

PAPER • OPEN ACCESS

Research on JOB SHOP Scheduling Problem

To cite this article: Zi Yongsuo *et al* 2020 *J. Phys.: Conf. Ser.* **1607** 012050

View the [article online](#) for updates and enhancements.

You may also like

- [Hybrid immune genetic algorithm with neighborhood search operator for the Job Shop Scheduling Problem](#)
Yunpeng Lu, Zongnan Huang and Lian Cao
- [Solving Flexible Shop Scheduling Problems Based on Improved NSGA-II](#)
Xueqi Lan and Jianxin Zhang
- [A novel discrete PSO algorithm for solving job shop scheduling problem to minimize makespan](#)
K Rameshkumar and C Rajendran



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

Research on JOB SHOP Scheduling Problem

Zi Yongsuo¹, Zhang Caimeng¹, Hong Yinsheng¹, Qiu Pengrui¹, Shu Wangjiao¹

¹. Kunming Metallurgy College Kunming 650033

515425862@qq.com

Abstract: Aiming at the dynamic job shop scheduling problem with extended process constraints, this paper constructs a mathematical model of job shop scheduling problem with the objective of minimizing the maximum completion time. Genetic algorithm is used to solve the minimum maximum completion time, and the results are compared with those obtained by WSPT, EDD, FCFS, LPT and Cr. Through computer simulation, it is verified that the minimum maximum completion time obtained by genetic algorithm is better than that by SPT, EDD, FCFS, LPT and Cr.

1. Introduction

The job shop scheduling problem in the machining workshop has the practicality and scientificity of optimization goals in practical applications such as improving production efficiency, expanding process constraints, dynamic job shop scheduling, and minimizing the maximum completion time, therefore many literatures apply different mathematical model analysis methods to analyze and study JOB SHOP scheduling problem.

In this paper, five scheduling rules, WSPT, EDD, FCFS, LPT and Cr. Through computer simulation, the conclusion is drawn, which verifies that the minimum maximum completion time obtained by genetic algorithm is better than that obtained by using SPT, EDD, FCFS, LPT, Cr five scheduling rules, and the minimum maximum completion time obtained by genetic algorithm is not affected by the number of workpieces and processing time difference.

2. Job shop scheduling problem description and model construction

2.1 problem initialization description

Job shop scheduling problem ^[1-4] is a typical problem, which is the program epitome of many workshop operations.

The initialization of this problem can be simply described as following the processing of J products on M machines, and the processing process of each product needs to go through N processes, the processing start time of each product is s_{ij} , all processes of this task must be completed within time R, and the processing time of process i in the j process is t_{ij} .

2.2 data structure description of symbols

s_{ij} : Indicates that the workpiece No. i initializes the processing time point on the machine No. j

t_{ij} : Indicates that the workpiece No. i completes the whole process period on the machine No. j

e_{ij} : Indicates the machining end time point of workpiece No. i on machine No. j

x_{ijk} : It is to arrange the machining process of the workpiece No. i on the machine No. j and the machine No. k.



2.3 model construction

2.3.1 model assumptions.

1. Assumption of processing times: assume that each machine can only process each workpiece once;
2. Sequence assumption: each workpiece can be processed successively after the previous process is completed;
3. Real time hypothesis of time: each workpiece cannot be processed on two or more machines at the same time, and the same machine cannot process two or more products at the same time or with time intersection.
4. Assumption of no breakpoint in the processing: the start time of any process of each workpiece must be greater than or equal to zero, and the workpiece.
Once entering a certain process, it must be completed without interruption.
5. Assumption of no conflict in the model: assume that in all machining engineering, there is no conflict problem caused by process moving exchange.

2.3.2 construction. of job shop scheduling model. Generally speaking, the job shop scheduling problem can define the mathematical model of $n \mid m \mid c_{\max}$ as follows:

$$\min \sum_{i \in j} \max\{e_{ij}\}, i, j = 1, 2, \dots, n, m$$

$$\text{st. } s_{ij} \geq 0 \quad (1)$$

$$e_{ij} = s_{ij} + t_{ij} \quad (2)$$

$$(s_{i_1j} + t_{i_1j} - s_{i_2j})(s_{i_2j} - s_{i_1j}) \leq 0, t_{ij} = 0 \quad (3)$$

$$(s_{i_1j} + t_{i_1j} - s_{i_2j})(s_{i_2j} - s_{i_1j}) < 0, t_{ij} \neq 0 \quad (4)$$

$$(s_{ij_1} + t_{ij_1} - s_{ij_2})(s_{ij_2} - s_{ij_1}) \leq 0, t_{ij} = 0 \quad (5)$$

$$(s_{ij_1} + t_{ij_1} - s_{ij_2})(s_{ij_2} - s_{ij_1}) < 0, t_{ij} \neq 0 \quad (6)$$

$$(s_{ij_1} - s_{ij_2}) \cdot x_{ijk} \leq 0 \quad (7)$$

$$x_{ijk} = \begin{cases} 1 \\ 0 \\ -1 \end{cases} \quad (8)$$

Explanation of mathematical model: objective function is the best solution to solve the maximum completion time.

3. Using genetic algorithm to solve the model

3.1 design genetic algorithm

Next, the model of JOB SHOP scheduling problem will be solved by the selection, crossover and replication of genetic algorithm.

3.1.1 coding chromosomes. According to the operation rules of genetic algorithm, we first code the problem, solve the scheduling problem to solve the maximum processing time and the minimum processing demand, and code the model. For example, if the sequence workpiece (3,1,2,4) is processed, its priority is its sequence (3,1,2,4).

3.1.2 group initialization. Take the processing of N initial groups as an example, randomly generate $n \times m$ matrix elements (take one to five as examples) for initialization research. Is the number of

workpieces n , and is the machining sequence m of workpieces.

3.1.3 calculation of individual fitness. According to the need of solving the problem, the following fitness functions are established:

$$F(x) = \begin{cases} C_{\max} - f(x), & \text{当 } f(x) < C_{\max} \\ 0, & \text{当 } f(x) \geq C_{\max} \end{cases}$$

Where $C_{\max}$ and $f(x)$ represent the theoretical maximum values of adjustable parameters and objective functions respectively [18].

3.1.4 select and copy process solution. According to the need of solving the problem, the copy process [18] is selected as follows:

(1) The sum of fitness values of each individual is solved by $\sum F$, and then the probability of each chromosome is solved. $P[F(i)] = \sum F / F(i)$;

(2) Then, all chromosomes were solved with $P_n = \sum_{i=1}^n P[F(i)]$ for the progressive probability;

(3) The random number $rand(.)$ of $\{0, 1\}$ solution is obtained;

(4) When $P_n < rand(.) < p_n + 1$, the $n + 1$ th chromosome was duplicated;

3.1.5 cross operation of problem solving. Randomly select a group of parent chromosomes, determine the starting position of several genes, and then carry out the cyclic cross operation to obtain the cross operation result chromosomes. The operation process is shown in Table 1.

Table1. Cross operation

XPaternal chromosome	6	9	5	1	4	2	8	3	7
XPaternal chromosome	2	1	3	8	5	7	4	9	6
New chromosomes X'	6	9	5	8	4	7	8	9	7
New chromosomes Y'	2	1	3	1	5	2	4	3	6
X' Variation results	6	9	5	8	4	7	8	9	7

3.1.6 mutation operation for problem solving. For the cross operation result X' generated in Table 1, carry out the mutation solution operation, and randomly generate the mutation number, such as 5 and 4, carry out the position mutation operation, and get the new result as shown in the data in column 5 of Table 1.

3.1.7 stop conditions. Considering the stability of calculation, four generations of genetic algorithm are set to stop running.

3.2 design flow chart of genetic algorithm

According to the flow chart shown in Figure 1, the algorithm is described step by step as follows:

Step 1: randomly generate N individuals to form the initial population; $P(0)$, order $k = 0$;

Step 2: evaluate each individual in the population $P(k)$;

Step 3: judge whether the convergence condition is satisfied, and output the optimization result if the condition is satisfied, otherwise, execute step 4;

Step 4: order $m = 0$; carry out the next step

Step 5: select two individuals from the parent population $P(k)$;

Step 6: judge the conditional crossover probability $p_c > \varepsilon \in [0, 1]$. if the conditions are met, two temporary individuals will be generated by the crossover. Otherwise, the selected parent will be taken as the temporary individual and proceed to the next step;

Step 7: after performing mutation operation on two temporary individuals with mutation probability p_m , put the new individuals into $p(k+1)$, and then order $m = m + 1$;

Step 8: judge whether $m < N$ is set up, and return to step 5 if it is set up; otherwise, order $k = 0$ to return to step 2.

4. simulation experiment

4.1 simulation parameters

In this paper, we use the random data generation method proposed by melouks et al. To obtain the data, and make the following standard division:

(1) The number of workpieces can be divided into five categories: J^1, J^2, J^3, J^4 and J^5 . The number of workpieces is 10, 20, 30, 40 and 50 respectively;

(2) The time period of processed products is uniformly distributed according to the $[1,20]$ region;

(3) The processing procedures are uniformly distributed according to the $[1,5]$ region;

(4) In this paper, the mutation probability is $p_m = 0.08$, the crossover probability is $p_c = 0.85$, the number of genetic iterations is $M = 100$, and the number of machine tools is 5.

In the simulation analysis of the example in this paper, the number of iterations, the number of particles and machines, the probability of crossover and mutation of genetic algorithm are determined, and 100 operations are performed on the example. The five general scheduling rules are as follows.

1. WSPT: weighted minimum processing time priority rule;
2. EDD: priority rule of the earliest construction period;
3. FCFS: first come first serve optimization rules;
4. LPT: priority rule of the longest processing time;
5. Cr: ratio minimum priority rule.

4.2 simulation results and analysis

Table 2 general processing schedule of genetic algorithm and other five scheduling rules

Number of workpieces	rule	Rule comments	Total processing time
10	WSPT	Weighted minimum processing time	210
	EDD	Priority of the earliest construction period	173
	FCFS	First come first serve processing	173
	LPT	Principle of priority of maximum processing time	167
	CR	ratio minimum priority rule	219
		Genetic algorithm program	141

Table 3 the end schedule of each work piece in six scheduling rules under the same machine operation sequence and processing time

Number of workpieces $n = 10$								
rule	workpiece	End time	rule	workpiece	End time	rule	workpiece	End time
WSPT	J1	116	WSPT	J2	210	WSPT	J3	53
EDD	J1	96	EDD	J2	111	EDD	J3	121
FCFS	J1	103	FCFS	J2	104	FCFS	J3	108

LPT	J1	101	LPT	J2	93	LPT	J3	152
CR	J1	79	CR	J2	127	CR	J3	46
genetic algorithm	J1	132	genetic algorithm	J2	127	genetic algorithm	J3	107
WSPT	J4	137	WSPT	J5	83	WSPT	J6	167
EDD	J4	101	EDD	J5	122	EDD	J6	131
FCFS	J4	109	FCFS	J5	122	FCFS	J6	149
LPT	J4	158	LPT	J5	116	LPT	J6	156
CR	J4	66	CR	J5	101	CR	J6	219
genetic algorithm	J4	141	genetic algorithm	J5	107	genetic algorithm	J6	139
WSPT	J7	76	WSPT	J8	191	WSPT	J9	91
EDD	J7	104	EDD	J8	173	EDD	J9	152
FCFS	J7	126	FCFS	J8	173	FCFS	J9	110
LPT	J7	167	LPT	J8	115	LPT	J9	124
CR	J7	74	CR	J8	171	CR	J9	143
genetic algorithm	J7	125	genetic algorithm	J8	140	genetic algorithm	J9	96
WSPT	J10	89	FCFS	J10	141	CR	J10	147
EDD	J10	141	LPT	J10	141	genetic algorithm	J10	96

Table 4 total processing schedule of genetic algorithm and other five scheduling rules with different number of workpieces

$n = 20$		$n = 30$		$n = 40$		$n = 50$	
rule	Total processing time	rule	Total processing time	rule	Total processing time	rule	Total processing time
WSPT	353	WSPT	444	WSPT	616	WSPT	671
EDD	323	EDD	424	EDD	476	EDD	547
FCFS	326	FCFS	418	FCFS	485	FCFS	548
LPT	327	LPT	420	LPT	486	LPT	547
CR	373	CR	451	CR	610	CR	656
genetic algorithm	259	genetic algorithm	352	genetic algorithm	468	genetic algorithm	547

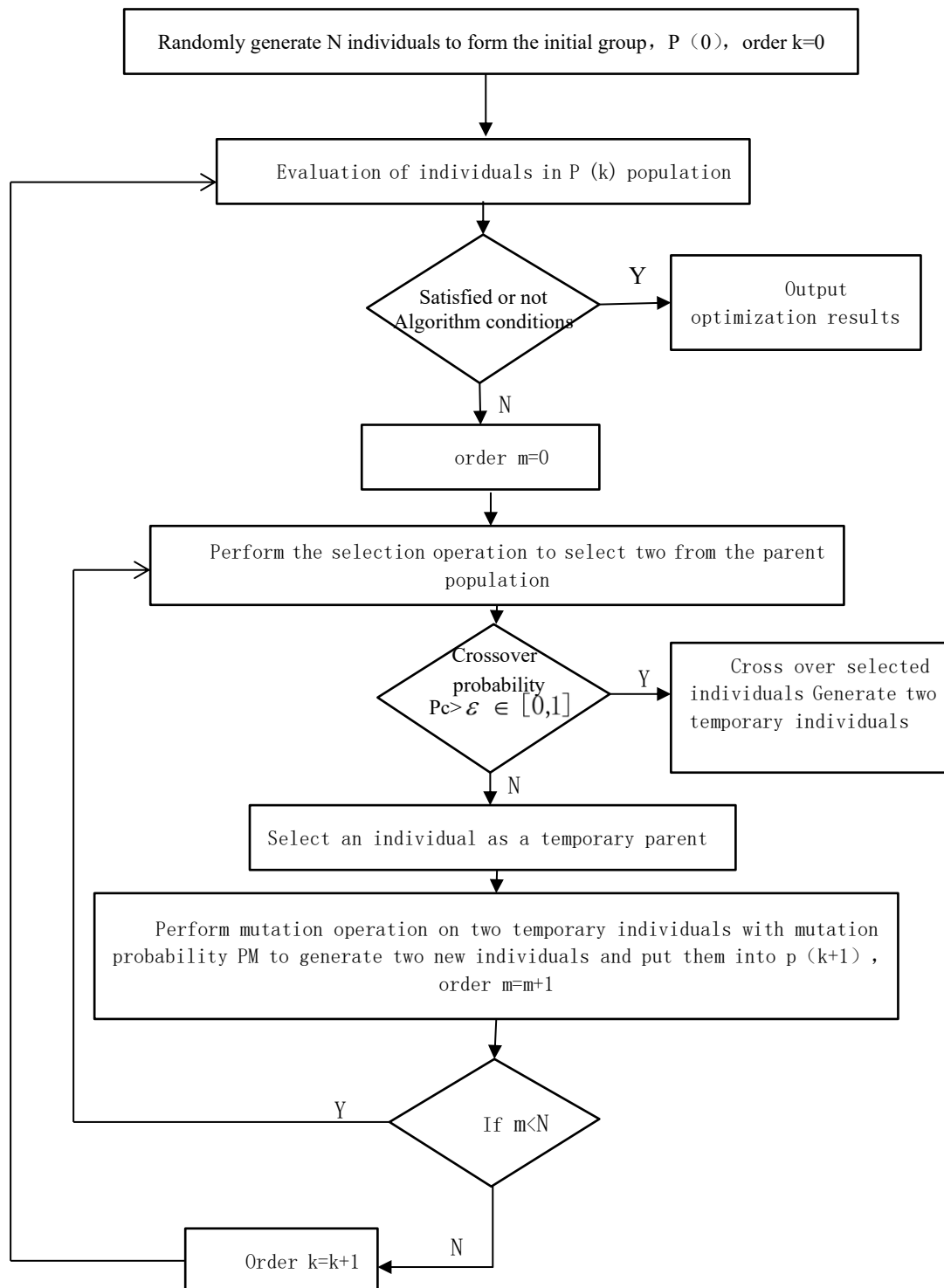


Figure 1. flow chart of genetic algorithm

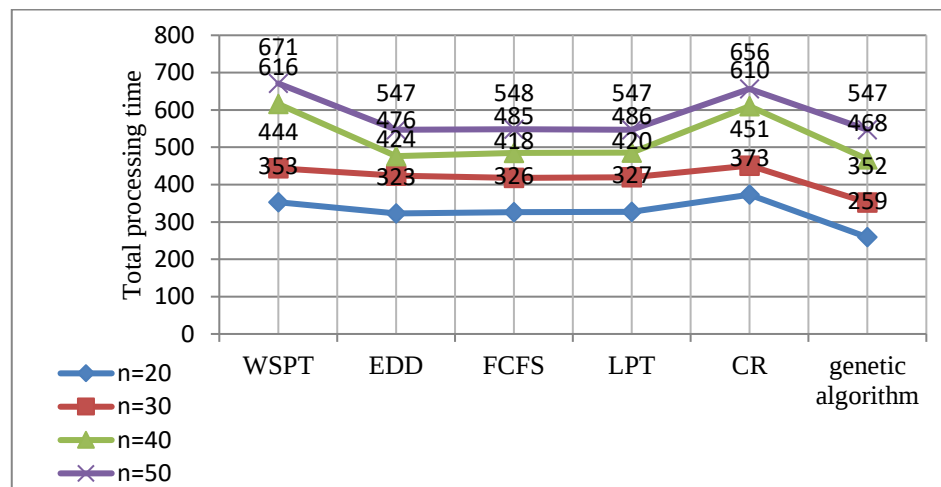


Fig. 2 Comparison of total processing time of different workpieces in six scheduling rules

To sum up, with the objective of minimizing the maximum completion time and considering the constraints of expanding process, with the transformation of workpiece scale and processing time, the genetic algorithm solution is not necessarily optimal for the end time of some workpieces, but optimal for the total delivery time.

5. Conclusion

Aiming at the dynamic job shop scheduling problem, this paper verifies that the minimum maximum completion time obtained by genetic algorithm is better than that obtained by WSPT, EDD, FCFS, LPT and Cr.

reference

- [1] Blazewicz J, Domschke W, Pesch E. The job shop scheduling problem: conventional and new techniques[J]. *European Journal of Operational Research*, 1996, 93(1): 1-33.
- [2] Wang Xilu, Yao Weili, Feng Enmin. Optimization model and algorithm of job shop scheduling problem [J]. *System engineering theory and practice*, 2000: 84-89.
- [3] Vinod V, Sridharan R. Scheduling a dynamic job shop production system with sequence-dependent setups: An experimental study[J]. *Robotics and Computer-Integrated Manufacturing*, 2008, 24(3): 435-449.
- [4] Liu Jia, Liu Lin. study on multi-objective job shop scheduling in parallel machine environment [J]. *Microcomputers and applications*, 2012, 31 (12): 64-66.
- [5] Jonson S M. Optional two and three-stage production scheduling with set-up times included[J]. *Naval Research Logistics Quarterly*, 1954, 1: 64-68.