

Flexible Job Shop Scheduling Problem Based on Multi-Objective Optimization Algorithm

Li Zhang^a, Lu Wang^{b,*}

College of Information Science and Engineering, Shandong Agricultural University, Tai' an,
Shandong 271018, China;

^a 2982002568@qq.com, ^bwangl@sdaau.edu.cn

Keywords: Flexible job shop scheduling, multi-objective optimization, genetic algorithm, energy consumption.

Abstract. Procedures of the flexible job shop scheduling problems with a workpiece can be machined with different processing time, and not necessarily equal, taking into account the green low carbon emissions is an urgent need to solve manufacturing problems, construct a to minimize the maximum completion time and the maximum energy consumption minimization model, an improved non-dominated sorting evolution algorithm is proposed (Modified non-dominated sorting genetic algorithms, MNSGA). The algorithm is mainly improved crossover and mutation in genetic manipulation to improve the global search ability of the algorithm and to prevent the emergence of local optimal solution. Finally, a set of test data is used to compare the NSGA- II algorithm and the improved multi-objective optimization algorithm MNSGA. The results show that the improved Pareto algorithm has better performance than the NSGA- algorithm in the diversity and convergence of the solution.

1. Introduction

In practical engineering problems, especially in network design, we often deal with mathematical models more than one objective function, but multiple objective functions. Multi-objective is difficult to determine the search direction of the optimal solution. Multi objective optimization is to find the optimal target can satisfy all the solutions, this solution is usually not a solution, but by a set of solution set of multi-objective optimization, one of the objectives of the improvement may cause other target performance was reduced, so the multiple targets at the same time is almost optimal impossible, but the coordination and control of the target, it is possible to make all optimal [1].

In recent years, energy saving and emission reduction have been paid more and more attention to the world. At the same time, the problem of workshop scheduling is also faced with the dual solution of environment and economy [2]. Therefore, by minimizing the energy consumption produced in the production process, it can not only improve the economic efficiency of the enterprise, but also help to reduce the pollution to the environment [3].

In 2014, Liu [4] and others studied the economic and energy consumption targets of multi-objective scheduling problem, and proposed a mathematical model considering drag delay, carbon footprint and power, and verified it with the improved algorithm. In 2015, Xiong and Fu [5] proposed a multi Agent scheduling system based on immune algorithm and solved the flexible job shop scheduling problem aiming at minimizing makespan. The improved algorithm has been improved in terms of flexibility and robustness. In 2016, Ahmadi [6] proposed a multi-objective optimization method based on evolutionary algorithm for the flexible job shop scheduling problem under machine random failures. In the same year, Singh [7] proposed a new optimization algorithm for solving multi-objective flexible job shop scheduling problem with optimal Pareto frontiers.

2. Problem Description and Mathematical Modeling

2.1 Multi Objective Optimization Theory

First, a general description of the multi-objective optimization problem [8] is given, and the given decision vector $X=(x_1, x_2, \dots, x_n)$, it meets the following constraints:

$$g_i(X) \geq 0 (i=1, 2, \dots, k) \quad - \quad (1)$$

$$h_i(X) \geq 0 (i=1, 2, \dots, k) \quad - \quad (2)$$

With r conflicting optimizations, the optimization target can be expressed as:

$$f(X) = (f_1(X), f_2(X), \dots, f_r(X)) \quad -- \quad (3)$$

Seek $X^*=(x_1^*, x_2^*, \dots, x_n^*)$ so that $f(X^*)$ is optimal when the constraints (1) and (2) are satisfied.

The definition of non-dominated solution is: Assuming that any two solutions x_1 and x_2 are better than x_2 for all optimization objectives, then x_1 dominates x_2 . If x_1 is not dominated by other solutions, then x_1 for non-dominated solution (not dominated solution), also known as Pareto solution. The optimal solution may not necessarily be a single one. In the research process of multi-objective problems, we can't simply compare the solutions, and we can usually find the solution set that is better than any other solution [9]. Compared with the traditional method, the goal of a single goal is to increase the diversity of understanding and provide decision makers with a better choice.

In the multi-objective optimization problem, the Pareto optimal solution is to ensure that there is no inferior solution compared with any other solution. Although the Pareto optimal solution can't be said to be superior to any other solution, it does not exist any better solutions.

2.2 Flexible Job Shop Scheduling Problem Theory

Flexible job shop scheduling problem (FJSP) is a kind of resource allocation problem for task configuration and order constraint. The description of FJSP is as follows [10]:

n pieces of work, set expressed as: $J = \{J_1, J_2, \dots, J_i, \dots, J_n\} (i \in [1, n])$.

m machines, set expressed as: $M = \{M_1, M_2, \dots, M_j, \dots, M_m\} (j \in [1, m])$.

Each workpiece needs to go through several processes to complete the process, the number of processes for each workpiece can be different, and each process can choose to be processed on different machines [11]. Flexible job shop scheduling problem has more flexibility than the traditional shop scheduling problem.

2.3 Mathematical Modeling

(1) The maximum completion time is minimized

The maximum completion time is the processing of multiple workpieces at the same time, the time required for the end of all workpieces is finished, the shorter the time, indicating that the higher the efficiency in actual production, the mathematical expression is as follows:

$$f(1) = \min T_{\max} = \min \{ \max T_i | i \in n \} \quad (4)$$

$$T_i = runi + idlei \quad - \quad (5)$$

(4) Where: T_{\max} refers to the maximum completion time, T_i said the completion of the workpiece i , n represents the total number of the workpiece; equation (5) represents the total time machine i is equal to the processing time $runi$ and standby time $idlei$ Sum.

(2) Maximum energy consumption is minimized

In addition to the machine will produce energy in the process of processing, the standby time also produces the corresponding energy consumption.

$$E_i = P_i \times runi + PDi \times idlei \quad (6)$$

$$E = \sum_{i=1}^m E_i \quad (7)$$

$$f(2) = \min E \quad (8)$$

(6) Where: P_i machine i processing power, $runi$ machine i running time; (7) Where: PDi for the machine i standby power, $idlei$ machine i standby time, E_i for the machine i energy consumption, E is the sum of processing power and standby power of machine i , E is the total energy consumption of m machines; (8) is that the objective function of maximum energy consumption is set as the minimum,

energy consumption is related to the completion time, and time. The longer the energy consumption is bigger.

As shown in Table 1, the M10 machine has a power of 2.7 during operation and a power of 0.4 during standby. If the machining time of M1 is 1 and the standby time is 1, the total energy consumption is calculated as $2.7 \times 1 + 0.4 \times 1 = 3.1\text{kw}$.

Table 1 Machine running power meter

Machine Power	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
Processing power P_i	2	1.8	1.6	2.4	2.4	4.1	3.5	4.1	2.8	2.7
Standby power P_d	0.5	0.6	0.3	0.4	0.4	0.6	0.8	0.9	0.3	0.4

3. Improved Multi - Objective Optimization Algorithm MNSGA

NSGA-II is one of the most widely used multi-objective evolutionary algorithms currently. It has the advantages of fast running speed and good convergence of solution set. The MNSGA algorithm is based on the fast-non-dominated sorting algorithm in [12] for fast and non-dominated sorting of individuals in the search space. The specific operation procedure of MNSGA is as follows.

3.1 Encoding Method

FJSP need to solve two sub-problems: the machine selection problem and the process sequencing problem. In order to deal with these two sub-problems, the segmentation coding method is adopted here. Two L -dimensional (L is the total number of work processes) Processes and machines, where the first line represents the sorting of processes, each of the components in the vector is an integer not greater than the total number of workpieces, the second line represents the selection order of the machines, and each component in the vector is not larger than the machine. The total number of integers.

The coding of the chromosomes is shown in Figure 1, which is a fully flexible job shop scheduling problem for 3 machines with 3 workpieces. The first line is about the sorting of work orders, where 1 means work 1, 2 represent work 2, The first occurrence of the first row, 1, represents the first operation of workpiece 1, the second occurrence of 1 represents the second operation of workpiece 1, and the second row relates to the selection of machines, where the first element 2 represents The first process of workpiece 1 is machined on machine 2 and the second element 1 represents the first process of workpiece 3 machined on machine 1.

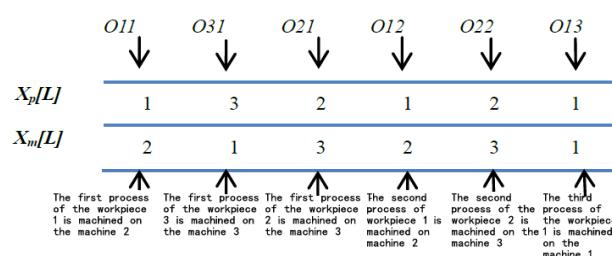


Figure 1 Encoding diagram

If the processing time is shown in Table 2, there are 3 workpieces in the table, 3 machines and 6 processes, the workpiece 1 has 3 processes, the workpiece 2 has one process, and the workpiece 3 has one process. In the table, the machines, workpieces and time correspond one by one.

Table 2 3 × 3 Part of the flexible job shop scheduling

Machine Workpiece	Process	M1	M2	M3
1	1	3	2	1
	2	1	3	4
	3	4	2	1
2	1	3	2	4
	2	2	1	2
3	1	5	4	1

The corresponding scheduling Gantt chart in Table 2 is shown in Fig.2. The machining order on the machine M1 is the second process of the workpiece 1, and the machining time is 1. On the machine M2, the machining sequence is the first process of the workpiece 2, Processing time is 2, the second process of workpiece 2, the processing time is 1; the machining sequence on the machine M3 is the first process of workpiece 1, the processing time is 1, the first process of workpiece 3 is processing time is 1, the third process of a workpiece, the processing time is 1. The figure C_{max} said the maximum completion time, the maximum completion time of 9.

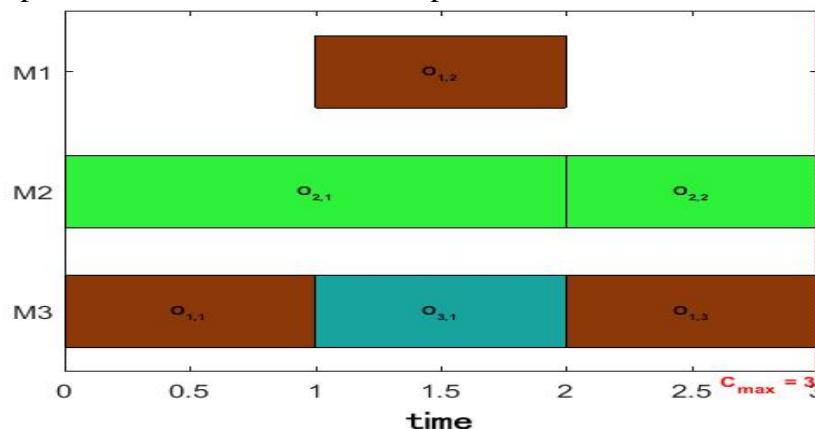


Figure 2 Table 2 Scheduling Gantt chart

3.2 Decoding Method

Decoding is to convert the chromosomes into corresponding time matrices, energy consumption matrices, machine matrices and corresponding machine vectors, and generates corresponding scheduling schemes according to the above transformations, so as to generate the corresponding scheduling Gantt charts according to the scheduling scheme.

3.3 Initialization

Initialize the parameters involved in the algorithm. According to the parameter setting, the chromosomes are initialized, the fitness of the initial chromosomes is solved, the chromosomes are sorted out of order, the local optimal chromosome solution sets and the global optimal chromosomes are initialized, and the global optimal chromosomes are saved.

3.4 Choose Operation

(1) The selection of fitness function: The selection of fitness function is very important for the solution of the problem. The specific operation method is to re-arrange the procedures according to the order, calculate the total processing time of the machine according to the start processing time and end time of the machine, The objective function to be reached is the two objective functions of the maximum completion time and the maximum energy consumption mentioned in section 2.3, and the corresponding chromosomes x_i are decoded one by one to obtain the total time vector C of the machine, the run time vector run , and the readout speed of each machine Processing power P_i and standby power Pd_i .

The maximum completion time of the first objective function is: $f_1(x_i) = \max(C_i)$, idle time vector of machine i idlei = $C_i - run_i$;

The maximum energy consumption of the second objective function is: $f_2(x_i) = \max(run_i \times P_i + idlei \times Pd_i)$.

The fitness function $f(x_i)$ corresponding to chromosome $x_i = [f_1(x_i) f_2(x_i)]$. The fitness function values corresponding to these two goals are calculated separately.

(2) Merging populations: The fitness values of each chromosome are calculated, and the merging of the populations is performed according to the individual fitness values. The number of individuals generated is the sum of the new and old individuals and is selected from the populations according to the crowding distance and grade values of the individuals, select the time, choose a small non-dominated level, if the same level, select the crowded distance. Then select the elite individuals to inherit into next generation population, and the number of individuals selected is the number of individuals of the original population.

3.5 Cross Operation

The choice is to pass on one of the more well-adapted chromosomes to the next generation while replacing the corresponding vector on the less-well-formed second half of the chromosomes with the well-fitted first half of the chromosomes. In the crossover operation, the latter half of the chromosomes are pairwise to be cross-linked as pairs of pairing factors. The resulting offspring will be compared with their parents, and the half of the high fitness will be selected to enter the next generation to update the position of the offspring.

$$y_1 = \alpha \cdot x_1^* + (1 - \alpha) \cdot x_2^* \quad (9)$$

$$y_2 = \alpha \cdot x_2^* + (1 - \alpha) \cdot x_1^* \quad - \quad (10)$$

The crossover operation is performed according to Eq. (9) and Eq. (10), where y_1, y_2 are the parent chromosomes generated, x_1, x_2 are the progeny chromosomes, α is the random number vector with the same dimension as x_1 , and each of the vectors the advantage of crossing $\alpha \in (0, 1)$ in this manner is that it inherits the good gene of the parent.

3.6 Mutation Operation

(1) Machining gene mutation operation, selecting a processing sequence gene of a workpiece, randomly generating a mutation probability, comparing the probability with the mutation probability set in advance, and if the probability is greater than the setting probability, performing a mutation operation. Otherwise, do not go ahead and plug it into a random place.

(2) The machine distributes the gene segment mutation operation, randomly selects a machine gene of the machine to assign a gene segment, compares the probability with a preset mutation probability, and if it is greater than the set probability, performs a mutation operation; otherwise, does not proceed. In the corresponding process of the gene processing machine focused on the selection of other machines to replace the gene.

3.7 Improved Multi-Objective Optimization Algorithm MNSGA Process

The overall calculation steps for solving multi-objective optimization problems by using MNSGA algorithm are as follows:

Step 0: Parameter setting. Including population size, crossover probability, mutation probability, maximum number of iterations, initial population

Step 1: Generate chromosomes (codes) of the initial solution to the constraints that satisfy the multi-objective optimization problem by using the piece-wise encoding method.

Step 2: Determine whether to generate a new generation of population. If yes, then calculate the objective function $f(x)$ of the multi-objective optimization problem according to the chromosome and convert the chromosome into an ordered processing operation table, which is decoding; otherwise, Sort, carry out genetic manipulation such as crossover mutation to form a new parent population;

Step 3: Population evolution iteration plus 1;

Step 4: Carry out the merger of the parental and offspring populations;

Step 5: Determine whether a new population has been generated; if so, execute a crossover or mutation operation to form a new population; otherwise, sort the fast-non-dominated solution set, and calculate the congestion distance at the same time, calculate the fitness function, and select a high fitness of individuals;

Step 6: Determine whether the maximum number of iterations is reached, and if yes, save the Pareto optimal solution and output the optimal scheduling solution. Otherwise, go to step 4;

Step 7: The algorithm ends.

4. Experiments and Results Analysis

4.1 Experimental Design

(1) The algorithm parameters are set: the iteration number is 100, the pop number is 100, the crossover probability pc is 0.8 and the mutation probability pm is 0.2.

(2) Test Data: Figures 3 and 4 show the test data used. The data is a part of the flexible job shop scheduling problem consisting of 7 processes, 8 processes, and 24 workpieces. "NaN" means that this process can't process on this machine. 1 in the first column of the table represents the first workpiece, 2 in the first column of the table means the second workpiece, the first column of the second column represents the first process of the workpiece 1, the second process of the workpiece 1 represents the second process And so on.

operation	process	M1	M2	M3	M4	M5	M6	M7
1	1	NaN	16	NaN	16	NaN	15	NaN
	2	17	18	NaN	14	NaN	NaN	16
	3	20	NaN	20	NaN	18	13	NaN
	4	20	NaN	17	NaN	13	NaN	19
2	1	NaN	17	NaN	18	NaN	17	NaN
	2	17	NaN	18	NaN	22	NaN	NaN
	3	19	17	NaN	21	19	17	19
3	1	18	NaN	NaN	20	NaN	NaN	NaN
	2	18	NaN	17	23	NaN	19	NaN
	3	NaN	22	20	NaN	NaN	NaN	17
4	1	NaN	NaN	NaN	NaN	16	NaN	18
	2	NaN	NaN	NaN	25	20	21	NaN
5	1	19	NaN	18	NaN	16	NaN	NaN
	2	NaN	12	9	NaN	NaN	10	NaN
	3	NaN	NaN	NaN	10	13	NaN	8
	4	13	16	NaN	NaN	14	NaN	12
	5	20	NaN	NaN	19	NaN	22	NaN
6	1	10	9	11	9	NaN	9	10
	2	NaN	NaN	17	NaN	11	NaN	NaN
	3	14	NaN	NaN	10	NaN	9	NaN
	4	NaN	13	11	NaN	NaN	NaN	10
7	1	10	9	NaN	7	NaN	NaN	NaN
	2	NaN	12	NaN	NaN	15	NaN	13
	3	NaN	NaN	11	10	NaN	8	NaN
8	1	15	NaN	NaN	NaN	15	NaN	12
	2	NaN	NaN	6	NaN	9	13	NaN
	3	NaN						

Figure 3 7 × 8 Flexible job shop scheduling problem diagram

Run power Pi	operation	process	M1	M2	M3	M4	M5	M6	M7
1	1	NaN	1.7	NaN	1.3	NaN	1.1	NaN	
	2	1.2	2.9	NaN	2.1	NaN	NaN	1.6	
	3	1.9	NaN	1.8	NaN	1.4	2	NaN	
	4	3	NaN	1.9	NaN	2.5	NaN	2	
2	1	NaN	2.6	NaN	2.1	NaN	1.8	NaN	
	2	2.6	NaN	2.2	NaN	1.4	NaN	NaN	
	3	1.5	1.8	NaN	2.1	2.4	2.1	2.5	
3	1	2.2	NaN	NaN	1.6	NaN	NaN	NaN	
	2	1.7	NaN	2	2.2	NaN	1.9	NaN	
	3	NaN	2.1	2	NaN	NaN	NaN	1.4	
4	1	NaN	NaN	NaN	NaN	1.7	NaN	1.6	
	2	NaN	NaN	NaN	1.3	1.5	1.4	NaN	
5	1	2.1	NaN	2.2	NaN	2.5	NaN	NaN	
	2	NaN	2.6	2.2	NaN	NaN	2.1	NaN	
	3	NaN	NaN	NaN	1.9	1.7	NaN	2.2	
	4	2.7	2.2	NaN	NaN	1.5	NaN	1.9	
	5	1.6	NaN	NaN	2.3	NaN	2.4	NaN	
6	1	1.8	2.1	1.6	1.5	NaN	2.2	1.5	
	2	NaN	NaN	1.8	NaN	2.1	NaN	NaN	
	3	2.3	NaN	NaN	2.4	NaN	2.5	NaN	
	4	NaN	2.1	1.9	NaN	NaN	NaN	1.1	
7	1	1.9	1.6	NaN	2.4	NaN	NaN	NaN	
	2	NaN	1.5	NaN	NaN	1.7	NaN	1.6	
	3	NaN	NaN	2.1	2.4	NaN	2.2	NaN	
8	1	NaN	NaN	2.5	NaN	1.4	NaN	1.7	
	2	1.6	NaN	NaN	NaN	2.2	1.5	NaN	
	3	NaN	NaN	2.5	NaN	2.2	1.5	NaN	
Standby power Pdi			0.5	0.6	0.3	0.4	0.5	0.6	0.4

Figure 4 7 × 8 Flexible job shop scheduling energy diagram

4.2 Results Analysis

(1) The purpose of the experiment: The experiment is based on the flexible job shop scheduling and multi-objective characteristics, relying on the mathematical model of minimizing the maximum completion time and minimizing the maximum energy consumption, respectively NSGA- II algorithm and MNSGA improved algorithm experiments, experiments The result is a Gantt chart that gives a set of Pareto solutions, the best scheduling solution, and the best scheduling solution for statistics on maximum completion time and maximum energy consumption.

(2) Analysis Results: The results of MNSGA experiment on each test set are shown in Table 3. The first set of data in the Pareto solution set (72,741) indicates the maximum completion time is 72, the latter means the maximum energy consumption is 741, And so on, the experiment was carried out 100 times, 200 times, 300 iterations.

Table 3 MNSGA Pareto solution table

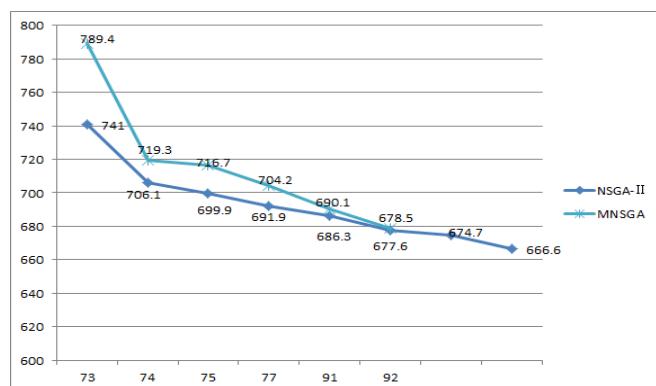
Iterations	Pareto solution set number	MNSGA algorithm Pareto solution set
		Pareto solution set
100 times	8	(72,741);(74,706.1);(77,699.9); (79,691.9);(80,686.3);(82,677.6); (85,674.7);(101,666.6)
200 times	5	(68,706.2);(69,703.5);(73,673.6); (74,672.4);(80,659.1)
300 times	9	(68,699.7);(73,696.4);(74,668.9); (75,667.8);(84,664.0);(86,658.2); (90,655.3);(97,654.0);(110,646.1)

NSGA-II algorithm in each test set on the experimental results obtained are shown in Table 4, the experiment iterations 100 times, 200 times, 300 times, the Pareto solution set number and Pareto solution set.

Table 4 NSGA-II Pareto solution table

Iterations	Pareto solution set number	NSGA-II algorithm Pareto solution set
		Pareto solution set
100 times	6	(73,730.0);(74,719.3);(75,716.7); (77,704.2);(91,678.5);(92,675.0)
200 times	5	(69,705.4);(70,692);(72,679.1); (77,673.9);(82,673.2);
300 times	4	(69,732.8);(71,704.7);(72,682.9); (77,672.8)

MNSGA and NSGA-II algorithm using the test data in Figure 3,4 Experimental data obtained Pareto solution set comparison experiment shown in Figure 5, the lower line of the figure shows the MNSGA Pareto solution set, a total of eight, The above cyan line represents the Pareto solution set obtained by NSGA-II, a total of six, by comparing the observation, we can see that the improved algorithm solution set more than the number of unimproved, while the maximum energy consumption is also lower than unmodified Astringency is also better.


Figure 5 NSGA-II and MNSGA non-dominated solution comparison chart

Different non-dominated solutions correspond to different scheduling schemes, so the optimal scheduling Gantt chart obtained is not the same. MNSGA algorithm using the test data to experiment one of the best scheduling program shown in Figure 6, the abscissa that the completion time, the ordinate that seven machines, from the figure you can see the different processes on different machines processing Situation, the same color represents the same workpiece, O1, 1 said the first step of the workpiece 1, and so on. The figure Cmax is the maximum completion time, the maximum completion time is 74.

NSGA-II algorithm using the test data to experiment on one of the best scheduling solution shown in Figure 7, the maximum completion time of 75.

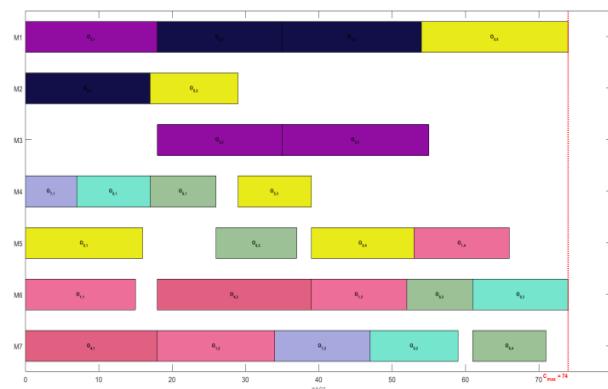


Figure 6 MNSGA algorithm optimal scheduling scheme

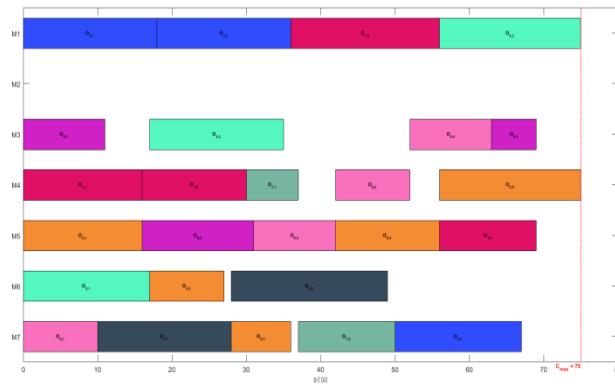


Figure 7 NSGA-II algorithm optimal scheduling scheme

5. Summary

In this paper, according to the characteristics of flexible job shop scheduling and multi-objective optimization, a mathematical model based on maximum completion time and maximum energy consumption is established. By minimizing the maximum completion time and minimizing the maximum energy consumption, under many constraints, Target optimization algorithm, the improved algorithm is applied to the flexible job shop scheduling application. Through the simulation experiments on the MATLAB platform, the optimal scheduling solution and Pareto solution set are provided, which provide a variety of choices for the decision makers. By comparing Experiments verify the effectiveness of the improved algorithm.

Acknowledgements

Fund Project: Major Research Plan of National Natural Fund (91746104); Key Cultivation Subject (Computer Science and Technology) Construction Project of Shandong Agricultural University.

References

- [1]. WU Dinghui, KONG Fei, TIAN Na, et al. Teaching and companion learning particle swarm optimization algorithm for multi-objective flexible job shop scheduling [J]. *Journal of Computer Applications*, 2015, 35 (6): 1617-1622.
- [2]. May, G., Taisch, M., Stahl, B., Sadr, V., 2013. Toward energy efficient industrial: a study on practices and viewpoint of the industry. *Advances in Production Management Systems*.
- [3]. Mouzon, G., Yildirim, M.B., Twomey, J., 2007. Operational methods for minimization of energy consumption of industrial equipment. *Int. J. Prod. Res.* 45 (18-19), 4247–4271.
- [4]. Liu, C.-H., Huang, D.-H., 2014. Reduction of power consumption and carbon footprints by applying multi-objective optimization via genetic algorithms. *Int. J. Prod. Res.* 52 (2), 337–352.

- [5]. Xiong, W., Fu, D., 2015. A new immune multi-agent system for the flexible job shop scheduling problem. *J. Intell. Ind.* 4, 17.
- [6]. Ahmadi, E., Zandieh, M., Farrokh, M., Emami, S.M., 2016. A multi-objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms. *Comput. Oper. Res.* 73, 56–66.
- [7]. Singh, M.R., Singh, M., Mahapatra, S.S., Jagadev, N., 2016. Particle swarm optimization algorithm embedded with maximum deviation theory for solving multi-objective flexible job shop scheduling problem. *Int. J. Adv. Ind. Technol.* 85, 2353–2366.
- [8]. Tao Wenhua, Liu Hongtao. Multi-objective Optimization Algorithm Based on Differential Evolution and NSGA-II [J]. *Computer Engineering*, 2016, 42 (11): 219-224.
- [9]. Wang Shujuan. Multi-objective dynamic and flexible job shop scheduling [D] .Shandong University, 2014.
- [10]. Liang Yan. Flexible manufacturing optimization based on hybrid evolutionary algorithm [D]. Dalian University of Technology, 2013.
- [11]. LI Congbo, SHEN Huan, LI Lingling, et al. Optimization scheduling model of multi-process flexible job shop oriented to energy consumption [J] .*Journal of Mechanical Engineering*, 2017, 53 (5): 12-23.
- [12]. Konak A, Coit D W, Smith A E. Multi-objective optimization using genetic algorithms: A tutorial [J]. *Reliability Engineering & System Safety*, 2006, 91(9):992-1007.