Multi-objective Green Flexible Workshop Scheduling Considering Multi-energy Consumption Factors

Shang Gao, Chang Liu and Guan Wang*
School of Economics and Management, Hebei University of Science and Technology, Shijiazhuang, Hebei, 050018, China
*Corresponding author’s e-mail: wangguan@hebust.edu.cn

Abstract. In order to solve the energy consumption problem of shop floor caused by machine processing, adjustment, idling and workpiece transportation, the green flexible workshop scheduling problem was studied, and a multi-objective optimization model was established with the maximum completion time, total machine load and total energy consumption as the objectives. The double-layer coding strategy was adopted to encode processes and machines respectively to optimize the scheduling scheme, and then the adaptive hybrid crossover scheme, mutation scheme and improved elite retention strategy were adopted to improve the operation efficiency and population diversity of NSGA-II algorithm. Finally, a case study is used to verify the effectiveness of the algorithm in solving the multi-objective green flexible workshop scheduling problem, which provides a reference for enterprises to achieve green manufacturing.

1. Introduction
The green workshop scheduling problem is a further study on the traditional workshop scheduling problem, which not only considers the problem of corporate efficiency, but also achieves low-carbon production, which is more complicated than the traditional workshop scheduling problem. Intelligent optimization algorithms are commonly used to solve workshop scheduling problems, such as genetic algorithm\(^1\), particle swarm algorithm\(^2\), ant colony algorithm\(^3\), Jaya algorithm\(^4\), artificial bee colony algorithm\(^5\) and so on. Among them, genetic algorithm is favored by researchers because of its good global search ability, faster solution speed and better scalability. Wu Xiuli et al.\(^6\) proposed a hybrid genetic algorithm based on the integration method. The simulation results verified the feasibility and rationality of the algorithm. Ju Quanyong et al.\(^7\) combined the advantages of genetic algorithm and particle swarm algorithm and proposed a multi-group hybrid algorithm to solve the multi-objective flexible job shop optimization problem. N. Srinivas and Kalyanmoy Deb proposed a non-dominated sorting genetic algorithm to find the Pareto optimal solution to solve the multi-objective optimization problem. The results show that this method can be extended to high-dimensional and more complex multi-objective problems. Subsequently, Kalyanmoy Deb et al. improved NSGA and proposed a multi-objective genetic algorithm based on non-dominated sorting, which can find a better diffusion solution in the problem. However, the standard NSGA-II still has the problem of easily falling into the local optimum. For this paper, the crossover method is improved to increase the diversity of the population and reduce the occurrence of "premature". In recent years, some scholars have also studied the problem of green workshop scheduling. Wen Xiaoyu et al.\(^8\) designed to improve NSGA-II based on the N5 domain structure to solve the green workshop scheduling problem. Liu Caijie et al.\(^9\) considered the
time-of-use electricity price for flexible workshops. For green scheduling, Dai Min et al.\textsuperscript{[10]} solved the green workshop scheduling considering the transportation time.

In summary, the key point of green workshop scheduling is that the construction of the workshop scheduling model must conform to the actual situation of the workshop to the greatest extent. The maximum completion time, processing energy consumption and total machine load as the objective function and establishes a mathematical model were adopted to this paper. The total energy consumption is composed of multiple factors such as adjustment energy consumption, processing energy consumption, idling energy consumption and transportation energy consumption. The crossover method uses process POX crossover and machine MPX crossover to improve the algorithm, and finally a satisfactory scheduling plan is obtained, which is conducive to the implementation of green manufacturing and sustainable development strategies.

2. Description of green flexible workshop scheduling problem and construction of mathematical model

2.1 Problem description

The green flexible workshop scheduling problem can be described as: There are $n$ workpieces to be processed. It will be processed on $m$ machines, the number of processing steps of different workpieces may not be equal, each step can be processed on two or more machines, and the processing time can be different. The processing time and optional machines of all the workpieces to be processed are known. Determine the processing sequence of the workpieces to be processed and the selected machine through a suitable method, and then find a satisfactory solution that takes into account all the goals, in order to be closer to the energy consumption of the actual production of the enterprise, The energy consumption process in the production process can be divided into processing, waiting, handling and adjustment stages.

2.2 Mathematical model construction

The mathematical model of workshop scheduling with the goal of minimum completion time, total machine load and minimum processing energy consumption is established in this paper. The formula is as follows:

In a certain scheduling period, the time taken from the first workpiece to be processed until the last workpiece is completed is represented by $C_T$, where $i$ represents the workpiece number, $j$ represents the process number, $k$ represents the machine number, and $S_{ij}$ represents the start processing time of the process $O_{ij}$, $T_{ij}^k$ represents the processing time of process $O_{ij}$ on machine $k$.

$$\min C_T = \max (F_{ij}) = \max (S_{ij} + T_{ij}^k) \quad (1)$$

The total load of the machine is represented by $P_M$, where $n$ is the number of workpieces, $q_i$ is the number of processes for workpiece $i$, and $m$ is the number of machines.

$$\min P_M = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{m} T_{ij}^k \quad (2)$$

The total energy consumption of the workshop is represented by $E$, $a_{ij}^k$ represents the load unit energy consumption of process $O_{ij}$ on machine $k$, $X_{ij}^k$ indicates whether the process $O_{ij}$ is processed on machine $k$, which is a 0-1 variable, $b_k^r$ represents the no-load unit energy consumption of machine $k$, $F_{ij}$ represents the end processing time of process $O_{ij}$, $t_{f_{i_l,k,l+1,k}}$ represents the machine adjustment energy consumption from the $l$ th task workpiece to the $l + 1$ th task workpiece on machine $k$, $\Omega(J_{l,k,l+1,k}) = \begin{cases} 0, & J_{l,k} = J_{l+1,k} \\ 1, & J_{l,k} \neq J_{l+1,k} \end{cases}$, $P_{\text{trans}}$ represents the transportation power required for the workpiece to be processed by different machines, $\Omega(M_{j,k}, M_{j+1,k}) = \begin{cases} 0, & M_{j,k} = M_{j+1,k} \\ 1, & M_{j,k} \neq M_{j+1,k} \end{cases}$. 

3. Improved NSGA-II algorithm design

Based on the standard NSGA-II algorithm, combined with the characteristics of the enterprise’s green flexible workshop, an adaptive hybrid crossover scheme based on POX process crossover and MPX machine crossover were designed by this paper, and makes corresponding improvements to the shortcomings of the traditional elite retention strategy.

The basic steps are as follows:

(1) Determine the initialization parameters, randomly generate the initial population containing N individuals, and perform non-dominated sorting on the population, then perform selection, mixed crossover and mutation to form the first generation subpopulation;

(2) The second generation as a new starting point, the new population will be re-sorted quickly and non-dominantly after the two generations of father and son are merged;

(3) Each individual in the non-dominated layer needs to calculate the degree of crowding, and select a new parent population suitable for reproduction through the binary tournament mechanism;

(4) Through adaptive hybrid crossover and mutation operations, an improved elite retention strategy is obtained for the offspring population, and returns to step 2 until the set number of iterations is reached;

(5) Take a satisfactory solution from the Pareto solution set and output it as a Gantt chart;

3.1 Encoding and decoding design

In this paper, the two-layer coding method of process and machine is used to solve the problem of process arrangement and machine allocation respectively. The decoding method adopts plug-in greedy decoding, which transforms individual coding into a scheduling Gantt chart.

3.2 Select operation

When solving multi-objective flexible workshop production scheduling, the fast non-dominated sorting algorithm process is as follows:

(1) Compare individuals in the population with each other, record all non-dominated individuals and set their level to 1;

(2) Remove all individuals with level 1 in the population, and repeat step 1 in the remaining population, assigning the level of all undominated individuals found to be 2;
(3) Repeat this process until all the individual levels in the population are graded.

3.3 Mixed crossover and mutation scheme

The core operator of the population in the evolution process is crossover, which can further expand the search and optimization ability of the algorithm taking into account the constraints. In this paper, the process of POX cross mode, and the machine adopts the MPX cross mode.

The population is easy to fall into the local optimum during the evolution process, and the mutation operation can reduce the occurrence of this situation and increase the diversity of the population. The process variation method in this paper is to randomly exchange the order of the two processes. After the exchange, the process sequence of each work piece is checked in turn and the illegal genes generated by the exchange process are repaired. The machine mutation method is to randomly select a process, then call the available machine set of the process, and randomly select a machine to replace the currently used machine, and finally update the corresponding objective function value.

3.4 Improve elite retention strategy

The traditional elite retention strategy is only to retain the superior individuals in the previous generation population, and there are problems in maintaining the poor diversity of the population and easy to fall into the local optimum. In response to these, this has made certain improvements to the elite retention strategy:

(1) Calculate the crowding degree distance for individuals in each level of the non-dominated layer in the population;

(2) In order to ensure that the elite solution is not lost, the individuals with level 1 are retained in the next-generation population. If their number is greater than 0.1 times the population as a whole, only the first 0.1\(N\) individuals will be retained;

(3) In order to ensure the diversity of the population, the method of partial retention to the next generation is adopted for the individuals of other levels. The proportion of the merged population of the population size \(N_p\) is selected, and \(N_i\) individuals are taken out to enter the next generation.

The specific functions are as follows:

\[
N_i = \begin{cases} 
|F_1|, i = 1, |F_1| \leq 0.1N \\
0.1N, i = 1, |F_1| > 0.1N \\
\frac{N}{N_p}, i \geq 2 
\end{cases}
\]  

(11)

In the above formula: \(i\) represents the non-dominated level; \(F_i\) represents the non-dominated level \(i\); \(N_i\) represents the number of individuals selected from the non-dominated level \(F_i\) of level \(i\); \(|F_i|\) represents the number of individuals in the non-dominated level \(i\); \(N_p\) represents the size of the population after the merger is completed; \(N\) represents the size of the progeny population.

3.5 Adaptive crossover probability and mutation probability

Crossover and mutation operations have a certain impact on the algorithm's global search ability and local search ability. In the early stage of algorithm iteration, the global search ability should be expanded, and a larger crossover probability should be used to speed up the evolution of the population; in the later stage of the algorithm iteration. It should be prevented from falling into the local optimum, and a larger mutation probability should be used to improve the local search ability. The crossover probability range is set to (0.5, 0.8), and the mutation probability is set to (0.01, 0.1). The specific calculation formula is as follows:

\[
P_c(i) = \min P_c + \frac{(\max P_c - \min P_c)(1 + \cos(\frac{\pi \cdot i}{\text{gen}}))}{2}
\]  

(12)

\[
P_m(i) = \min P_m + \frac{(\max P_m - \min P_m)(1 + \sin(\frac{\pi \cdot i}{\text{gen}} - \frac{\pi}{2}))}{2}
\]  

(13)
In the above formula: $P_c(i)$ represents the crossover probability of the $i$th iteration; $P_m(i)$ represents the mutation probability of the $i$th iteration; $gen$ represents the set total number of iterations; $\max P_c$ represents the set maximum crossover probability; $\min P_c$ represents The set minimum crossover probability; $\max P_m$ represents the set maximum mutation probability; $\min P_m$ represents the set minimum mutation probability.

4. Simulation

The effectiveness of the algorithm is verified through a scheduling example in a mechanical processing workshop of an enterprise. The workshop requires 8 different types of workpieces to be processed in this scheduling cycle. The process information of each workpiece is known, and each workpiece has a specified delivery. During the period, the specific data is shown in Table 1. Use Matlab R2018a to write algorithm-related functions and run them on a computer with Windows 10, Intel Core i7-5500U, 4GB of RAM, and 64-bit operating system. The population size is set to 200 and the number of iterations is 100.

| Product $J_i$ | Process $O_{ij}$ | $P_{ijk}$ |
|--------------|-----------------|----------|
|              |                 | $M_1$    | $M_2$ | $M_3$ | $M_4$ | $M_5$ | $M_6$ | $M_7$ | $M_8$ |
| $J_1$        | $O_{11}$        | 12       | 10    | 14    | 11    | 17    | 17    |
| $J_1$        | $O_{12}$        | 17       | 24    | 10    | 11    | 17    | 10    |
| $J_2$        | $O_{21}$        | 11       | 10    | 21    | 14    | 17    | 17    |
| $J_2$        | $O_{22}$        | 8        | 12    | 19    | 11    | 17    | 17    |
| $J_2$        | $O_{23}$        | 15       | 21    | 18    | 9     | 10    | 10    |
| $J_2$        | $O_{26}$        | 9        | 7     | 10    | 8     | 10    | 10    |
| $J_3$        | $O_{31}$        | 14       | 17    | 17    | 17    | 17    | 17    |
| $J_3$        | $O_{32}$        | 23       | 23    | 18    | 18    | 18    | 18    |
| $J_3$        | $O_{33}$        | 20       | 9     | 22    | 22    | 22    | 22    |
| $J_3$        | $O_{34}$        | 7        | 10    | 11    | 9     | 9     | 9     |
| $J_4$        | $O_{41}$        | 18       | 17    | 17    | 17    | 17    | 17    |
| $J_4$        | $O_{42}$        | 10       | 12    | 12    | 12    | 12    | 12    |
| $J_4$        | $O_{43}$        | 8        | 9     | 9     | 9     | 9     | 9     |
| $J_5$        | $O_{51}$        | 24       | 16    | 16    | 16    | 16    | 16    |
| $J_5$        | $O_{52}$        | 18       | 19    | 19    | 19    | 19    | 19    |
| $J_6$        | $O_{61}$        | 20       | 22    | 22    | 22    | 22    | 22    |
| $J_6$        | $O_{62}$        | 18       | 18    | 18    | 18    | 18    | 18    |
| $J_6$        | $O_{63}$        | 17       | 19    | 19    | 19    | 19    | 19    |
| $J_7$        | $O_{71}$        | 16       | 17    | 17    | 17    | 17    | 17    |
| $J_7$        | $O_{72}$        | 20       | 22    | 22    | 22    | 22    | 22    |
| $J_7$        | $O_{73}$        | 18       | 23    | 23    | 23    | 23    | 23    |
| $J_7$        | $O_{74}$        | 10       | 24    | 24    | 24    | 24    | 24    |
| $J_9$        | $O_{91}$        | 11       | 21    | 21    | 21    | 21    | 21    |
| $J_9$        | $O_{92}$        | 25       | 25    | 25    | 25    | 25    | 25    |

The Gantt chart of the scheduling plan finally found from the Pareto solution is shown in Figure 1 below. The horizontal axis represents the processing time, and the vertical axis represents the machine number. The maximum completion time is 75, the total load of the machine is 381, and the total energy consumption is 104.59. The original scheduling, where the maximum completion time is 94, the total machine load is 389, and the total energy consumption is 112.79. Compared with the original scheme, the optimized scheme reduces the maximum completion time, machine load time and workshop energy consumption by 20.21%, 2.05% and 7.27% respectively.
5. Conclusion

Based on the characteristics of the green flexible workshop, a scheduling model suitable for the actual situation of the workshop is established in this paper, focusing on the factors that affect the energy consumption of the workshop. An adaptive hybrid cross-mutation scheme and an improved elite retention strategy are proposed to improve the efficiency of the algorithm. The improved algorithm was verified with data from a certain enterprise scheduling cycle. Compared with the original scheduling plan, the maximum completion time, machine load and workshop energy consumption were reduced by 20.21%, 2.05% and 7.27%, respectively.

Acknowledgments

There are too many people who have helped me. I sincerely thank all those who care about and help me. Thank them for giving me the power to challenge myself, surpass myself, and pursue continuous pursuit. Thank you very much.

References

[1] Wang C, Zhang M, Ji ZZ, et al.(2017) Genetic algorithm for solving multi-objective dynamic flexible job shop scheduling. Journal of System Simulation,29:1647-1657.
[2] Bai JJ, Gong YG, Wang NS, et al.(2010) Multi-objective flexible job shop scheduling with lot-splitting. Computer Integrated Manufacturing Systems,16:396-403.
[3] Lu HH, Lu YJ.(2019) Flexible job-shop scheduling based on hybrid estimation of distribution and ant colony algorithm,36:568-573.
[4] Wang JH, Pan YJ, Sun R.(2021) Multi-objective flexible job shop green scheduling problem with self-adaptive Jaya algorithm. Control and Decision,36:1714-1722.
[5] Wu R, Guo SS, Li YB, et al.(2019) Improved artificial bee colony algorithm for distributed and flexible jobshop scheduling problem. Control and Decision,34:2527-2536.
[6] Wu XL, Sun SD, Yu JJ, Zhang HF.(2006) Research on multi-objective optimization for flexible job shop scheduling. Computer Integrated Manufacturing Systems,5:731-736.
[7] Ju QY, Zhu JY.(2007) Flexible job-shop optimization scheduling for multi-objective batch production. Chinese Journal of Mechanical Engineering,8:148-154.
[8] Wen XY, Sun HQ, Li H.(2020) Research on multi-objective green job shop scheduling problem based on improved NSGA-II.Journal of Henan Polytechnic University(Natural Science),39:120-129.
[9] Liu CJ, Xu ZT, Zhang Q, et al.(2020) Green Scheduling of Flexible Job Shops Based on NSGA-II under TOU Power Price. China Mechanical Engineering,31:576-585.
[10] Dai M, Zhang YW, Zeng L.(2020) A Research on Green Job Shop Scheduling Considering Transportation Time. Industrial Engineering Journal,23:41-48.