Flexible job shop scheduling model with parallel processes based on genetic algorithm

Bo Bao[1, a], Lin Zhang[1, b], Bo Zhang[1, c]
1 Air Force Engineering University, Xi'an 710051, China;
[a]18220553171@163.com, [b]962400117@qq.com, [c]zhb8706@163.com

Abstract: This paper studies the flexible job shop scheduling problem under the condition that with parallel processes. In this paper, we analyses the characteristics of the process parallelism in the job shop scheduling problem, puts forward the concept of parallel efficiency, establishes the flexible job shop scheduling model with parallel processes based on the objective criteria of minimizing the makespan, and designs the model solution algorithm based on genetic algorithm. The feasibility of the scheduling model and the effectiveness of the algorithm are verified by simulation experiments. The parameters of process parallel rate are compared and analyzed. The experimental results show that it is effective to improve the scheduling efficiency by improving the parallelism of the process route.

Keywords: scheduling model, parallel processes, genetic algorithm.

1. Introduction

Job shop scheduling problem is a hot topic in the field of production scheduling, which is mainly divided into single machine scheduling[1], parallel machine scheduling[2], flow shop scheduling[3], and job shop scheduling[4], based on the different of the artifacts and the structure of workshop. Among them, the job shop scheduling is widely used in production practice, which has a great practicability. In the related field of job shop scheduling, the research on the flexibility of process route and the flexibility of equipment is more abundant, but the research on the characteristics of parallelism is relatively. With the development of the application field of the workshop scheduling technology, the parallelism between the processes is significantly enhanced, and the research on processes parallelism is increasingly attention. Su Zhaofeng[5] compared the serial scheduling model and the parallel scheduling model of the flexible job shop scheduling problem. James C. Chen[6] studied the job shop scheduling problem with parallel machines and rework. Xu Benzhu[7] proposed a new scheduling algorithm involved the parallel processes of sub-batch, and solve the batch processing scheduling problem effectively. For job shop scheduling, it is also important to consider the parallelism of the process while improving the flexibility of scheduling.

Based on the flexible job shop scheduling problem, considering the parallel process, mathematical model is established based on the objective criteria of minimizing the makespan in this paper, and design a genetic algorithm to solving the scheduling model. The feasibility and effectiveness of the method is validated by an example, and the related parameters were analyzed.

The rest of this paper is as follows: The problem is analyzed and a mathematical model is established in the Section 2. The genetic algorithm is designed in Section 3. The Section 4 provides the experimental results and the analysis. The paper is concluded in the final Section.

2. Problem Analysis and Mathematical Model

2.1 Processes Parallelism Section Headings

In the job shop scheduling, both serial scheduling and parallel scheduling must satisfy the topological sort. For serial scheduling, at any given time, only one process can be processed. For parallel scheduling, at any given time, a process with no precedence relationship can be processed simultaneously on different machines. For example, figure 1 describes the process route of a job, which includes a total of 5 processes, and there is no constraint between the process 1, 2 and the process 3, 4, which reflecting the scheduling process parallelism.

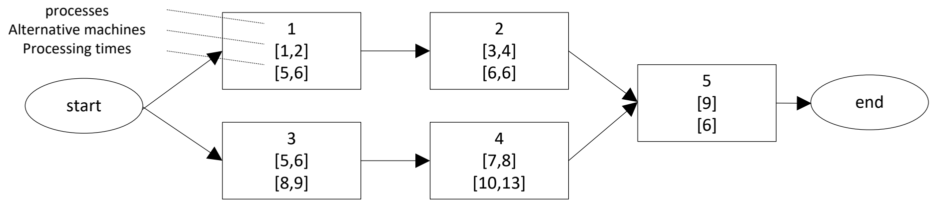

Fig. 1 Schematic diagram of process route

For the parallel relationship between processes, it can be described in the form of data tables. The parallel relationship between the various processes in Figure 1 described as follows:

Table 1 Parallel working table

Processes	1	2	3	4	5
1	×	×	√	√	×
2	×	×	√	√	×
3	√	√	×	×	×
4	√	√	×	×	×
5	×	×	×	×	×

In order to describe the level of parallelism of process in the scheduling effectively, the concept of parallel rate is proposed in this paper, and the calculation formula is as follows:

$$P = number\ of\ parallel\ relations / number\ of\ total\ relations, \quad 0 \le P \le 1 \tag{1}$$

It is indicated that the ratio of the number of parallel relations of processes to the number of total relations of processes in the process route. For example, in Table 1, the number of parallel relations is 4, and the number of total relations is $\sum_{i=1}^{4} i = 10$, there is $P = 4/10 = 0.4$. When all the processes are serial, there is $P = 0$.

2.2 Shop Scheduling Mathematical Model

The flexible job shop scheduling can be described as: a number of jobs J_i are processed in some machines M_j , $i = 1,2,\cdots,n$, $j = 1,2,\cdots,m$. O_{ik} is the k th operation of the job. Job can choose machine M_j to handle the operation O_{ik} , expressed as p_{ijk} , T_{ijk} is the corresponding operation time of p_{ijk} . The operation of the jobs must conform to the process route, and each operation can processed on a set of machines. In order to achieve the best performance index, it is necessary to arrange the jobs on the machines reasonably, under the condition of meet the relevant constraints.

Based on the analysis above, the flexible job shop scheduling problem with parallel that with the objective of minimizing the makespan can be described as a mathematical model:

$$Z = \min \max_{\substack{1 \le i \le n \\ 1 \le k \le l}} (e_{ik}) \tag{2}$$

$$s.t. \quad \sum_{l}^{m} x_{ijk} = 1, \forall i, \forall j, \forall k \tag{3}$$

$$e_{ik} = s_{ik} + T_{ijk} \tag{4}$$

$$e_{ik} \le s_{i(k+1)} \tag{5}$$

$$e_{ju} \le s_{j(u+1)} \tag{6}$$

$$c_i = e_{il} \tag{7}$$

Where the formula (2) is the objective function, e_{jk} is the end time of process O_{ik} , and the start time is s_{jk} . Formula (3) indicates that each operation can chose only one machine. Formula (4) indicates the constraint of the processing time. Formula (5), (6) indicates the time constraints of the operation of jobs, and the time constraints of operations on each machine, and the s_{iu} , e_{iu} are the start and end time of the u th operation on machine M_i . Formula (7) indicates the completion time of each job.

3 Genetic Algorithm

The flexible job shop scheduling problem is a typical NP hard problem, and the parallelism of processes increases the complexity of the problem. In order to solve the flexible job shop scheduling problem, many algorithms have been proposed, such as tabu search algorithm[8], genetic algorithm[9], ant colony algorithm[10], particle swarm optimization algorithm[11] and so on. Genetic algorithm shows great applicability in solving similar problems, so this paper uses the genetic algorithm to solve the problem[12-14].

The flow chart of the double coding genetic algorithm for the job shop scheduling is shown in figure 2.

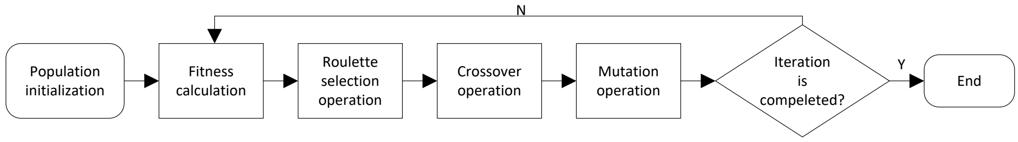

Fig. 2 Flow chart of algorithm

3.1 Chromosome Coding and Decoding

For the flexible job shop scheduling problem, it is necessary to describe the two kinds of information, the processing sequence of the jobs and the processing machine. In this paper, a double layer coding structure is used to encode the chromosome. The first layer represents the processing sequence of the process of the jobs, and the second layer represents the corresponding processing machine for each process. Setting chromosome as:

$$[x_1, x_2, \cdots, x_N, y_1, y_2, \cdots, y_N]$$

Where $x_k (k = 1, 2, \cdots, N)$ is expressed as the number of jobs, $y_k (k = 1, 2, \cdots, N)$ is expressed as the number of machines corresponding to the processes. The length of the chromosome is $2\sum_{i=1}^{n} J_i m_i$, m_i is denoted the operations quantity of job J_i .

The chromosome decoding needs to be based on the information of the optional machine, the processing time and the parallel process. Like the chromosome $[1,2,1,2,3,3,2,1,3,3,1,2]$, assuming that the two processes of job 1 and job 2 are parallel, the two processes of job 3 are serial, without considering the specific value of the operation time, based on the basis of decoding rules, the chromosome decoding process can described as the Gantt chart in figure 2.

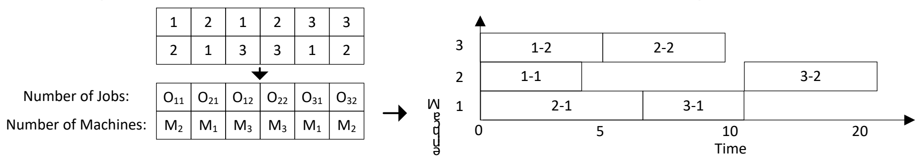

Fig. 3 Chromosome decoding

3.2 Fitness Function

In this paper, the objective function of mathematical model is minimizing the makespan. According to the objective function, the whole task completion time is taken as the fitness value of the chromosome, and the formula is as follows:

$$fitness = \max_{\substack{1 \le i \le n \\ 1 \le k \le l}} (e_{ik}) \tag{8}$$

The smaller the fitness values of chromosome, the shorter the task completion time, the better the corresponding chromosome.

3.3 Genetic Operators

3.3.1 Selection Operator

The selection operation uses the roulette method to select a strong adaptability chromosome based on the probability, the better the fitness value of the chromosome, the greater the probability to be selected. The formula of probability is as follows:

$$p(i) = Fitness(i) / \sum_{i=1}^{n} Fitness(i) \tag{9}$$

$$Fitness(i) = 1/\ fitness(i) \quad (10)$$

3.3.2 Crossover Operator

Crossover operation is divided into two steps: to exchange the genes at the intersection of the two chromosomes, to adjust the chromosomes after crossing, so that insuring the chromosome to meet the constraints. Crossover operations such as: for two chromosomes crossover operation, assuming that the crossover position is 3. First, switch the genes of the crossover points. However, after the crossover operation, some processes increase, or decrease, in this regard, turn the excess operations into a lack operation, and the second half of the chromosome unchanged, thus avoiding the generation of illegal chromosomes. The example is as follows:

$$\begin{matrix} \text{chrosome 1}:[3,2,1,2,3,1,1,2,3,3,2,1] \\ \text{chrosome 2}:[1,1,2,3,2,3,1,2,3,1,2,3] \end{matrix} \xrightarrow{crossover} \begin{matrix} [1,1,2,2,3,1,1,2,3,3,2,1] \\ [3,2,1,3,2,3,1,2,3,1,2,3] \end{matrix} \xrightarrow{adjust} \begin{matrix} [1,1,2,2,3,3,1,2,3,3,2,1] \\ [3,2,1,3,2,1,1,2,3,1,2,3] \end{matrix}$$

3.3.3 Mutation Operator

At first, two mutation points are randomly selected in the first layer, and then the first stage code and the corresponding position of the second machine code are exchanged. For example, the cross mutation operation position of chromosome in point 2 and 3:

$$[2,1,3,1,2,3,1,2,3,3,2,1] \xrightarrow{mutation} [2,3,1,1,2,3,2,1,3,3,2,1]$$

4 Experiments and Analysis

Assuming that the factory needs to complete the task of 4 sets of jobs, each task includes 5 processes, there are 4 sets of equipment available. The optional processing machines and the corresponding processing time as in table ([number of optional machine]/[corresponding processing time (day)]). The process routes of process 1, 2 are same, as the parallel process information table as table 3. The process routes of process 3, 4 are same, as the parallel process information table as table 4. According to the information above, we make the scheduling scheme.

Table 2 Optional number of equipment and corresponding time

jobs	O_{i1}	O_{i2}	O_{i3}	O_{i4}	O_{i5}
J_1	[1]/[6]	[1,3]/[8,6]	[3]/[5]	[1,4]/[9,7]	[4]/[5]
J_2	[1]/[5]	[2,3]/[6,7]	[2,3]/[5,4]	[3]/[3]	[4]/[6]
J_3	[1]/[7]	[4]/[4]	[2,3]/[5,7]	[3,4]/[4,5]	[4]/[5]
J_4	[2,4]/[5,4]	[3]/[4]	[1,4]/[7,7]	[2]/[2]	[1,4]/[6,6]

Table 3 J1 /J2 Parallel process information

Processes	1	2	3	4	5
1	×	√	√	√	×
2	√	×	√	√	×
3	√	√	×	√	×
4	√	√	√	×	×
5	×	×	×	×	×

Table 4 J3/J4 Parallel process information

Processes	1	2	3	4	5
1	×	×	×	√	×
2	×	×	×	√	×
3	×	×	×	√	×
4	√	√	√	×	×
5	×	×	×	×	×

4.1 Parameter Setting and Generation Scheme

MATLAB coding is used to complete the above genetic algorithm, the number of population is 40, the genetic algebra is 50, the crossover rate is 0.8, and the mutation rate is 0.6. Run the algorithm to get a set of scheduling scheme, the result as shown in Figure 4, and the ‘a0b’ in figure indicates the bth operation of job a. The optimal convergence curve of the solving process is shown in figure 5.

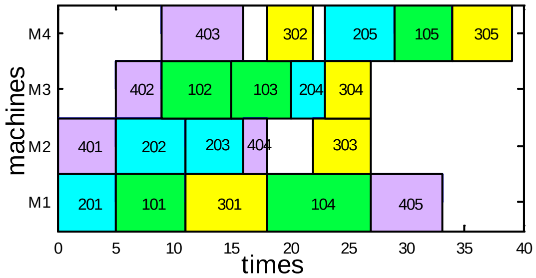

Figure 4 Parallel scheduling results

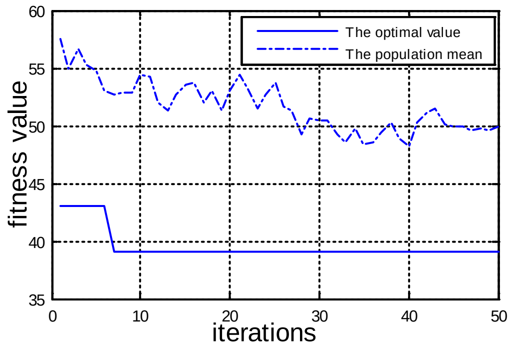

Figure 5 Convergence curve

From the Gantt chart, we can clearly understand which machine the job select and the start and end time of the processes in the plan, and the expected task completion time is 39 days. The optimization result can be seen from the convergence curve. In the scheduling scheme, some processes, such as O_{11}-O_{12}, O_{33}-O_{34}, are parallel processes. Parallel scheduling makes the total completion time shortened, and improves the processing efficiency.

4.2 Simulation Results Analysis

The simulation results of serial scheduling and parallel scheduling are compared and analyzed. Serial decoding of the chromosome, that is, setting the process of the jobs does not exist the parallel process. Serial scheduling results corresponding Gantt chart as shown in figure 6.

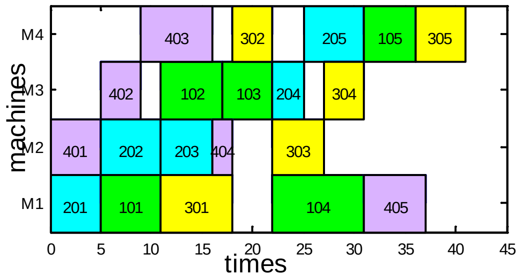

Figure 6 Serial scheduling results

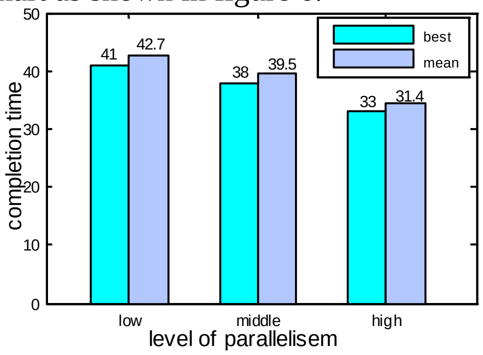

Figure 7 Scheduling efficiency comparison chart

By comparing the results of serial scheduling and parallel scheduling, it can be seen that the processes of parallel scheduling are more closely linked, and the completion time of some processes is shorter than serial scheduling. At the same time, the scheduling time is shorter relatively.

4.3 Parallel Rate Parameter Analysis

By setting the different level of parallelism parameters in the process route of the jobs, that is, different parallel rate, the influence of parallelism on the scheduling efficiency is compared and analyzed. Setting three sets of parallel level parameters: low parallel rate, that is the process route does not have the parallel process (the parallel rate is 0), middle parallel rate, the process route as shown in Table 3 and table 4, high parallel rate, that is all the processes can be parallel operation (the parallel rate is 1). Based on the three sets of process parameters, the simulation experiments were carried out, and the calculation of each group was repeated 10 times. The results are represented as a histogram, as shown in figure 7.

Figure 7 shows the scheduling results under different levels of parallelism, which shows the optimal value and the average value of the scheduling results. It can be found that the higher the parallelism of the scheduling process, the more conducive to improve processing efficiency.

5 Conclusions

In this paper, we study the flexible job shop scheduling problem under the condition that with parallel processes. Based on the analysis of the characteristics of the process parallelism of the job shop scheduling problem, a flexible job shop scheduling model with parallel processes is established, and the optimization algorithm based on genetic algorithm is proposed. The feasibility of the scheduling model and the effectiveness of the algorithm are verified by simulation experiments and results analysis. The simulation results show that enhancing the parallelism of the jobs is helpful to improve the scheduling efficiency. For the following research, it is necessary to further explore the influence of flexibility on the parallelism, and further study the flexible parallel job shop scheduling problem under the condition of large quantity and complex process.

6. References

[1] Suh-JenqYang, Dar-Li Yang, Minimizing the total completion time in single-machine scheduling with aging/deteriorating effects and deteriorating maintenance activities. Computers and Mathematics with Applications, 60 (2010) 2161-2169.
[2] J Lamothe, F Marmier, M Dupuy, et al, Scheduling rule to minimize total tardiness in a parallel machine problem with setup and calendar constraints, 39(2015) 1236-1244.
[3] Rubén Ruiz, José Antonio Vázquez-Rodríguez, The hybrid flow shop scheduling problem, European Journal of Operational Research, 205 (2010) 1–18.
[4] B. Naderi, S.M.T. Fatemi Ghomi, M. Aminnayeri, A high performing metaheuristic for job shop scheduling with sequence—dependent setup times, Applied Soft Computing, 10 (2010) 703-710.
[5] SU Zhao- feng, QIU Hong- ze, Symbiotic evolutionary algorithm for parallel job shop scheduling problem. Computer Engineering and Applications, 44 (2008) 235-238.
[6] James C. Chen, Cheng-Chun Wu, Chia-Wen Chen, et al, Flexible job shop scheduling with parallel machines using Genetic Algorithm and Grouping Genetic Algorithm. Expert Systems with Applications, 39 (2012) 10016-10021.
[7] XUBenzhu, FEI Xiaolu, ZHANG Xingling. Batch division and parallel scheduling optimization of flexible job shop . Computer Integrated Manufacturing Systems, 22(2016) 1953-1964.
[8] Jun-qing Li, Quan-ke Pan, Yun-Chia Liang. An effective hybrid tabu search algorithm for multi-objective flexible job-shop scheduling problems. Computers & Industrial Engineering, 55 (2010) 647-662.
[9] L. De Giovanni, F. Pezzella. An Improved Genetic Algorithm for the Distributed and Flexible Job-shop Scheduling problem. European Journal of Operational Research, 200 (2010) 395–408.
[10] Li-Ning Xing, Ying-Wu Chen, Peng Wang, et al. A Knowledge-Based Ant Colony Optimization for Flexible Job Shop Scheduling Problems. Applied Soft Computing, 10 (2010) 888-896.
[11] Guohui Zhang, Xinyu Shao, Peigen Li. An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem. Computers & Industrial Engineering, 56 (2009) 1309-1318.
[12] X Wang, L Gao, C Zhang, et al, A multi-objective genetic algorithm based on immune and entropy principle for flexible job-shop scheduling problem. The International Journal of Advanced Manufacturing Technology, 51(2010) 757-767.
[13] W Teekeng，A Thammano. Modified Genetic Algorithm for Flexible Job-Shop Scheduling Problems. Procedia Computer Science, 12(2012) 122–128.
[14] Y Demir, SK İşleyen. An effective genetic algorithm for flexible job-shop scheduling with overlapping in operations. International Journal of Production Research, 52(2014) 3905-3921.