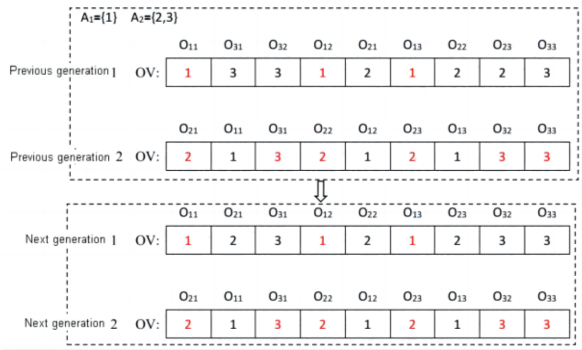

Fig. 4. Subtraction operation.

3.4. Global pollination operation.

The global pollination operation of the classic FPA algorithm is mainly to update the operation by relying on the update formula of global pollination by external organisms. While the update operation of pollination is discretely defined, an adaptive adjustment method is introduced during the global pollination stage. The update formula of the global pollination operation after the update is shown in Equation 6.

$$x_i' = x_i + L(x_i - g_{best}) + \delta(x_j - x_k) \tag{6}$$

In Equation (6), x_i and x_i' respectively represent the pollen individuals before and after the update, x_j and x_k are two randomly selected pollen individuals of x_i, which g_{best} is the globally optimal individuals in the current population, and L is the random movement of pollen used to characterize the pollination intensity. The step size is subject to levy distribution. In this paper, the Mantegna algorithm is used to simulate the levy distribution result, as shown in Equation (7).

$$L \approx \frac{\mu}{|v|^{1/\lambda}} S_o \tag{7}$$

In Equation (7), S_0 is the minimum step size, usually taking $S_0 = 0.01$, μ and v satisfy the normal distribution, as shown in Equation (8),

$$\begin{cases} \mu \sim N(0, \sigma_\mu^2) \\ v \sim N(0,1) \\ \sigma_\mu = \left(\frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma((1+\beta)/2)\beta 2^{(\beta-1)/2}} \right)^{1/\beta} \end{cases} \tag{8}$$

δ is the adaptive adjustment probability introduced in this paper. Equation (9) provides a depiction of the calculation method, where *Gen* defines the maximum number of iterations, *iter* is the current number of iterations, and $p1$ and $p2$ are the given two adjustment probabilities. It can be seen from Equation (9) that δ also changes with the current iteration number *iter*, realizing the process of dynamic adaptive adjustment.

$$\delta = p_1 - \frac{p_2 - p_1}{Gen} iter \tag{9}$$

3.5. Local pollination operation.

The local pollination operation of the classic FPA algorithm relies on the update formula by non-biological self-pollination to update the algorithm to improve the local search performance.

$$\boldsymbol{x}_i' = \begin{cases} \boldsymbol{x_i} + \boldsymbol{x_j} - \boldsymbol{x_k} & (r > rand) \\ \boldsymbol{x_i} + \boldsymbol{x_j} & (other) \end{cases} \tag{10}$$

In Equation (10), x_i and x_i' represent the pollen individuals before and after the update respectively, x_j and x_k are two randomly selected pollen individuals different from x_i, r is a random number within the interval of (0, 1).

To enhance the algorithm's local search capability, this article further introduces the mutation mechanism of *MV* and *SV* building upon the basic local pollination operation.

The process of the specific mutation mechanism is as follows:

Equation (10) will be updated according to certain rules and the result will be obtained. The individual performs the mutation operation of *MV* and *SV*. Specifically, it keeps the individual process vector *OV* unchanged. Then, it randomly selects a process in *OV* and randomly selects the processing machine and the speed gear of the selected process within the legal range so as to obtain the mutation. Subsequently, if the new individual is superior to the original one, then x_i' is replaced by x_{new}. The mutation mechanism of *MV* and *SV* is shown in Fig. 5.

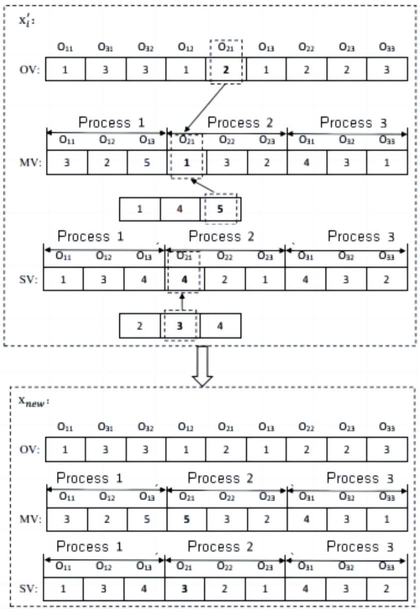

Fig. 5. Variation mechanism of *MV* and *SV*.

3.6. Adaptively adjusted handover probability.

The natural pollination methods are mainly divided into global pollination and local pollination. In the classic FPA algorithm, the switching between these two methods is achieved through the parameter $p \in (0, 1)$. But the normal switching threshold p is a fixed value. Considering that the size of the p value will directly impact the solution performance of the algorithm, if the p value is large, the algorithm will focus on the global pollination operation, and the algorithm's local search ability is weak and the convergence ability is poor; when the value of parameter p is relatively small, the algorithm will prioritize the local pollination operation, increasing the likelihood of the algorithm getting trapped in local optima.

Therefore, in IFPA of this paper, the adaptive adjustment probability is used as the switching threshold of the algorithm to balance the global search ability and local search ability. The calculation method of the adaptively adjusted switching probability of IFPA is shown in Equation (11). As shown in Equation (11), where *iter* is the current number of iterations and *Gen* is the maximum number of iterations, the adaptively adjusted switching probability p will change with the change of the current number of iterations *iter*. At the beginning of the iteration, the algorithm focuses on global search, and as the number of iterations increases, the algorithm will shift its focus to local search. This allows the IFPA algorithm to strike a balance between global search capability and local search capability, better adapting to different problems and search stages, and improving the algorithm's performance and convergence speed.

$$\mathbf{p} = \frac{Gen-iter}{Gen} \tag{11}$$

4. Calculation experiment

This paper carried out related calculation experiments, and the experimental environment was as follows: a 2.50-GHz PC, 16GB RAM, and a Windows 10, 64-bit operating system personal computer.

4.1. Selection of test examples and evaluation indicators.

To evaluate the performance of IFPA in solving MGFJSP, fifteen examples of *MK1-MK10* in Brandimarte's standard example [16] were used for calculation experiments. This is because these examples are widely recognized and can provide a reliable basis for evaluating the performance of the algorithm. Considering that the mathematical model in this paper takes into account the green index of the total machine energy consumption, the test data needs to be expanded accordingly. Moreover, considering that the standard test example requires fifteen machines, this paper selects machines with processing power and standby power from [17].

It is necessary to select appropriate metrics to judge the algorithm performance, considering the focus of this article. Inverse generation distance (*IGD*) and error rate (*ER*) [18] are selected as the evaluation indexes of the algorithm.

(1) Inverse Generational Distance

IGD represents the distance between the algorithm's approximate Pareto frontier and the actual Pareto frontier. A smaller value of *IGD* indicates better convergence and diversity of the algorithm's approximate Pareto frontier, implying that it is closer to the actual Pareto frontier. The specific calculation method of *IGD* is shown in Equation (12), where P is the real Pareto frontier, δ is the set of all feasible solutions obtained by the algorithm, and $\boldsymbol{f_m^{max}}$ is the maximum value of the *m-th* objective function. Similarly, $\boldsymbol{f_m^{min}}$ is the minimum value of the *m-th* objective function.

$$\mathbf{GD} = \frac{1}{|P|}\sum_{i=1}^{|P|} \min_{j=1}^{|\delta|} \sqrt{\sum_{m=1}^{M}\left(\frac{f_m(p_i)-f_m(a_j)}{f_m^{max}-f_m^{min}}\right)^2} \tag{12}$$

(2) Error Ratio

R is employed to denote the ratio of non-dominated solutions (which are obtained by the algorithm) that are not part of the actual Pareto frontier among all the non-dominated solutions. A smaller *ER* indicates a higher number of non-dominated solutions provided by the algorithm that are part of the actual Pareto frontier, indicating better algorithm performance. The *ER* calculation method is illustrated in Equation (13). In Equation (13), Q denotes the number of non-dominated Pareto solutions obtained by the algorithm, and q is the number of non-dominated Pareto solutions in Q that are not part of the true Pareto frontier.

$$\mathrm{ER} = \frac{q}{Q} \tag{13}$$

4.2. Parameter setting.

There are three main parameters that affect the performance of IFPA algorithm: population size *Popsize*; iteration number *Gen*; archive set size *AS*. In order to find the parameter value that makes the algorithm run best, this paper uses Design of experiment (DOE) to carry out the parameter level test. Since the test takes into account these three factors, namely *Popsize*, *Gen* and *AS*, this paper designed five levels for each factor. The corresponding 3-factor 5-level factor level table is shown in Table (1).

The standard orthogonal table is adopted to design the Experiments (DOE) in this paper, and *MK5* is chosen as the DOE test example. The experimental result *RA* is determined by considering the number of non-dominated solutions in the reference set for each parameter combination. The better the results obtained by running under the combination of parameter levels are, the more preferable the parameter combination is. The orthogonal test table obtained by the test is shown in Table (2).

Table 1

Factor level table

factor / Level	Popsize	Gen	AS
1	500	30	50
2	400	25	40
3	300	20	30
4	200	15	20
5	100	10	10

Table 2

Orthogonal test table

factor / Test	Popsize	Gen	AS	RA
1	500	30	50	0.891
2	500	25	40	0.842
3	500	20	30	0.887
4	500	15	20	0.952
5	500	10	10	0.781
6	400	30	40	0.798
7	400	25	30	0.925
8	400	20	20	0.718
9	400	15	10	0.752
10	400	10	50	0.892
11	300	30	30	0.821
12	300	25	20	0.798
13	300	20	10	0.724
14	300	15	50	0.813
15	300	10	40	0.676

16	200	30	20	0.599
17	200	25	10	0.614
18	200	20	50	0.758
19	200	15	40	0.684
20	200	10	30	0.617
21	100	30	10	0.547
22	100	25	50	0.414
23	100	20	40	0.503
24	100	15	30	0.561
25	100	10	20	0.513
$\bar{I}$	0.8706	0.7312	0.7536	
$\overline{II}$	0.8170	0.7186	0.7006	
$\overline{III}$	0.7664	0.7180	0.7622	
$\overline{IV}$	0.6544	0.7524	0.7160	
$\bar{V}$	0.5076	0.6958	0.6836	

The fifth column in Table (2) is the value of *RA* obtained under the corresponding parameter level combination. Lines 26-30 are the average value of the *RA* result under the corresponding parameter level under each factor. As a demonstration, the value in the second column of line 27 “0.8170” means that the average value of the *RA* result obtained when *Popsize* = 400 is 0.8170.

In order to observe the internal law between the experimental results in Table (2) more intuitively and quickly obtain the optimal level parameter combination of factors, the average trend graph under each factor based on the data in Table (2) has been plotted, as shown in Figs. 6 - 8.

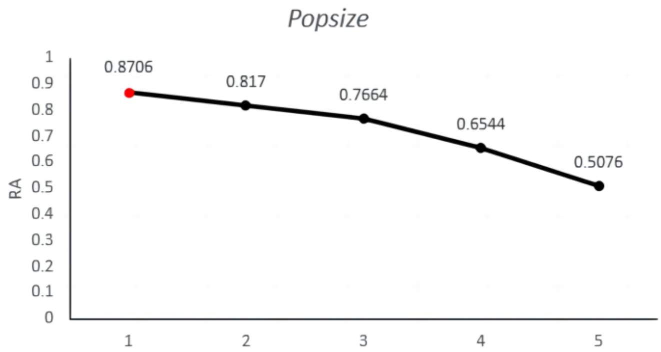

Fig. 6. Trend chart of the average values of *Popsize*.

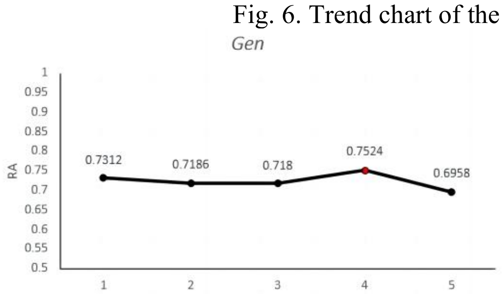

Fig.7. Trend chart of the average values of *Gen*.

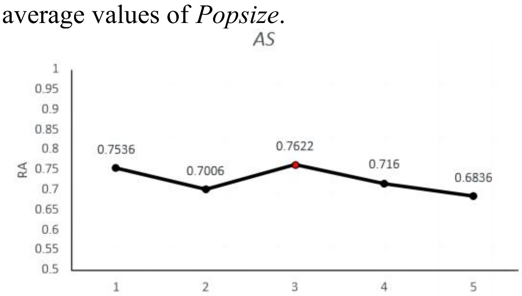

Fig. 8. Trend chart of the average values of *AS*

The results in Figs. 6 - 8 show that when *Popsize* selects the first level, *Gen* selects the fourth level, and *AS* selects the third level, the algorithm has the best experimental effect. That is, when conducting algorithm experiments in this paper, *Popsize* = 50, *Gen* = 15, *AS* = 30.

4.3. Discussion of results.

In this paper, we chose hybrid differential evolution [19], improved discrete particle swarm optimization algorithm [20] and improved non-dominated sorting *GA* as the comparison algorithm to verify the performance of IFPA. Each algorithm will be independently run 30 times in each set of extended test examples, and the average of these 30 run results will be considered as the final test result. Table (3) demonstrates the comparative findings from the algorithm's final examination.

Table 3

Algorithm test results

Count	IFPA		HDE		DPSO		NSGA2	
	IGD	ER	IGD	ER	IGD	ER	IGD	ER
MK1	0.12	0.02	1.98	1.00	1.01	0.78	1.82	0.92
MK2	0.08	0.04	1.84	0.95	0.49	0.62	1.49	1.04
MK3	0.04	0.01	1.21	0.96	0.38	0.71	1.00	0.98
MK4	0.04	0.00	1.41	0.99	0.72	0.69	1.76	1.00
MK5	0.12	0.02	0.79	1.00	0.59	0.82	0.98	0.79
MK6	0.01	0.00	0.82	1.00	0.62	0.75	1.02	0.87
MK7	0.05	0.00	1.25	1.00	0.71	0.61	1.13	1.00
MK8	0.11	0.02	0.82	0.97	0.74	0.67	0.89	0.91
MK9	0.09	0.01	0.75	0.99	0.89	0.81	0.91	0.87
MK10	0.05	0.03	0.82	0.96	0.52	0.78	0.63	0.91

The experimental outcomes of the comparative algorithm, as presented in Table (3), demonstrate that the IFPA test results outperform the other three comparison algorithms significantly in terms of both evaluation indicators *IGD* and *ER*. To visually depict the data concentration, the data from Table (3) was graphically represented using boxplots, depicted in Fig. 9-10.

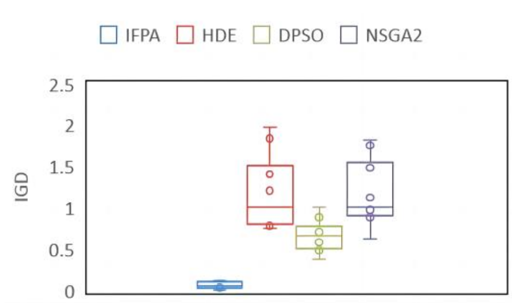

Fig. 9. Box line chart of *IGD*.

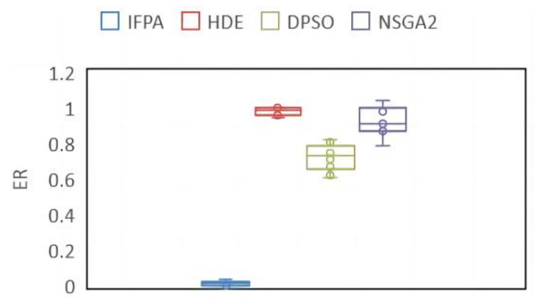

Fig. 10. Box line chart of *ER*

Apparently, as can be seen from Figs. 9 - 10, for *IGD* and *ER* indicators, IFPA has obtained the optimal median, upper limit and lower limit, and the ranges corresponding to IFPA are the smallest, indicating the best concentration of data.

In summary, the IFPA algorithm proposed in this study exhibits better convergence and diversity of Pareto solutions than the other three algorithms within the same test time. In other words, the feasibility and effectiveness of IFPA are verified.

5. Conclusion

The study of MFJSP pondering green indicators has significant practical implications. The results of research on MFJSP are crucial for factory production efficiency and resource utilization. By addressing MFJSP, job scheduling can be optimized to enhance production efficiency, reduce production time and costs, thereby increasing the competitiveness of the factory.

a) In this article, an IFPA for MGFJSP is designed. Under the premise of satisfying the relevant constraints, the optimization aims to minimize both the total energy consumption and the maximum completion time.

b) The relevant operations in the FPA are redefined for the discretization problem in this paper, and related encoding and decoding schemes are designed. To further improve the algorithm performance, this article updates the global pollination according to the addition and subtraction operations defined by the discretization. The operation is discretely defined, and an adaptive adjustment method is introduced during the global pollination process to boost the global search capability; the MV and SV mutation mechanisms are introduced during the local pollination stage to increase the local search capability.

c) The algorithm uses adaptive adjustment probability instead of the fixed switching threshold in the past to balance the ability of global search and local search.

Through experimental verification, the feasibility and efficiency of IFPA were demonstrated. The contribution of the IFPA algorithm in MGFJSP lies in its ability to effectively handle multiple optimization objectives, such as minimizing production time and maximizing resource utilization. By balancing the trade-offs between multiple objectives, the IFPA algorithm can generate a set of high-quality scheduling solutions, assisting factory managers in making more informed decisions. The application of the IFPA algorithm can effectively optimize production planning, improve production efficiency, reduce production costs, and perform well in addressing multi-objective optimization problems.

In the future, research on MGFJSP pondering green indicators may have the following development directions.

a) Sustainability and Environmental Protection. With the growing societal concern for environmental issues, green production is becoming a significant trend in manufacturing. Future research will focus on optimizing resource utilization, reducing waste and emissions, etc., to achieve sustainable development.

b) Refinement and Expansion of Green Indicators. Current research mainly focuses on basic green indicators. Other environmental impact factors, such as carbon emissions and waste generation, can be further refined and expanded to more comprehensively assess the environmental sustainability of factories.

c) Research on Multi-objective Optimization Algorithms. Research on more advanced solutions to handle multiple goals simultaneously, such as shortening production cycle, reducing cost, and reducing energy consumption.

d) Dynamic Scheduling Strategies. Research on real-time responses to unforeseen occurrences during the production process, such as equipment breakdowns or order modifications.

e) Integration of Intelligent Technologies. The integration of technologies such as artificial intelligence and machine learning enables the incorporation of intelligent and adaptive scheduling solutions into MGFJSP. For example, more accurate scheduling decisions can be achieved through predictive and optimization models.

In summary, these research directions provide many challenges and opportunities for improving production efficiency, decreasing energy consumption and reducing environmental repercussions and promoting sustainable development of factories.

Acknowledgments

The authors acknowledge the financial supported by Hubei Provincial Natural Science Foundation of China (Grant No.2022CFB331).

REFERENCES

[1] S. Afsar, J. J. Palacios, J. Puente, C. R. Vela, and I. Gonzalez-Rodriguez, Multi-objective enhanced memetic algorithm for green job shop scheduling with uncertain times, Swarm and Evolutionary Computation, vol. 68, p. 101016, 2022.

[2] P. Brucker and R. Schlie, Job-shop scheduling with multipurpose machines, Computing, 1990.

[3] B. Tutumlu and T. Sara¸c, A MIP model and a hybrid genetic algorithm for flexible job shop scheduling problem with job-splitting, Computers & Operations Research, vol. 155, p. 106222, 2023.

[4] T.-C. Chiang and H.-J. Lin, A simple and effective evolutionary algorithm for multi-objective flexible job shop scheduling, International journal of production economics, vol. 141, no. 1, pp. 87–98, 2013.

[5] K. Z. Gao, P. N. Suganthan, Q. K. Pan, T. J. Chua, C. S. Chong, and T. X. Cai, An improved artificial bee colony algorithm for flexible job-shop scheduling problem with fuzzy processing time, Expert Systems with Applications, vol. 65, pp. 52–67, 2016.

[6] Q. Gu, Y. Liu, L. Chen, and N. Xiong, An improved competitive particle swarm optimization for many-objective optimization problems, Expert Systems with Applications, vol. 189, p. 116118, 2022.
[7] K. Karabulut, D. Kizilay, M. F. Tasgetiren, L. Gao, and L. Kandiller, An evolution strategy approach for the distributed blocking flowshop scheduling problem, Computers & Industrial Engineering, vol. 163, p. 107832, 2022.
[8] L. M. San-Jose-Revuelta and P. Casaseca-de-la Higuera, A new flower pollination algorithm for equalization in synchronous DS/CDMA multiuser communication systems," Soft Computing, vol. 24, no. 17, pp. 13069–13083, 2020.
[9] R. H. Caldeira, A. Gnanavelbabu, and T. Vaidyanathan, An effective backtracking search algorithm for multi-objective flexible job shop scheduling considering new job arrivals and energy consumption, Computers & Industrial Engineering, vol. 149, p. 106863, 2020.
[10] S. Afshin Mansouri and E. Aktas, Minimizing energy consumption and makespan in a two machine flowshop scheduling problem, Journal of the Operational Research Society, vol. 67, no. 11, pp. 1382–1394, 2016.
[11] C. Destouet, H. Tlahig, B. Bettayeb, and B. Mazari, Flexible job shop scheduling problem under industry 5.0: A survey on human reintegration, environmental consideration and resilience improvement, Journal of Manufacturing Systems, vol. 67, pp. 155–173, 2023.
[12] X.-S. Yang, Flower pollination algorithm for global optimization, in international conference on unconventional computing and natural computation, pp. 240–249, Springer, 2012.
[13] S.-y. Ding, B. Li, and H.-b. Shi, Study on flexible job-shop scheduling problem based on improved discrete particle swarm optimization algorithm, Computer Science, vol. 4, pp. 233–240, 2018.
[14] X. Huang, S. Chen, T. Zhou, and Y. Sun, Survey on genetic algorithms for solving flexible job shop scheduling problem, Computer Integrated Manufacturing Systems, vol. 2, pp. 536–551, 2022.
[15] O. Abdel-Raouf, I. El-Henawy, and M. Abdel-Baset, A novel hybrid flower pollination algorithm with chaotic harmony search for solving sudoku puzzles, International Journal of Modern Education and Computer Science, vol. 6, no. 3, p. 38, 2014.
[16] A. Amirteimoori and R. Kia, Concurrent scheduling of jobs and AGVs in a flexible job shop system: a parallel hybrid PSO-GA meta-heuristic, Flexible Services and Manufacturing Journal, vol. 35, no. 3, pp. 727–753, 2023.
[17] Y. Liu, Research on many-objective green flexible job shop scheduling problem, in Chengdou: Southwest Jiaotong University, 2011.
[18] H. Liu and F. Li, Self-adaptive differential evolution algorithm for multi-objective optimization, Computer Applications and Software, vol. 12, pp. 1249–252, 2015.
[19] H. T. Rauf, W. H. K. Bangyal, and M. I. Lali, An adaptive hybrid differential evolution algorithm for continuous optimization and classification problems, Neural Computing and Applications, vol. 33, no. 17, pp. 10841–10867, 2021.
[20] S. Aminbakhsh and R. Sonmez, Discrete particle swarm optimization method for the largescale discrete time–cost trade-off problem, Expert systems with applications, vol. 51, pp. 177–185, 2016.