



Energy-Aware Scheduling of Two-Stage Flexible Flow Shop Based on Group Technology: Consideration of the Inconsistency of Process Differences

Zhiqiang Zeng¹ · Le Gao¹ · Zhiyong Hong¹

Received: 23 March 2022 / Accepted: 27 July 2022
 © The Author(s) 2022

Abstract

The diversity of consumer demand makes the manufacturing industry shift from the production mode of small variety and large batch to the production mode of multi-variety and small batch, which leads to the increase of production cost and the decrease of production efficiency, especially the increase of energy consumption. In this paper, the energy-aware scheduling problem of two-stage flexible flow shop based on group technology is studied, in which the process differences of jobs in the two stages are inconsistent. To solve the problem of unbalanced load of machines caused by group technology, this paper uses a dynamic splitting strategy to balance the load of machines on the basis of increasing the setup energy consumption as little as possible. Considering that the research problem is a multi-objective optimization problem, this study develops an improved multi-objective scatter search algorithm. According to the characteristics of the problem, the coding scheme, decoding scheme, diversity generation method and combination method are designed. The experimental results show that the group technology can effectively reduce energy consumption, and the dynamic split strategy can balance the load of machines effectively. Moreover, the results show the proposed method is better than the existing methods in terms of obtaining better solutions.

Keywords Energy-aware scheduling · Flexible flow shop · Group technology

List of Symbols

n	Number of jobs	p'	Number of groups of jobs in the first stage
m	Number of machines	G'	Job group set in the first stage
J	Job set	G'_i	The i -th job group in the first stage
J_i	The i -th job	p'_i	Number of jobs of i -th job group in the first stage
M	Machine set	$\pi_{G'_i}$	Sequence of jobs in group G'_i
M_i	The i -th machine	$\pi_{G'_i}(j)$	The j -th job in group G'_i
$m1$	Number of machines in the first stage	c	Number of job types in the second stage
M^p	Machine set in the first stage	c'	Number of groups of jobs in the second stage
M_i^p	The i -th machine in the first stage	G''	Job group set in the second stage
$m2$	Number of machines in the second stage	G''_i	The i -th job group in the second stage
M^c	Machine set in the second stage	c''_i	Number of jobs of i -th job group in the second stage
M_i^c	The i -th machine in the second stage	$\pi_{G''_i}$	Sequence of jobs in group G''_i
p	Number of job types in the first stage	$\pi_{G''_i}(j)$	The j -th job in group G''_i
		C	Makespan
		E	Total energy consumption
		$C_{i,h}$	Complete time of group i or the complete time of job i on machine h
		E^p	Processing energy consumption
		E^s	Setup energy consumption

✉ Le Gao
 le.gao@nscg-gz.cn
 Zhiqiang Zeng
 zhiqiang.zeng@outlook.com

¹ Faculty of Intelligent Manufacturing, Wuyi University, Jiangmen 529020, China

$O_{i,j}$	Binary function
$U_{i,j}^p$	Processing power of group i on machine j
$T_{k,j}^p$	Processing time of group k or job k on machine j
$Q_{i,j,h}$	Binary function
$P_{i,j,h}^s$	Power of group i switching to group j on machine h
$T_{i,j,h}^s$	Setup time of group i switching to group j on machine h
$B_{j,h}$	Start time of the group j or job j on machine h
Ref	Reference set
Ref1	Part of reference set
Ref2	Part of reference set
Subset1	Subset of solution
Subset2	Subset of solution
popn	A new population
Ref1_num	Size of Ref1
Ref2_num	Size of Ref2
popu	Merger of Ref and popn
popu_num	Size of popu

1 Introduction

Processing energy consumption and setup energy consumption are the most important two parts of energy consumption in production activities. Processing energy consumption refers to the energy consumption in the process of job processing. Due to the differences in the machines, there are differences in processing energy consumption generated by jobs processed in different machines. Setup energy consumption refers to the energy consumption of the machine in the production switching process. When the processing processes of two consecutive processing jobs are inconsistent, the machine needs to be readjusted to process subsequent jobs, and the machine adjustment process will produce energy consumption that cannot be ignored. Energy-aware scheduling (energy efficiency scheduling) is one of the effective ways to reduce these two-part energy consumptions, which has been widely studied [1–5]. With the development of science and technology and the improvement of living standards, consumers are not satisfied with the products that provide standard functions. Personalized customization has become more and more popular. Facing the demand of large-scale personalized customization, the manufacturing industries have changed from the traditional mass production model to the multi-variety and small batch manufacturing model. In the multi-variety and small batch manufacturing mode, a large number of production switching lead to the increase of manufacturing cost and the decrease of production efficiency. In the process of scheduling, group technology is used to combine the jobs with the same process to

produce together, so as to reduce production switching and production costs, which is an effective way to solve the problem of multi-variety and small batch manufacturing mode. There are two main grouping strategies in group technology [6]. The first one is to combine all jobs of the same process into a group for processing. This strategy can minimize production switching, but may increase the makespan and lead to the imbalance of machine load. The second is to allow multiple setups in jobs of the same process, that is, jobs of the same process can be divided into multiple groups. This strategy will reduce the makespan and balance the machine load, but may increase the production switching.

The group scheduling problem is more complex than the traditional scheduling problem. When the first group strategy is adopted, the group scheduling problem needs to solve two subproblems: (i) the traditional scheduling problem; and (ii) the jobs sorting problem within a group. When the second grouping strategy is adopted, group scheduling needs to solve three subproblems. In addition to the above two problems, it also needs to solve the jobs grouping problem of the same process (how many groups of jobs of the same process are divided into and what jobs each group contains). Flexible flow shop is a type of flow shop, which is a manufacturing system that exists widely in the real world. Jobs need to be processed in multiple stages in order to form the final product, and there are multiple parallel machines in each stage. In the study of group scheduling in flow shop, the first grouping strategy is used in most literature [7–17]. There are few studies on the second grouping strategy, Shahvari and Logendran [18] optimized the completion time and tardiness simultaneously in the group scheduling of flexible flow shop, they abandoned the assumption that each group has single setup, allowing multiple setups for each type of job. Feng et al. [19] allowed multiple setups in the group scheduling of flexible flow shop, the makespan and preventive maintenance were considered simultaneously in their study. Jain et al. studied the group scheduling problem of flow shop with sequence-dependent setup time (SDST), which considered SDST among groups and among part types within a group [20].

At present, most of the researches on group scheduling in flow shop consider that the process differences of jobs are consistent in all stages [21–24]. However, there are many manufacturing systems in reality, and the differences of jobs process are not consistent [6, 25]. For example, jobs may have the same process in the first stage, but may not have the same process in the later stages. There are few studies on group scheduling of flow shop considering inconsistency of the process differences of jobs. Andrés et al. [26] studied the group scheduling problem of tile industry, which is a three-stage flexible flow shop. They defined similarity coefficient for each job, and grouped jobs based on similarity coefficients to reduce setup time. Among the existing studies

on group scheduling, most of them use group technology to reduce the setup time, makespan, earliness and tardiness and so on [27–37]. However, setup time minimization is not the same as setup energy consumption minimization. Up to now, the research on group scheduling to reduce energy consumption is rare. He et al. studied the group scheduling problem of flow shop, which optimized makespan, total flow time and total energy consumption at the same time [38].

Inspired by the production scheduling of tissue paper enterprises, this paper studies the problem of energy-aware scheduling of two-stage flexible flow shop considering inconsistency of the process differences of jobs, which intends to reduce the energy consumption by using group technology. The studied manufacturing system consists of two processing stages. The jobs with the same processing technology in the second stage must have the same processing technology in the first stage. On the contrary, jobs with the same processing technology in the first stage may not have the same processing technology in the second stage. By combining group technology with scheduling, this study not only improves production efficiency by optimizing the makespan, but also optimizes processing energy consumption and setup energy consumption to improve energy efficiency. The main contributions of this paper are as follows:

1. A new energy-aware scheduling method based on group technology is proposed for the two-stage flexible flow shop considering inconsistency of the process differences of jobs.
2. An improved multi-objective scatter search algorithm is proposed. According to the characteristics of this problem, a three-layer chromosome encoding scheme is proposed, and a dynamic split strategy is used to split the jobs dynamically in the process of chromosome decoding, which ensures the load balance between machines. In order to ensure the diversity of population, a method of generating population diversity and a method of updating reference set are designed. In addition, according to the characteristics of encoding scheme, a combination method and a local improvement method are designed.
3. A case study is carried out based on the real data from a tissue paper enterprise to evaluate performance of the proposed method.

The rest of the paper is arranged as follows: Sect. 2 describes the proposed problem and establishes the mathematical model of the problem; Sect. 3 describes the proposed multi-objective scatter search method in detail; Sect. 4 analyzes the performance of the proposed method through a case study; The last section is the conclusion, which summarizes the content of this study from many aspects.

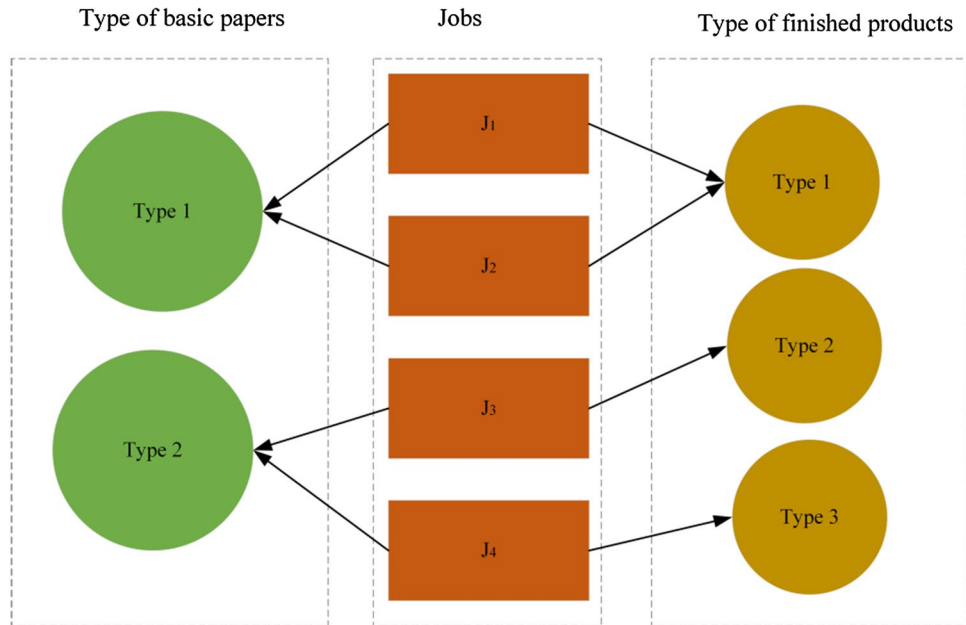
2 Problem Description and Modeling

2.1 Problem Description

This study is inspired by the production scheduling of tissue paper enterprises, which belongs to two-stage flexible flow shop manufacturing system, including papermaking stage and conversion stage. There are many parallel production lines in the papermaking stage and the conversion stage, each production line can process many types of products. Any job needs to be processed in any production line in the papermaking stage first, and then in any production line in the conversion stage. In the papermaking stage, the pulp board, water, chemicals, etc. are mixed, and the base paper is produced through pulping, refining, forming, pressing, drying and reeling. In the conversion stage, the base paper is processed by the rewinder to produce the rewinder strip, then cut into small rolls, and finally packaged by the packaging machine to get the final product. In the production process, various accessories, such as paper core, packaging film and perfume, are needed. For the same type of finished products, they must use the same base paper. However, different types of finished products may also use the same base paper. According to the characteristics of the products produced in the papermaking stage and the conversion stage, the types of jobs of tissue paper enterprises may appear as shown in Fig. 1. Figure 1 shows the processing technology of J1 and J2 is the same in papermaking stage and conversion stage. The J3 and J4 have the same processing technology in the papermaking stage, while the processing technology in the conversion stage is different. Among the tissue paper enterprises, there are hundreds type of finished products and dozens type of base paper. In production scheduling of tissue paper enterprises, jobs with the same type of processing technology in the papermaking stage are often arranged to be produced together, and the same type of processing technology in the conversion stage are arranged to be produced together as far as possible. However, due to the differences between the machine, the two jobs of continuous production in the papermaking stage are not necessarily continuous production in the conversion stage, which makes it difficult to reduce production switching.

Tissue paper enterprises are energy intensive manufacturing industries. Processing energy consumption and setting energy consumption are the two parts that account for the largest proportion of total energy consumption. The processing energy consumption of the job is not only related to the process of the product, but also to the machine. In the papermaking stage, the grammage of the product is one of the most important factors affecting the energy consumption. The larger the grammage, the greater the energy consumption. The processing energy consumption of conversion

Fig. 1 The example of types of jobs in tissue paper enterprises



stage is mainly related to the height and length of the small roll. The speed of the machine varies greatly according to the height and length of the small roll, which leads to the difference of energy consumption between products. In the tissue paper enterprises, setup time and setup energy consumption are related to the processing sequence and machine. For example, in the papermaking stage, there is a difference between the energy consumption of the conversion from low grammage products to high grammage products and the energy consumption of the conversion from high grammage products to low grammage products. In addition, according to the characteristics of production scheduling in tissue paper enterprises, this paper has the following assumptions:

1. Each product can be produced in any production line.
2. The second operation can only be started after the first operation is completed.
3. Any production line can only process one job at a time.
4. Once a job starts production, it cannot be interrupted.
5. A job can no longer be divided into multiple jobs.

2.2 Modeling

In the two-stage flexible flow shop, it is assumed that there are n jobs $J = \{J_1, J_2, \dots, J_n\}$ and m machines $M = \{M_1, M_2, \dots, M_m\}$. There are m_1 machines $M^p = \{M_1^p, M_2^p, \dots, M_{m_1}^p\}$ in the first stage and m_2 machines $M^c = \{M_1^c, M_2^c, \dots, M_{m_2}^c\}$ in the second stage. According to the process characteristics of jobs in the first stage, jobs can be divided into p types, according to these p types of jobs, all jobs can be divided into p' groups $G' = \{G'_1, G'_2, \dots, G'_{p'}\}$.

The Group G'_i contains p'_i jobs, each group contains jobs of the same type, different groups may have the same type of jobs. $\pi_{G'_i}$ represents the sequence of jobs in group G'_i , $\pi_{G'_i}(j)$ represents the j -th job in group G'_i . According to the process characteristics of jobs in the second stage, jobs can be divided into c types, according to these c types of jobs, all jobs can be divided into c' groups $G'' = \{G''_1, G''_2, \dots, G''_{c'}\}$. The Group G''_i contains c'_i jobs, each group contains jobs of the same type, different groups may have the same type of jobs. Each G''_i is consist of multiple G''_i . $\pi_{G''_i}$ represents the sequence of jobs in group G''_i , $\pi_{G''_i}(j)$ represents the j -th job in group G''_i . Because jobs of the same process can be divided into multiple groups, so $p \leq p'$ and $c \leq c'$. Because of the jobs with the same process in the second stage, they must have the same process in the first stage. On the contrary, jobs with the same process in the first stage, they may not have the same process in the second stage. Therefore, $p \leq c$, but p' may not be less than c' . Combined with group technology, the energy-aware scheduling model of two-stage flexible flow shop can be represented by the following formula:

$$\begin{cases} \min(C) \\ \min(E) \end{cases}, \quad (1)$$

$$E = E^p + E^s, \quad (2)$$

$$C = \max(C_{i,h}) \forall i \in J; \quad \forall h \in M, \quad (3)$$

$$E^p = \sum_{i \in G'} \sum_{j \in M^p} \left(O_{i,j} \cdot U_{i,j}^p \cdot \sum_{k \in G'_i} T_{k,j}^p \right) + \sum_{i \in G''} \sum_{j \in M^c} \left(O_{i,j} \cdot U_{i,j}^p \cdot \sum_{k \in G''_i} T_{k,j}^p \right), \quad (4)$$

$$E^s = \sum_{i \in G'} \sum_{j \in G'} \sum_{h \in M^p} Q_{i,j,h} \cdot P_{i,j,h}^s \cdot T_{i,j,h}^s + \sum_{i \in G''} \sum_{j \in G''} \sum_{h \in M^c} Q_{i,j,h} \cdot P_{i,j,h}^s \cdot T_{i,j,h}^s, \quad (5)$$

$$Q_{i,j,h} (B_{j,h} - C_{i,h} - T_{i,j,h}^s) = 0 \quad \forall i, j \in G'; \quad \forall h \in M^p, \quad (6)$$

$$\begin{cases} \sum_{h \in M^c} O_{j,h} B_{j,h} = \sum_{k \in M^p} O_{\pi_{G''}(1),k} C_{\pi_{G''}(1),k} \\ \text{if } j \text{ is the first job group on the machine } \forall j \in G'' \\ Q_{i,j,h} \left(B_{j,h} - \max \left(C_{i,h}, \sum_{k \in M^p} O_{\pi_{G''}(1),k} C_{\pi_{G''}(1),k} \right) - T_{i,j,h}^s \right) \\ = 0 \text{ else } \forall i, j \in G''; \forall h \in M^c \end{cases}, \quad (7)$$

the same group have the same process, the processing power of the jobs of the same group on the same machine is the same, $U_{i,j}^p$ is the processing power of group i on machine j . $T_{k,j}^p$ represents the processing time of group k or job k on machine j . $Q_{i,j,h} = \{0, 1\}$, when group i and group j are two consecutive processing job groups of machine h , $Q_{i,j,h} = 1$, otherwise, $Q_{i,j,h} = 0$. $P_{i,j,h}^s$ represents the power of group i switching to group j on machine h , it equals the power of last job of group i switching to the first job of group j on machine h . $T_{i,j,h}^s$ represents the setup time of group i switching to group j on machine h , it equals the setup time of the last job of group i switching to the first job of group j on machine h .

Formulas (6) and (7) ensure that the machine can only process one group at the same time, where $B_{j,h}$ represents the start time of the group j or job j on machine h . In stage one, the start time of the next group is the end time of the previous group plus the setup time. In stage two, the start time of the next group is equal to the maximum value of the completion time of the first operation of first job of the

$$B_{\pi_{G'_i}(k+1),h} - C_{\pi_{G'_i}(k),h} = 0 \quad \forall i \in 1, 2, \dots, p'; \quad \forall k \in 1, 2, \dots, p'_i - 1; \quad \forall h \in M^p, \quad (8)$$

$$B_{\pi_{G''_i}(k+1),h} - \max(C_{\pi_{G''_i}(k),h}, \sum_{j \in M^p} O_{\pi_{G''_i}(k+1),j} \cdot C_{\pi_{G''_i}(k+1),j}) = 0 \quad \forall i \in 1, 2, \dots, c'; \quad \forall k \in 1, 2, \dots, c''_i - 1; \quad \forall h \in M^c, \quad (9)$$

$$T_{i,j}^p = \sum_{k \in i} T_{k,j}^p \quad \forall i \in G', \quad (10)$$

$$T_{i,j}^p = \sum_{k \in i} T_{k,j}^p \quad \forall i \in G'', \quad (11)$$

$$C_{i,h} - B_{i,h} = T_{i,h}^p \quad \forall i \in G' \cup J; \quad \forall h \in M^p, \quad (12)$$

$$C_{i,h} - B_{i,h} = T_{i,h}^p \quad \forall i \in G'' \cup J; \quad \forall h \in M^c, \quad (13)$$

$$\sum_{j \in M^p} O_{i,j} = 1 \quad \forall i \in J, \quad (14)$$

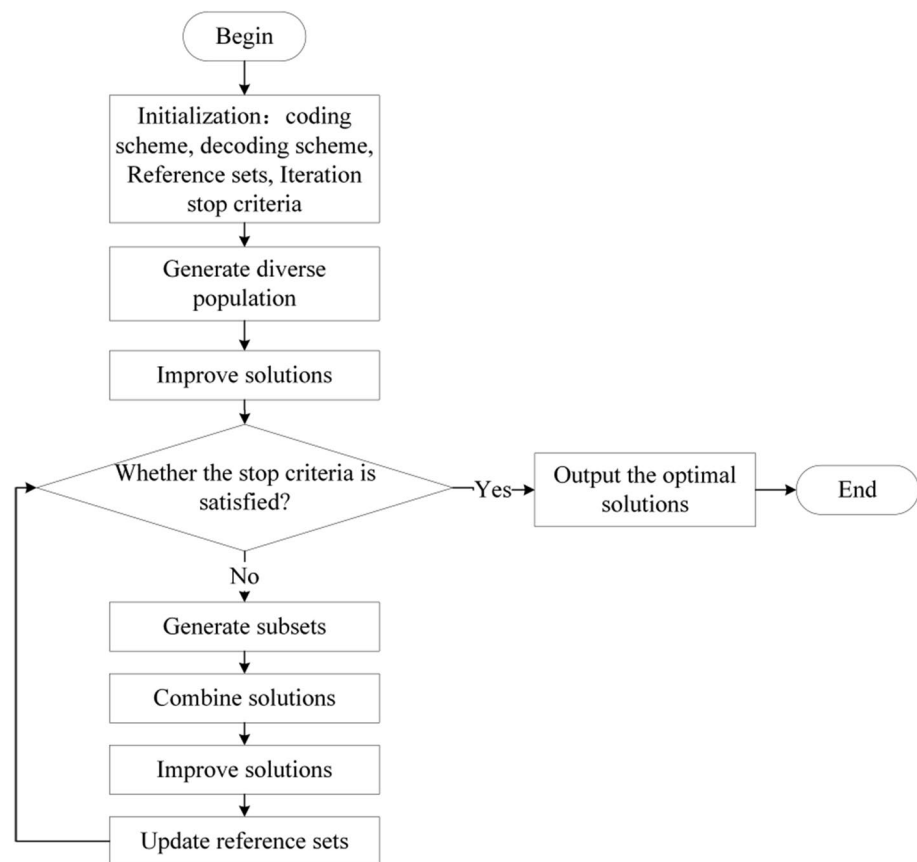
$$\sum_{j \in M^c} O_{i,j} = 1 \quad \forall i \in J, \quad (15)$$

where C is the makespan, E is the total energy consumption, $C_{i,h}$ represents the complete time of group i or the complete time of job i on machine h . E^p represents the processing energy consumption, E^s represents the setup energy consumption. $O_{i,j} = \{0, 1\}$, it equals one means that group i or job i is processed on machine j , it equals zero means that group i or job i is not processed on machine j . Because the jobs of

group and the completion time of the previous group plus the setup time. In addition, formulas (8) and (9) indicate the jobs in each group must not be preempted, that means the start of the next job must wait until the end of the previous job. Moreover, in stage two, the start of the job must wait until the completion of the previous operation of the job. Equations (10) and (11) show that the processing time of each group is equal to the total processing time of all jobs in the group. Formulas (12) and (13) indicate that each job cannot be interrupted once it is started. Formulas (14) and (15) indicate that each job must and can only be processed once in each stage.

3 The Improved Multi-objective Scatter Search Algorithm

The scatter search algorithm was first proposed by Glover [39] to solve the integer programming problem. The scatter search algorithm defines a flexible framework, which contains multiple modules, including diversity generation module, solution improvement module, reference set update module, subset generation module, solution combination module, etc. The scatter search algorithm forms high-quality

Fig. 2 The flow chart of scatter search algorithm

and diverse solutions in the reference set, and then generates new solutions through the subset generation and combination methods, and improves the new solutions by the solution improvement method, and then renews the reference set, and obtains the global optimal solution or satisfactory solution by continuously cycling the above steps. The Fig. 2 is the flow chart of the scatter search algorithm. According to the characteristics of the problem, combined with the fast non-dominated sorting algorithm, this paper develops an improved multi-objective scatter search algorithm (IMSS). First, a three-layer chromosome coding scheme is proposed, and a dynamic split strategy is used to split the job group dynamically in the process of chromosome decoding to ensure the load balance between machines. To ensure the diversity of population, a method of population diversity generation is developed and applied to the reference set updating method. In addition, according to the characteristics of encoding, a combination method and a local improvement method are designed. Each step of the proposed algorithm is shown in Fig. 2.

3.1 Encoding and Decoding

According to the research problem, in the first stage, the problems of processing sequence of job groups

$G' = \{G'_1, G'_2, \dots, G'_{p'}\}$ and machine assignment need to be solved. Because the jobs in the same group are of the same type, the order of jobs in the same group does not affect the setup energy consumption and completion time. In addition, this paper uses the rule that the machine with the earliest available time is assigned first. When the order of job groups is determined, the problem of machine assignment is solved. Therefore, the processing sequence of job groups $G' = \{G'_1, G'_2, \dots, G'_{p'}\}$ can be represented by a vector. To reduce setup energy consumption, we combine jobs with the same process in the first stage into a group. For example, when $p' = 3$, it can use [1–3] to represent the processing sequence of groups in stage one. The first chromosome determines the processing order of groups $G' = \{G'_1, G'_2, \dots, G'_{p'}\}$, but it cannot determine the processing order of groups $G'' = \{G''_1, G''_2, \dots, G''_{c'}\}$. As the job sequence of each group G'_i will affect the completion time and energy consumption of the second stage. Therefore, we use a chromosome to represent the jobs sequence of each group G'_i . To reduce switching, the minimum unit of sequencing for each group G'_i is set to a G''_i . Suppose G'_1 consists of G''_1, G''_2 and G''_3 , G'_2 consists of G''_4, G''_5 and G''_6 , G'_3 consists of G''_7 and G''_8 . Then, the second chromosome can be

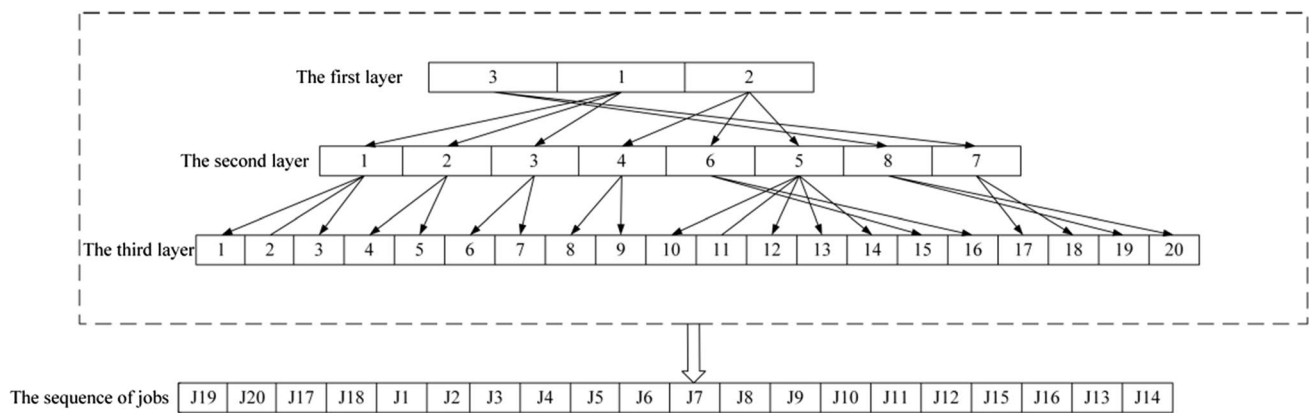


Fig. 3 Three-layer chromosome scheme

represented by a vector [1–8]. In addition, the sequence of jobs in each group G_i'' does not affect energy consumption, but affects the completion time. Because the jobs of each group G_i'' are processed on the same machine in the first and second stages, to reduce the makespan, according to the NEH rule, the jobs of each group G_i'' are sorted according to the processing time from small to large. So, the third layer chromosome represents the jobs sequence of each group G_i'' , and it is fixed. Figure 3 shows a three-layer chromosome representation of jobs J1–J20. Here, it is assumed that the processing time of jobs J1–J20 increase from small to large.

In the iteration process, chromosomes need to be decoded to form a feasible production schedule. The first layer of chromosomes represents sequence of G' . Because the jobs of the same group G_i' have the same process in the first stage, in order to reduce the setup energy consumption, the jobs of the same group G_i' should be processed on the same machine in the first stage. Assume there are three machines $M^p = \{M_1^p, M_2^p, M_3^p\}$ in the first stage, M_1^p has the earliest available time, then M_2^p , and finally M_3^p . Assume there are three machines $M^c = \{M_1^c, M_2^c, M_3^c\}$ in the second stage, M_1^c has the earliest available time, then M_2^c , and finally M_3^c . According to the above chromosome coding, for the

convenience of demonstration, it is assumed that the processing time of all jobs is the same. The first layer chromosome is [1–3], which means that the jobs of G_3' are first arranged to be processed in M_1^p . According to the second layer chromosome and the third layer chromosome, the job sequence of G_3' is [17–20], so the job processing sequence on M_1^p is [17–20]. Similarly, the processing sequence on M_2^p is [1–7], and the processing sequence on M_3^p is [8–16]. Figure 4 is the job processing sequence diagram of the three machines. From the figure, it can be seen that the load of the machine is unbalanced, the load of M_1^p is much less than that of M_3^p . For this reason, a dynamic split strategy (DSS) is proposed, the J_{10} and J_{11} are moved to the M_1^p through DSS and thus balance the load of M_1^p and M_3^p . The detail of the DSS is described in Sect. 3.2. Although G_2' is split and the setup energy consumption is increased, but the load of the machine is balanced, the result is shown in Fig. 5.

According to the processing sequence of group $G'' = \{G_1'', G_2'', \dots, G_c''\}$ in the first stage, calculate the starting processing time of each group G_i'' , sort $G'' = \{G_1'', G_2'', \dots, G_c''\}$ from small to large according to the start processing time, then the processing sequence of

Fig. 4 The jobs processing sequence in stage one

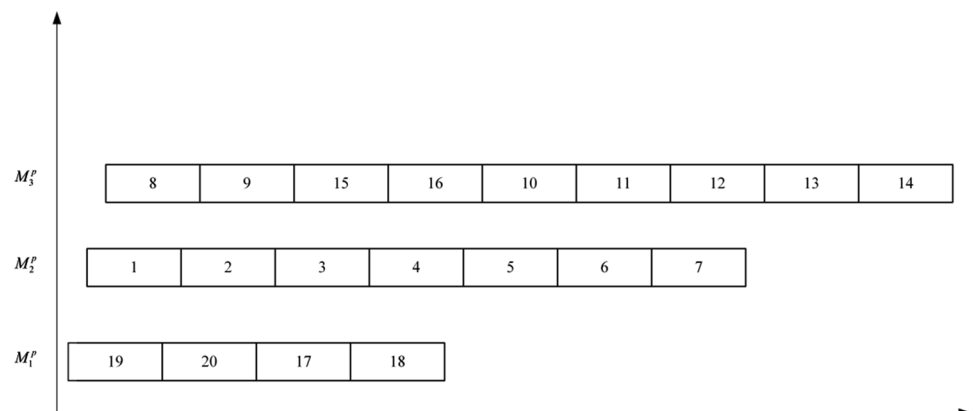
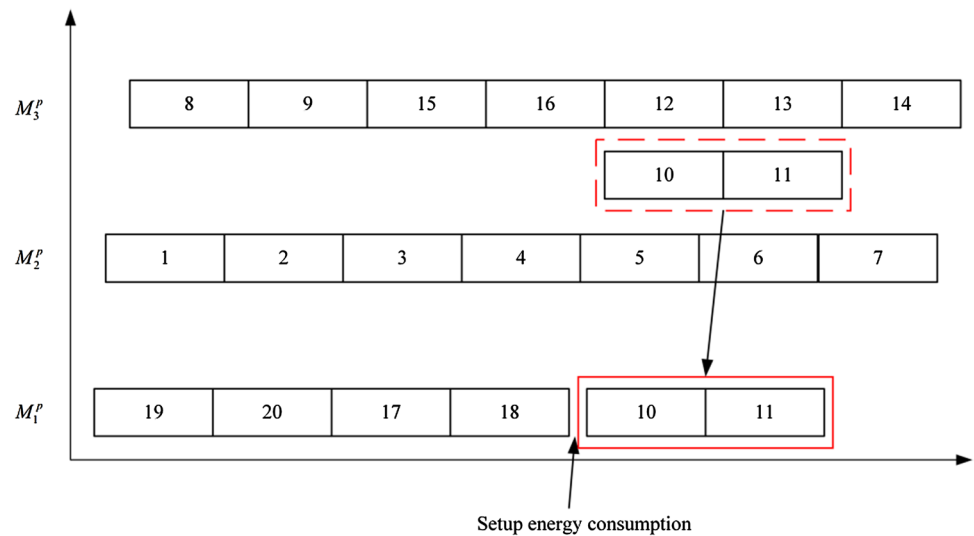


Fig. 5 The adjusted job processing sequence of stage one



$G'' = \{G''_1, G''_2, \dots, G''_{c'}\}$ in the second stage can be obtained. For example, according to the processing sequence of the first stage, the order of $G'' = \{G''_1, G''_2, \dots, G''_{c'}\}$ is $[8, 1, 4, 7, 6, 2, 5_1, 5_2, 3]$, 5_1 consists of $J10$ and $J11$, 5_2 consists of $J12$, $J13$ and $J14$, 5_1 and 5_2 are part of G''_5 and has the same process. According to the third layer chromosome and $[8, 1, 4, 7, 6, 2, 5_1, 5_2, 3]$, it can obtain the jobs processing sequence of M_1^c is $[13, 14, 17-20]$, and the jobs processing sequence of M_2^c is $[1-6]$, and the jobs processing sequence of M_3^c is $[7-12, 15, 16]$. Figure 5 shows the job processing sequence of the machines in the second stage. It can be seen from the Fig. 6, although the setup energy consumption is increased when move $J14$ from M_3^c to M_1^c via dynamic split strategy, but the load of the machines is more balanced, the result is shown as Fig. 7. In this paper, dynamic splitting strategy is used to achieve load balancing of the machines

with the minimum increment of setup energy consumption. It should be noted that when adjusting the job in stage two, the completion time of the job in stage one should be considered to ensure that the processing of the job follows the constraints of processing sequence.

3.2 Dynamic Split Strategy

It is easy to cause the load imbalance of the machines by scheduling jobs in groups as described above. The imbalance referred to here can be understood as that there are some jobs that can be processed earlier when it is moved to other machines for processing, such as $J14$ in Fig. 6. Therefore, the machine load balance here refers to that no job can be processed earlier by moving jobs to other machines. In this paper, a dynamic splitting strategy is developed to balance the load of the machines. The specific steps are as follows:

Fig. 6 The jobs processing sequence in stage two

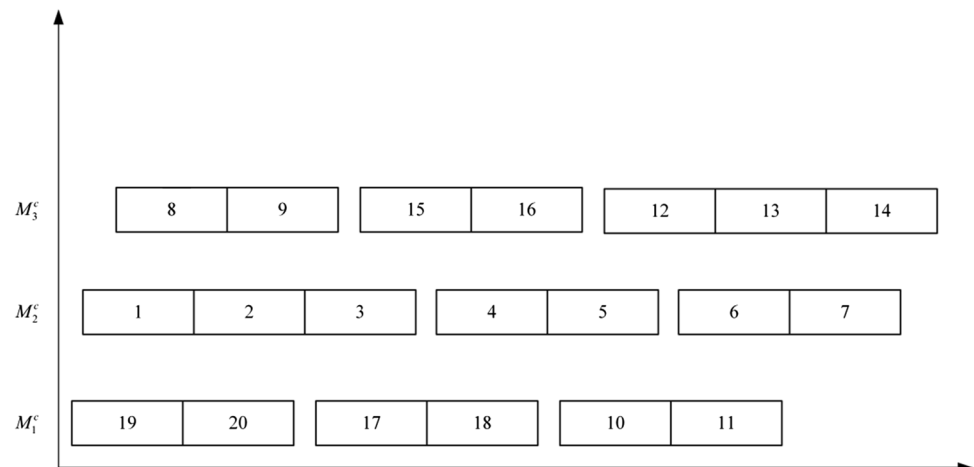
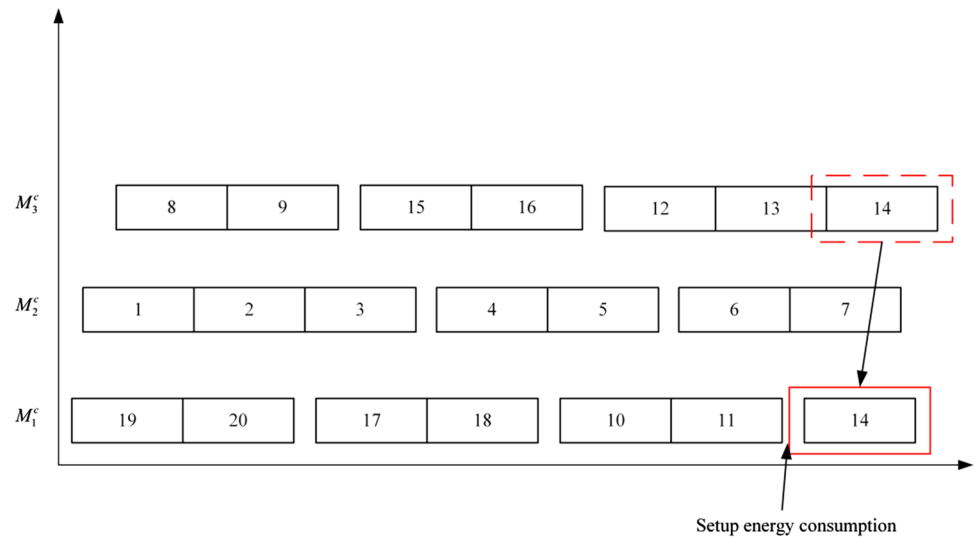


Fig. 7 The adjusted job processing sequence in stage two



Algorithm 1 Dynamic splitting strategy

01. Input: A feasible scheduling.
02. Output: The adjusted scheduling.
03. While true do:
04. Find the job with the largest start time in the last job of all machines (J_s), record the start time of J_s (Max_start) and the machine that process J_s (Max_start_m). Find the job with the smallest complete time in the last job of all machines (J_e), record the complete time of J_e (Min_end) and the machine that process J_e (Min_end_m).
05. If $Max_start > Min_end$:
06. Traverse the jobs on machine Max_start_m to find the first job with start time greater than Min_end (J_a). Move J_a to the machine Min_end_m , judge whether J_a can be processed in advance, if yes, then update the start time and completion time of the jobs after J_a on machine Max_start_m , and update the start time and complete time of the last job on machine Min_end_m . Otherwise, cancel the move, and find the next job with start time greater than Min_end on machine Max_start_m . Then, adjust the processing machine of the finding job according to the above method. Repeat the above steps until there is no job can be moved to other machines.
07. Else:
08. Terminate while loop.
09. End If
10. End While

3.3 Method of Diversity Population Generation

Generating diverse population is one of the conditions to ensure the global search ability of scatter search algorithm. A diverse population means that each individual has a large distance. There are many indicators to measure individual distance, suppose $D(x_1, x_2)$ is the distance between individual x_1 and x_2 , in this paper, $D(x_1, x_2)$ is set as the sum of different edges of two individuals. For example, given two individuals $x_1 = [1, 3, 5, 4, 2]$ and $x_2 = [5, 4, 1, 2, 3]$. The edges of x_1 are (1,3), (3,5), (5,4), (4,2), the edges of x_2 are (5,4), (4,1), (1,2), (2,3), so $D(x_1, x_2) = 3$. Because the third layer chromosome is fixed, only the first and second layers of chromosomes need to be generated. The proposed population generation method is as follows.

3.4 The Solution Improvement Method

Apply the improvement method to the solution can further improve the quality of the solution. The solution improvement method is generally designed based on the local search method. Although the complex method can improve the quality of the solution more effective, it will also increase the operation time. However, the too simple method is difficult to improve the quality of the solution. In this paper, a solution improvement method is designed based on variable neighborhood search algorithm. The key of variable neighborhood search algorithm is neighborhood action. According to the characteristics of chromosome coding, three kinds of neighborhood actions are designed in this paper. The first neighborhood action is to use the exchange, inversion and

Algorithm 2 The proposed population generation method	
01.	Input: population size N , p' and G' .
02.	Output: diversity population.
03.	For $i=1$ to N :
04.	If $i==1$:
05.	The first layer chromosome and the second layer chromosome of the new individual are randomly generated and add the new individual to the existing population.
06.	Else :
07.	Randomly generate 20 chromosomes, and the distance between each generated chromosome and the first layer chromosome of the existing individuals is calculated, and the chromosome with the largest distance is selected as the first layer chromosome of the new individual.
08.	For $j=1$ to p' do:
09.	Since each G'_j corresponds to a segment of the second layer chromosome, 20 segments are randomly generated for each G'_j , then the distance between each generated segment and the existing corresponding chromosome segment is calculated, and the segment with the largest distance is selected as the segment corresponding to G'_j in the second layer chromosome of the new individual.
10.	End For
11.	Add the new individual to existing populations.
12.	End If
13.	End For

insertion operators to disturb the first layer chromosome, so as to generate new individuals. The second neighborhood action is to use the exchange, inversion and insertion operators to disturb a segment of the second layer chromosome, which the segment is the segment of the second layer chromosome that correspond to the job group G'_i which is randomly selected from G' . The third neighborhood action is to combine the first two neighborhood actions. The following is the flow of the proposed solution improvement method.

3.6 The Solution Combination Method

According to the characteristics of coding, this paper designs three solution combination methods. In the first method, the first layer chromosome of the two solutions is crossed and the second layer and the third layer chromosome remain unchanged. In the second method, the second layer chromosome of the two solutions is crossed, and the first layer and third layer chromosome remain unchanged. The third method is to cross the first layer and second layer

Algorithm 3 The proposed solution improvement method

```

01. Input: a solution.
02. Output: the improved solution
03. For  $i=1:10$  do:
04.   While true do:
05.     Use the first neighborhood action to generate a new solution for the
current solution, and judge whether the new solution is improved. If there is
improvement, replace the current solution with the new solution and carry out the
next cycle. Otherwise, judge whether there is no improvement for 5 consecutive
times. If so, exit the cycle. Otherwise, continue the next cycle.
06.   End While
07.   While true do:
08.     Use the second neighborhood action to generate a new solution for the
current solution, and judge whether the new solution has been improved. If there is
improvement, replace the current solution with the new solution and skip to Line
04. Otherwise, judge whether there is no improvement for 5 consecutive times. If
so, exit the cycle, otherwise continue the next cycle.
09.   End While
10.   While true do:
11.     Use the third neighborhood action to generate a new solution for the
current solution, and judge whether the new solution is improved. If there is
improvement, replace the current solution with the new solution and jump to Line
04. Otherwise, judge whether there is no improvement for 5 consecutive times. If
so, skip to Line 03. Otherwise, continue the next cycle.
12.   End While
13. End For

```

3.5 Subset Generation Method

In the scatter search algorithm, the reference set Ref consists of two parts, Ref1 and Ref2, respectively. Where Ref1 stores high-quality solutions, while Ref2 stores diverse solutions. In this paper, two subsets are generated, the first subset $Subset1 = \{p_1, p_2\}$ is composed of two solutions selected by the tournament selection mechanism from Ref1. The second subset $Subset2 = \{p_3, p_4\}$ is also composed of two solutions, one is selected from Ref1 by the tournament selection mechanism, and the other is selected from Ref2 by the tournament selection mechanism.

chromosomes at the same time, and the third layer chromosome remain unchanged. The crossing method used in this paper is the linear order crossover (LOX) [40]. First, three methods are used to combine the solutions of subset one and subset two, and a total of six new solutions are obtained. Then the Pareto rank and crowding distance of each solution are calculated. Finally, according to the Pareto rank and crowding distance, the best one of the six new solutions is selected to inherit to the next generation.

Table 1 The details of 25 scheduling problems

Problems	Number of jobs	The number of process types in stage one	The number of process types in stage two
J1	200	20	100
J2	200	20	100
J3	200	20	100
J4	200	20	100
J5	200	20	100
J6	250	25	125
J7	250	25	125
J8	250	25	125
J9	250	25	125
J10	250	25	125
J11	300	30	150
J12	300	30	150
J13	300	30	150
J14	300	30	150
J15	300	30	150
J16	350	35	175
J17	350	35	175
J18	350	35	175
J19	350	35	175
J20	350	35	175
J21	400	40	200
J22	400	40	200
J23	400	40	200
J24	400	40	200
J25	400	40	200

3.7 Reference Set Updating Method

Through the solution combination method and the solution improvement method, a new population *popn* will be obtained. Based on the old reference set and the new population, the new reference set can be obtained using the reference set updating method. The developed reference set update method is as follows: (1) Merge the new population *popn* and Ref, remove the repeated solution to get the remaining solution *popu*, and calculate the Pareto rank and crowding distance of each solution of *popu*. (2) Assume the size of Ref1 is *Ref1_num*, the size of Ref2 is *Ref2_num*, the size of *popu* is *popu_num*, if the *popu_num* less than *Ref1_num + Ref2_num*, then *Ref1_num + Ref2_num - popu_num* solutions are generated by diversity generation method and added to *popu* to obtain a new population *popun*, and calculate the Pareto rank and crowding distance of each solution of *popun*. Select *Ref1_num* best solutions as the new Ref1 and *Ref2_num* solutions with the largest distance as the new Ref2 from *popun*. Otherwise, select *Ref1_num* best solutions as the new Ref1 and *Ref2_num*

solutions with the largest distance as the new Ref2 from *popu*.

4 Experiment

To prove the superiority of the proposed algorithm, this paper compares the proposed algorithm with NSGA-II, which is an outstanding multi-objective optimization algorithm. All algorithms are written in MATLAB 2015b and the operating environment: Windows 10, 8G RAM, 2.11 GHz Intel Core i7-8650U.

4.1 Experiment Data

Because of the particularity of the problem, there is no standard data set available. In this paper, based on the real data of tissue paper enterprise, Monte Carlo method is used to generate the required experimental data. A total of 25 scheduling problems are used to test the performance of the proposed algorithm, and the information contained in each problem is shown in Table 1. The number of process types indicates the process selection range of the jobs of the scheduling problem in each stage. For example, the number of process types in the first stage is 20, indicating that the process in the first stage of each job of the scheduling problem is randomly selected from 20 processes. Similarly, the number of process types in the second stage is 100, indicating that the process in the second stage of each job of the scheduling problem is randomly selected from 100 processes. The number of machines for both the first and second stages is set to five. The size of each job is randomly generated from [120,000,180,000], and the processing speed of each machine is randomly generated from [900,1200], where the machines have the same speed when processing the job with same processes. The processing energy consumption per unit time of machine in stage one and stage two is generated from [1200,1500] and [120,150], respectively. The machines have the same processing energy consumption per unit time when processing the jobs with the same processes. The setup energy consumption per unit time of machine stage one and stage two is generated from [800,1000] and [80,100], respectively. The setup time in stage one and stage two is generated from [10,60]. The setup energy consumption per unit time and setup time relate to the processes of the two consecutive jobs and machine.

4.2 Evaluation Methods

The multi-objective optimization algorithm obtains the non-dominated solution set. The quality of the non-dominated solution set can be measured from the uniformity, universality and the degree of approaching the optimal solution.

At present, many evaluation methods only evaluate one of the indexes of the solution. The hypervolume (HV) index is one of the rare indexes that can evaluate the uniformity, universality and the degree of approaching the optimal solution at the same time [41]. In this paper, hypervolume index is used to evaluate the quality of solution. The larger the hypervolume index is, the better the quality of solutions is. In addition, the evolutionary algorithm is stochastic, and it is not reliable to compare the differences between the algorithms only through one run. In this paper, each algorithm is run 20 times, and t-test is used to test whether there is significant difference between the two algorithms. The significance level is set to 95%, and the corresponding p value is 0.05, that is, when p value is less than 0.05, it can be considered that there is significant difference between the two algorithms. All algorithms have the same population size and number of iterations, that is, the population size is set to 100, and the maximum number of iterations is set to 100. In addition, the crossover and mutation rates of NSGA-II are set to 0.8 and 0.2, respectively.

4.3 Results and Discussion

To prove the effectiveness of group technology in reducing energy consumption, 25 scheduling problems are solved by NSGA-II without group technology, and then 25 scheduling problems are solved by NSGA-II with group technology (NSGA-II-GT). Table 2 shows the average energy consumption and the average makespan obtained by NSGA-II and NSGA-II-GT. It can be seen from Table 2 that not only the energy consumption of NSGA-II-GT is lower than that of NSGA-II, but also the makespan of NSGA-II-GT is smaller than that of NSGA-II. This is because after grouping the same type of jobs, a lot of setup time is reduced, thus the makespan is reduced. Figure 8 shows the energy consumption and makespan reduction rate of NSGA-II-GT compared with NSGA-II, the energy consumption reduction rate can reach 20.25%, while the makespan reduction rate can reach 13.68%. The average energy consumption reduction rate and makespan reduction rate are 19.23% and 13.06%, respectively. It can be seen from Table 2 and Fig. 8 that group

Table 2 The average energy consumption and the average makespan obtained by NSGA-II and NSGA-II-GT

Problems	Energy consumption		Metric	Makespan		Metric
	NSGA-II	NSGA-II-GT		NSGA-II	NSGA-II-GT	
J1	834,450	666,911	+	7707	6728	+
J2	830,510	671,116	+	7702	6698	+
J3	841,519	672,746	+	7845	6828	+
J4	826,057	658,766	+	7737	6721	+
J5	837,281	668,368	+	7698	6701	+
J6	1,042,808	847,716	+	9619	8350	+
J7	1,049,222	845,061	+	9678	8459	+
J8	1,057,701	857,602	+	9626	8359	+
J9	1,064,636	863,027	+	9653	8332	+
J10	1,054,478	855,426	+	9671	8440	+
J11	1,251,866	1,011,555	+	11,488	9996	+
J12	1,274,342	1,022,181	+	11,611	10,092	+
J13	1,262,525	1,025,688	+	11,546	9996	+
J14	1,270,066	1,022,274	+	11,577	10,039	+
J15	1,263,347	1,019,270	+	11,588	10,093	+
J16	1,489,374	1,209,501	+	13,524	11,779	+
J17	1,484,925	1,212,374	+	13,513	11,726	+
J18	1,495,354	1,210,407	+	13,632	11,868	+
J19	1,479,235	1,195,657	+	13,487	11,709	+
J20	1,478,787	1,198,872	+	13,527	11,798	+
J21	1,700,021	1,383,907	+	15,349	13,331	+
J22	1,700,846	1,374,400	+	15,411	13,364	+
J23	1,688,963	1,362,222	+	15,411	13,382	+
J24	1,721,091	1,401,151	+	15,507	13,462	+
J25	1,704,244	1,371,262	+	15,420	13,421	+

"+" means NSGA-II-GT is better than NSGA-II, "-" means NSGA-II-GT is worse than NSGA-II, "=" means there is no significant difference between NSGA-II-GT and NSGA-II. Bold indicates the winner

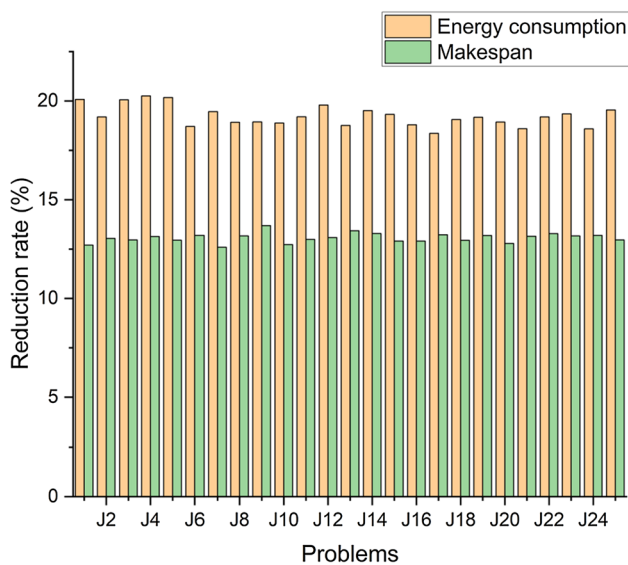


Fig. 8 The energy consumption reduction rate and makespan reduction rate

technology has significant effect on reducing energy consumption and makespan.

Group technology merges all kinds of the same jobs together, which may reduce the freedom of scheduling and leads to the imbalance of machine load. Figure 9 is the Gantt chart of a solution obtained by NSGA-II-GT. It can be seen from the figure that the load of both the first stage machine and the second stage machine is unbalanced, which means that some jobs can be processed in advance, such as job 147, 139, 130, etc. Besides, it can be seen that the load unbalance of stage one is higher than that of stage two. The reason is that there are more jobs in each group in stage one, which leads to lower freedom of scheduling. It should be noted that the process of the job is uniformly distributed in this study. If not, the size of each group is more different, which is more likely to lead to load imbalance. For this reason, a dynamic splitting strategy is designed to reduce the machine imbalance caused by group technology. Figure 10 shows the Gantt chart of the solution obtained by NSGA-GT with dynamic split strategy (NSGA-GTDSS), it can be seen from the figure that some jobs have been started earlier, and some additional setup time and setup energy consumption may be increased at the same time. To analyze the performance of the dynamic splitting strategy, for each problem, 1000 individuals are randomly generated and then decoded by decoding scheme with dynamic splitting strategy (DS-DSS) and the decoding scheme with no dynamic splitting strategy (DS-NDSS), respectively. Figures 11 and 12 show the average makespan and energy consumption of DS-DSS and DS-NDSS. Figure 11 shows that the makespan of DS-DSS is smaller than that of DS-NDSS, but Fig. 12 shows that the energy consumption of DS-DSS is higher than that of

DS-NDSS. Figure 13 shows the degree of unbalance of DS-DSS and DS-NDSS, it can be seen the degree of unbalance of DS-DSS less than 4%, while the degree of unbalance of DS-NDSS greater than 10% and the highest is about 20%. From Fig. 13, it can be concluded that the dynamic splitting strategy can effectively balance the load of the machines. Although dynamic splitting strategy leads to the increase of energy consumption, but the makespan is reduced, and the machine load is more balanced.

To make a fair comparison between NSGA-II and IMSS, IMSS and NSGA-II adopt group technology and dynamic splitting strategy simultaneously (IMSS-GTDSS, NSGA-II-GTDSS). Because the solutions obtained by NSGA-II-GTDSS and IMSS-GTDSS are non-dominated solution sets, the HV index is used to measure the quality of the solutions of the two algorithms, and analyze the difference between the two algorithms through t-test. Table 3 lists the HV metrics of the solutions obtained by the two algorithms and the result of the t-test. From Table 3, it can be seen that the HV index of the solutions obtained by IMSS-GTDSS algorithm is larger than that of NSGA-II-GTDSS, indicating that the solutions obtained by IMSS-GTDSS have better quality. The p value of all t-tests is less than 0.05, which indicates that there is a significant difference between IMSS-GTDSS and NSGA-II-GTDSS. Table 4 shows the average energy consumption and the average makespan of NSGA-II-GTDSS and IMSS-GTDSS, it can be seen from the table that the average makespan obtained by IMSS-GTDSS are less than those obtained by NSGA-II-GTDSS, and the most of average energy consumption obtained by IMSS-GTDSS are less than those obtained by NSGA-II-GTDSS. There is no energy consumption and makespan obtained by NSGA-II-GTDSS is less than that obtained by IMSS-GTDSS simultaneously. Figure 14 shows the energy consumption and makespan reduction rate of IMSS-GTDSS compared with NSGA-II. It can be seen from the figure that the energy consumption can be reduced by 20.18% and the makespan can be reduced by 15.64%. The average energy consumption reduction rate and makespan reduction rate are 19.24% and 15.07%. Compared with NSGA-II-GT, the energy consumption reduction rate is similar, but the makespan reduction rate is increased by about 2%, and it is mentioned that the dynamic splitting strategy has a significant effect on the load balance of the machine. Thus, the dynamic splitting strategy can be considered in decoding. Moreover, in order to evaluate the robustness of the proposed method, the coefficient of variation (CV) is used to evaluate the discrete degree of the solution. Figure 15 shows the CV of HV index of NSGA-II-GTDSS and IMSS-GTDSS. It can be seen from the figure that most of the CV of IMSS-GTDSS is smaller than that of NSGA-II-GTDSS, indicating that the solutions of IMSS-GTDSS algorithm are less discrete and the IMSS-GTDSS has better robustness than NSGA-II-GTDSS.

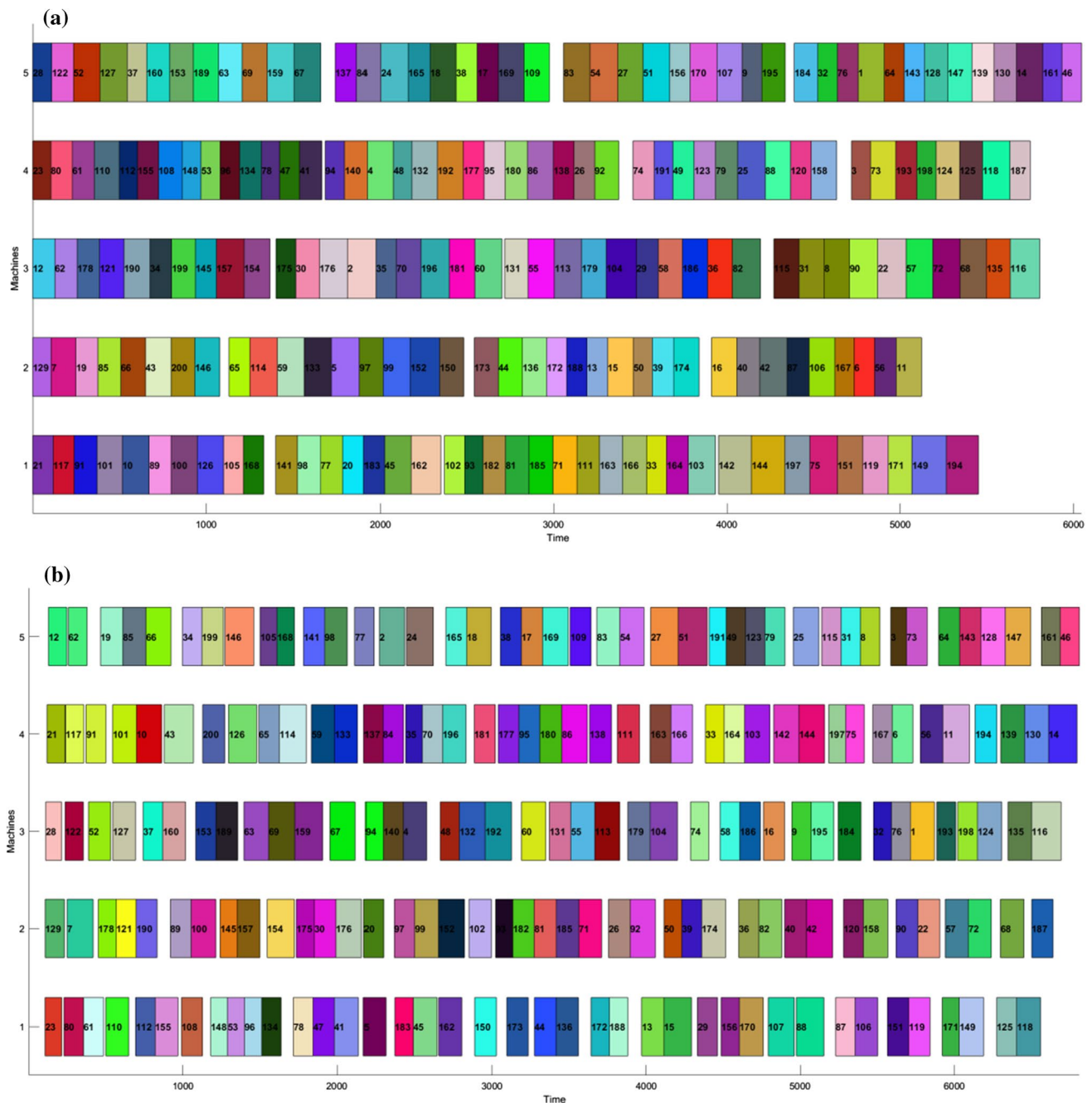


Fig. 9 The Gantt chart of the solution of NSGA-II-GT. **a, b** correspond to the Gantt chart of the first stage and the second stage, respectively

5 Conclusions

In this paper, the problem of energy-aware scheduling of two-stage flexible flow shop considering inconsistency of the process differences of jobs is studied. Group technology is used to reduce setup energy consumption, and dynamic split strategy is used to solve the problem of unbalanced machine load caused by group technology. According to the

characteristics of the research problem, an improved multi-objective scatter search algorithm is proposed to solve the problem. The experimental results show that the group technology can not only reduce the energy consumption, but also reduce the makespan, and the dynamic split strategy can effectively balance the load of the machine. Finally, by comparing with the existing excellent algorithms, it shows



Fig. 10 The Gantt chart of the solution of NSGA-II-GTDSS. **a**, **b** correspond to the Gantt chart of the first stage and the second stage, respectively

that the improved multi-objective scatter search algorithm has better performance.

The personalized needs of consumers are constantly improving. The traditional manufacturing systems are bound to seek transformation and upgrading. The manufacturing system of enterprises must develop more advanced

technology, which can effectively improve production efficiency and reduce costs on the premise of meeting the needs of consumers. This study can provide a reference way for enterprises to reduce energy consumption cost and improve production efficiency in the operation level under the multi-variety and small batch manufacturing mode.

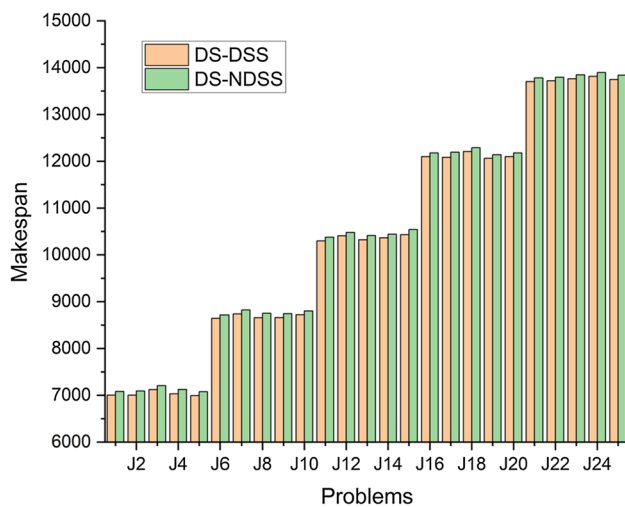


Fig. 11 The average makespan of DS-DSS and DS-NDSS

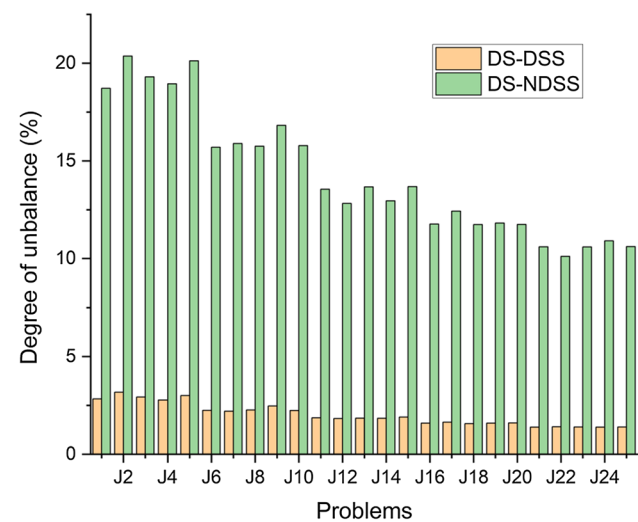


Fig. 13 The degree of unbalance of DS-DSS and DS-NDSS

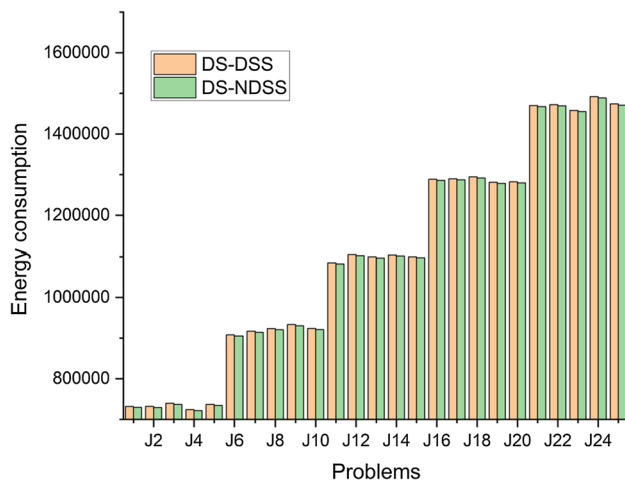


Fig. 12 The average energy consumption of DS-DSS and DS-NDSS

Job delivery time is one of the important factors affecting production scheduling, and future work will consider the group strategy with job delivery time. The energy consumption cost is closely related to the electricity price, and the optimization of energy consumption cost under time-of-use electricity tariffs is also one of the future works. In addition, in practical applications, the speed of the algorithm is a non-negligible problem, and how to speed up the algorithm in solving large-scale problems is a challenge.

Table 3 The HV metrics obtained by NSGA-II-GTDSS and IMSS-GTDSS and the result of the t-test

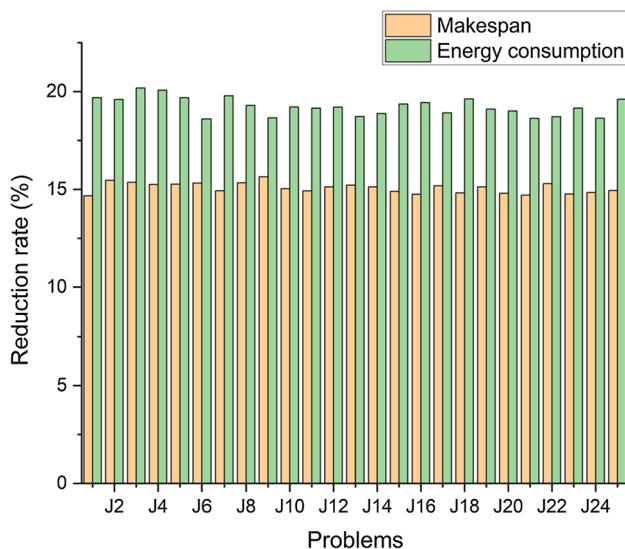
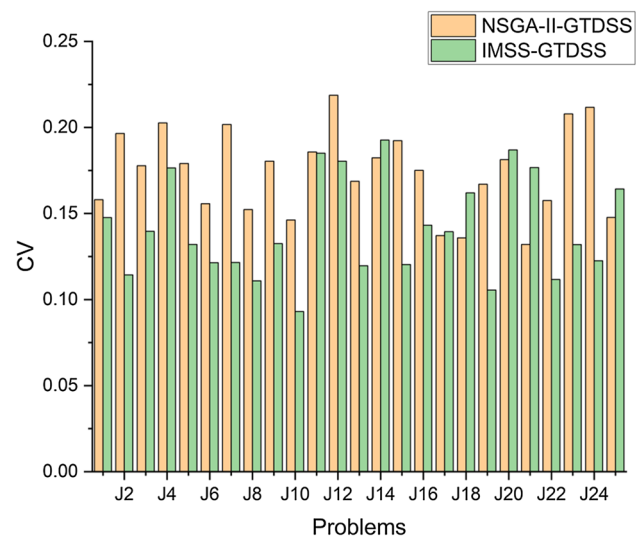
Problems	NSGA-II- GTDSS	IMSS-GTDSS	Metric
J1	2.19E+07	2.87E+07	+
J2	2.25E+07	3.11E+07	+
J3	2.24E+07	3.03E+07	+
J4	1.36E+07	2.36E+07	+
J5	1.67E+07	2.52E+07	+
J6	2.99E+07	3.96E+07	+
J7	3.01E+07	4.42E+07	+
J8	2.87E+07	3.86E+07	+
J9	1.83E+07	2.74E+07	+
J10	3.60E+07	4.90E+07	+
J11	2.08E+07	2.77E+07	+
J12	2.35E+07	3.05E+07	+
J13	3.60E+07	5.15E+07	+
J14	2.25E+07	2.94E+07	+
J15	3.41E+07	4.80E+07	+
J16	4.70E+07	6.07E+07	+
J17	5.06E+07	7.25E+07	+
J18	3.38E+07	5.94E+07	+
J19	4.37E+07	5.72E+07	+
J20	4.10E+07	6.04E+07	+
J21	3.24E+07	5.70E+07	+
J22	5.74E+07	8.51E+07	+
J23	4.65E+07	7.01E+07	+
J24	5.57E+07	8.10E+07	+
J25	5.58E+07	8.43E+07	+

"+" means IMSS-GTDSS is better than NSGA-II-GTDSS, "-" means IMSS-GTDSS is worse than NSGA-II-GTDSS, "=" means there is no significant difference between IMSS-GTDSS and NSGA-II-GTDSS. Bold indicates the winner

Table 4 The average energy consumption and the average makespan obtained by NSGA-II-GTDSS and IMSS-GTDSS

Problems	Makespan			Energy consumption		
	NSGA-II-GTDS	IMSS-GTDSS	Metric	NSGA-II-GTDS	IMSS-GTDSS	Metric
J1	6714	6576	+	672,404	670,176	=
J2	6672	6511	+	672,429	667,819	+
J3	6817	6640	+	676,225	671,685	+
J4	6715	6557	+	667,581	660,282	+
J5	6692	6523	+	671,641	672,427	=
J6	8335	8145	+	852,751	848,866	+
J7	8419	8233	+	845,157	841,643	+
J8	8348	8150	+	857,827	853,628	+
J9	8328	8143	+	865,993	866,018	=
J10	8403	8217	+	857,387	851,939	+
J11	9949	9774	+	1,003,730	1,012,080	-
J12	10,080	9855	+	1,031,200	1,029,610	=
J13	9984	9789	+	1,027,110	1,026,130	=
J14	10,019	9826	+	1,027,650	1,030,270	=
J15	10,106	9862	+	1,022,650	1,018,730	=
J16	11,746	11,529	+	1,201,700	1,199,850	=
J17	11,734	11,461	+	1,205,840	1,204,110	=
J18	11,853	11,612	+	1,211,460	1,201,950	+
J19	11,707	11,447	+	1,198,710	1,196,670	=
J20	11,737	11,525	+	1,193,380	1,197,710	=
J21	13,357	13,092	+	1,386,680	1,383,250	=
J22	13,343	13,055	+	1,380,890	1,382,580	=
J23	13,392	13,136	+	1,366,540	1,365,500	=
J24	13,446	13,206	+	1,404,230	1,400,350	=
J25	13,424	13,116	+	1,375,460	1,370,130	=

"+" means IMSS-GTDSS is better than NSGA-II-GTDSS, "-" means IMSS-GTDSS is worse than NSGA-II-GTDSS, "=" means there is no significant difference between IMSS-GTDSS and NSGA-II-GTDSS. Bold indicates the winner

**Fig. 14** The energy consumption reduction rate and makespan reduction rate of IMSS-GTDSS**Fig. 15** The CV of IMSS-GTDSS and NSGA-II-GTDSS

Acknowledgements The authors are very grateful to those who contributed to the study.

Author contributions Conceptualization, methodology, writing, funding acquisition, and data processing and result analysis were performed by ZZ. Funding acquisition and supervision were performed by LG. Software, review, and editing were performed by ZH.

Funding This work was supported by Natural Science Foundation of Guangdong Province (No. 2020A1515011468); Guangdong University Scientific Research Project (No. 2019KTSCX189); Wuyi University Youth Team Research Project (No. 2019td10).

Data availability The datasets generated in this study are available from the corresponding authors.

Declarations

Conflict of interest The authors declare that they have no competing interests.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

Ethical Approval Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Liu, Y.Y., Liao, X.Y., Zhang, R.: An enhanced MOPSO algorithm for energy-efficient single-machine production scheduling. *Sustainability* **11**, 5381 (2019)
2. Wu, X.L., Shen, X.L., Li, C.B.: The flexible job-shop scheduling problem considering deterioration effect and energy consumption simultaneously. *Comput. Ind. Eng.* **135**, 1004–1024 (2019)
3. Zheng, X., Zhou, S., Xu, R., Chen, H.: Energy-efficient scheduling for multi-objective two-stage flow shop using a hybrid ant colony optimisation algorithm. *Int. J. Prod. Res.* **12**, 1–18 (2019)
4. Oukil, A., El-Bouri, A., Emrouznejad, A.: Energy-aware job scheduling in a multi-objective production environment—an integrated DEA-OWA model. *Comput. Ind. Eng.* **168**, 108065 (2022)
5. Hong, Z., Zeng, Z., Gao, L.: Energy-efficiency scheduling of multi-cell manufacturing system considering total handling distance and eligibility constraints. *Comput. Ind. Eng.* **151**, 106998 (2021)
6. Wilson, A.D., King, R.E., Hodgson, T.J.: Scheduling non-similar groups on a flow line: multiple group setups. *Robot. Cim-Int. Manuf.* **20**, 505–515 (2004)
7. Alireza, K., Fariborz, J., Morteza, B.: Integrating sequence-dependent group scheduling problem and preventive maintenance in flexible flow shops. *Int. J. Adv. Manuf. Tech.* **77**, 173–185 (2015)
8. Ebrahimi, M., Ghomi, S.M.T.F., Karimi, B.: Hybrid flow shop scheduling with sequence dependent family setup time and uncertain due dates. *Appl. Math. Model.* **38**, 2490–2504 (2014)
9. Feng, H.X., Xi, L., Xiao, L., Xia, T., Pan, E.: Imperfect preventive maintenance optimization for flexible flowshop manufacturing cells considering sequence-dependent group scheduling. *Reliab. Eng. Syst. Safe.* **176**, 218–229 (2018)
10. Keshavarz, T., Salmasi, N.: Makespan minimisation in flexible flowshop sequence-dependent group scheduling problem. *Int. J. Prod. Res.* **51**, 6182–6193 (2013)
11. Keshavarz, T., Salmasi, N., Varmazyar, M.: Minimizing total completion time in the flexible flowshop sequence-dependent group scheduling problem. *Ann. Oper. Res.* **226**, 351–377 (2015)
12. Shahvari, O., Salmasi, N., Logendran, R., Abbasi, B.: An efficient tabu search algorithm for flexible flow shop sequence-dependent group scheduling problems. *Int. J. Prod. Res.* **50**, 4237–4254 (2012)
13. Costa, A., Cappadonna, F.V., Fichera, S.: Minimizing makespan in a Flow Shop Sequence Dependent Group Scheduling problem with blocking constraint. *Eng. Appl. Artif. Intell.* **89**, 103413 (2020)
14. Yuan, S., Li, T., Wang, B.: A co-evolutionary genetic algorithm for the two-machine flow shop group scheduling problem with job-related blocking and transportation times. *Expert. Syst. Appl.* **152**, 113360 (2020)
15. Dorfeshan, Y., Tavakkoli-Moghaddam, R., Mousavi, S.M., Vahedi-Nouri, B.: A new weighted distance-based approximation methodology for flow shop scheduling group decisions under the interval-valued fuzzy processing time. *Appl. Soft Comput.* **91**, 106248 (2020)
16. Cheng, C.Y., Pourhejazy, P., Ying, K.C., Liao, Y.H.: New benchmark algorithms for no-wait flowshop group scheduling problem with sequence-dependent setup times. *Appl. Soft Comput.* **111**, 107705 (2021)
17. Yuan, S., Li, T., Wang, B.: A discrete differential evolution algorithm for flow shop group scheduling problem with sequence-dependent setup and transportation times. *J. Intell. Manuf.* **32**(2), 427–439 (2021)
18. Shahvari, O., Logendran, R.: Hybrid flow shop batching and scheduling with a bi-criteria objective. *Int. J. Prod. Econ.* **179**, 239–258 (2016)
19. Feng H., Tan C., Xia T., Pan E., Xi L.: Joint optimization of preventive maintenance and flexible flowshop sequence-dependent group scheduling considering multiple setups. *Eng. Optimiz.* **51**, 1529–1546 (2019)
20. Jain, A., Jain, A.: An approach for optimisation of flexible flow shop group scheduling with sequence dependent set-up time and preventive maintenance. *Int. J. Comput. Aided Eng. Technol.* **16**(1), 40–66 (2022)
21. Adressi, A., Hassanpour, S.T., Azizi, V.: Solving group scheduling problem in no-wait flexible flowshop with random machine breakdown. *Decis. Sci. Lett.* **5**, 157–168 (2016)
22. Bozorgirad, M.A., Logendran, R.: Bi-criteria group scheduling in hybrid flowshops. *Int. J. Prod. Econ.* **145**, 599–612 (2013)

23. Karimi, N., Zandieh, M., Karamooz, H.R.: Bi-objective group scheduling in hybrid flexible flowshop: a multi-phase approach. *Expert Syst. Appl.* **37**, 4024–4032 (2010)
24. Zandieh, M., Karimi, N.: An adaptive multi-population genetic algorithm to solve the multi-objective group scheduling problem in hybrid flexible flowshop with sequence-dependent setup times. *J. Intell. Manuf.* **22**, 979–989 (2010)
25. Nekoiemehr, N., Selvarajah, E., Zhang, G.: Scheduling of jobs with cross families in two stage manufacturing systems. *Int. J. Prod. Econ.* **167**, 88–96 (2015)
26. Andrés, C., Albarracín, J.M., Tormo, G., Vicens, E., García-Sabater, J.P.: Group technology in a hybrid flowshop environment: a case study. *Eur. J. Oper. Res.* **167**, 272–281 (2015)
27. Costa, A., Cappadonna, F.A., Fichera, S.: A hybrid genetic algorithm for minimizing makespan in a flow-shop sequence-dependent group scheduling problem. *J. Intell. Manuf.* **28**, 1269–1283 (2015)
28. Huang, X.: Bicriterion scheduling with group technology and deterioration effect. *J. Appl. Math. Comput.* **60**, 455–464 (2018)
29. Li, W.X., Zhao, C.L.: Single machine scheduling problem with multiple due windows assignment in a group technology. *J. Appl. Math. Comput.* **48**, 477–494 (2015)
30. Liou, C.D., Hsieh, Y.C.: A hybrid algorithm for the multi-stage flow shop group scheduling with sequence-dependent setup and transportation times. *Int. J. Prod. Econ.* **170**, 258–267 (2015)
31. Wang, J.J., Liu, Y.J.: Single-machine bicriterion group scheduling with deteriorating setup times and job processing times. *Appl. Math. Comput.* **242**, 309–314 (2014)
32. Yu, C., Ji, Y., Qi, G., Gu, X., Tao, L.: Group-based production scheduling for make-to-order production. *J. Intell. Manuf.* **26**, 585–600 (2013)
33. Goli, A., Keshavarz, T.: Just-in-time scheduling in identical parallel machine sequence-dependent group scheduling problem. *J. Ind. Manag. Optim.* (2021). <https://doi.org/10.3934/jimo.2021124>
34. Liao, B., Song, Q., Pei, J., Yang, S., Pardalos, P.M.: Parallel-machine group scheduling with inclusive processing set restrictions, outsourcing option and serial-batching under the effect of step-deterioration. *J. Global. Optim.* **78**(4), 717–742 (2020)
35. Liao, B., Wang, H., Zhu, X., Yang, S., Pardalos, P.M.: Improved DE search for competing groups scheduling with deterioration effects. *Optim. Lett.* **15**(2), 469–494 (2021)
36. Wang, J.B., Jia, X., Yan, J.X., Wang, S.H., Qian, J.: Single machine group scheduling problem with makespan objective and a proportional linear shortening. *RAIRO-Oper. Res.* **56**(3), 1523–1532 (2022)
37. Wang, W., Wang, J.J., Wang, J.B.: Solution algorithms for single-machine group scheduling with learning effect and convex resource allocation. *Complexity* **2021**, 6615824 (2021)
38. He, X., Pan, Q.K., Gao, L., Wang, L., Suganthan, P.N.: A greedy cooperative co-evolutionary algorithm with problem-specific knowledge for multi-objective flowshop group scheduling problems. *IEEE T. Evolut. Comput.* (2021). <https://doi.org/10.1109/TEVC.2021.3115795>
39. Glover F.: A template for scatter search and path relinking. In: *European Conference on Artificial Evolution*. Springer, Berlin (1997).
40. Wu, X.L., Sun, Y.L.: A green scheduling algorithm for flexible job shop with energy-saving measures. *J. Clean. Prod.* **172**, 3249–3264 (2018)
41. While, L., Hingston, P., Barone, L., Huband, S.: A faster algorithm for calculating hypervolume. *IEEE Trans. Evol. Comput.* **10**, 29–38 (2006)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.