A survival duration-guided NSGA-III for sustainable flexible job shop scheduling problem considering dual resources

Li Hongyu | Wu Xiuli

Abstract
Considering the increasing concern on sustainable development from manufacturers, focus is given to three kinds of indicators of sustainable development, that is, economy, environment, and society, and schedule two types of resource, that is machines and workers, simultaneously in the classical flexible job shop scheduling problem. The authors define it as a sustainable flexible job shop scheduling problem considering dual resources (SFJSPCDR). First, a model of the SFJSPCDR is formulated to optimise the makespan, the energy consumption, and the ergonomic risk simultaneously. Second, an improved survival duration-guided NSGA-III algorithm (SDG-NSGA-III) is proposed to solve SFJSPCDR. The survival duration of each individual determines whether it takes part in generating offspring. In order to balance the energy consumption and ergonomic risk while minimising the makespan, a double-low decoding algorithm is proposed, which is composed of two decoding algorithms. Cross-generation selection is employed with the non-dominated sorting, and the next-generation population is selected according to the reference point-based selection strategy. In addition, a restart strategy is also integrated to improve the exploration and exploitation performance of the SDG-NSGA-III algorithm. Finally, a group of experiments are carried out and the results prove the effectiveness of the proposed algorithm.

1 | INTRODUCTION
The concept of sustainable development was first proposed in 1988. It emphasises that economic development should not only meet the needs of contemporary people, but also should not damage the interests of the future generations. Sustainable development promotes a balance between the social, environmental and economic development in any organisation [1]. According to a survey [2], economic indicators have always been the focus of development, and people's attention to environmental indicators and social indicators has also increased by 62% and 52%, respectively, in recent years. Under this situation, more and more industries have been changing to sustainable development [2].

As one of the pillar industries, manufacturing industry also has been paying increasing attention to sustainable development. Taking production planning and scheduling as an example, indicators such as makespan, tardiness, and the cost of workers and machines can be classified as economic indicators, indicators such as the carbon emission, energy consumption, and noise emissions can be classified as environmental indicators, and indicators such as people's occupational risk, ergonomic risk, work intensity, satisfaction, and so on can be classified as social indicators. Since no consensus has yet been reached about the concept of social indicators, there are few studies on production scheduling problems considering social indicators [3]. However, many scholars believe that social indicators should be considered in making production planning and scheduling decisions [4–6]. In a manufacturing system, there is not only the machines resource, but also the workers resource. It is important for both kinds of resource to cooperate to complete the processing tasks [7]. Therefore, the authors consider the three sustainable indicators jointly. Makespan is taken as the economy indicator, energy consumption is taken as the environment indicator; and ergonomic risk is taken as the society indicator. At the same time, in order to be more in line with the actual workshop environment, the flexibility of dual resources (machine and worker) also are
considered in this study. The authors define it as the sustainable flexible job shop scheduling problem considering dual resources (SFJSPCDR).

Compared to the classic flexible job shop scheduling problem [8, 9], the solution space of the SFJSPCDR increases greatly. Since the FJSP has been proved to be NP-hard, SFJSPCDR is definitely NP-hard, too. Before the authors' approach is proposed, the related literature is reviewed, including studies on the multi-resources scheduling problem and those on the sustainable scheduling problem.

In the multi-resources scheduling problem, the constraints involved in studies mainly include machines, workers, and fixtures. Heuristics or meta-heuristics are usually employed to solve the multi-resource scheduling problem. For example, Lei et al. [10] proposed a new decoding method for the dual resource scheduling problem. In their subsequent research [11], based on this decoding method, the algorithm was improved with the simulated annealing to obtain a better solution; Li et al. [12] employed a branch population genetic algorithm; Gao et al. [13] employed the shuffled multi-swarm micro-migrating birds optimisation (SM²-MBO) algorithm; Zheng et al. [14] employed a knowledge-guided fruit fly optimisation algorithm; Yazdani et al. [15] employed the evolutionary algorithm; and Wu et al. [16] employed the NSGA-II algorithm to explore the dual resource scheduling problem.

For the sustainable scheduling problem, according to one study [17], there are not enough studies considering all three sustainable indicators simultaneously. Gong et al. [3] considered the makespan as the economic indicator, the green cost as the environmental indicator, and the worker's operating cost as the social indicator to formulate the sustainable flexible job shop scheduling problem. Coca et al. [18] considered the completion time as the economic indicator, carbon dioxide emission, water consumption, metal waste, and another four factors as the environmental indicators, and work intensity, environmental noise, environmental temperature, and vibration amplitude as the social indicators. Gong et al. [19] employed the NSGA-III algorithm to solve the sustainable scheduling problem. Wu et al. [20] employed a multi-objective hybrid pigeon-inspired optimisation and simulated annealing (MOHPIOSA) algorithm to solve the sustainable problem.

In summary, there are few studies that consider all three sustainable indicators simultaneously, and even fewer studies combine the dual resource constraints with sustainable indicators in studying the scheduling problem. Therefore, the sustainable flexible job shop scheduling problem considering dual resources (SFJSPCDR) is studied here. The main contributions of this study are as follows: (1) an optimisation model for the SFJSPDR is formulated, (2) an improved survival duration guide NSGA-III algorithm (SDG-NSGA-III) is proposed, (3) a double-low decoding algorithm is developed, and (4) a new crossover method is proposed to balance the exploration and exploitation performance of SDG-NSGA-III.

Section 2 describes the SFJSPCDR. Section 3 formulates the SFJSPCDR. Section 4 proposes the survival duration-guided NSGA-III algorithm (SDG-NSGA-III) for the SFJSPCDR. Section 5 reports the case study, and Section 6 concludes the study.

2 | DESCRIPTION OF THE SFJSPCDR

The SFJSPCDR can be described as follows: there are \( n \) jobs, \( w \) workers, and \( m \) machines. Each job is composed of \( n_a \) operations, and the processing of each operation \( O_{ab} \) needs two kinds of resource, that is, machines and workers. Define the available machines for \( O_{ab} \) as \( M_{ab} \subset \{1, 2, \ldots, m\} \) and the available workers who can operate the machine \( c \) as \( W_{abc} \subset \{1, 2, \ldots, w\} \) for operation \( O_{ab} \). The processing time of operation \( O_{ab} \) is different with the different machine \( c \) and worker \( d \). The risk for workers is different when operating different machines. Assume a machine has two states: the processing state and the idle state, and the powers in different states of each machine are different. The task of the SFJSPCDR is (1) to assign jobs to machines and (2) to sequence the operations on each machine. The objective is makespan, energy consumption, and maximal ergonomic risk.

3 | FORMULATING THE SFJSPCDR

3.1 | Notations

All the notations are listed in Table 1.

3.2 | Assumptions

To simplify the model, some assumptions are set as follows:

(1) Jobs are independent and the pre-emption is not allowed.
(2) All jobs and machines are available at the beginning.
(3) The processing time of each operation on different combinations of machine and worker is known.
(4) Each combination of machine and worker can process only one job at a time.
(5) Each job can only be processed by one worker on one machine at a time.
(6) Once processing starts, it cannot be interrupted.
(7) The machines and workers are independent of each other.

3.3 | Formulations of SFJSPCDR

Before the SFJSPCDR is formulated, three models are built first for the objectives.

(1) The model for calculating the makespan

The makespan is determined by the longest completing time of all the machines.
**TABLE 1** Notations

| Symbol | Description |
|--------|-------------|
| \(m\)  | Machine amount |
| \(n\)  | Job amount |
| \(n_a\) | Operation amount of job \(a\) |
| \(w\)  | Worker amount |
| \(a\)  | Job index, \(a = 1, 2, \ldots, n\) |
| \(b\)  | Operation index, \(b = 1, 2, \ldots, n_a\) |
| \(c\)  | Machine index, \(c = 1, 2, \ldots, m\) |
| \(d\)  | Worker index, \(d = 1, 2, \ldots, w\) |
| \(O_{ab}\) | \(b\)-th operation of job \(a\) |
| \(M_{ab}\) | Available machines for operations, \(O_{ab}\) |
| \(W_{abc}\) | Available workers for machine \(c\), \(c \in M_{ab}\) |
| \(T_{abcd}\) | Time for worker \(d\) to operate machine \(c\) for processing operation \(O_{ab}\) |
| \(T_{\text{work}}(c)\) | Processing time of machine \(c\) |
| \(T_{\text{idle}}(c)\) | Idle time of machine \(c\) |
| \(R_{abcd}\) | The risk value of the worker \(d\) operating the machine \(c\) processing operation \(O_{ab}\) |
| \(ST_{ab}\) | The starting time of operation \(O_{ab}\) |
| \(FT_{ab}\) | The completing time of operation \(O_{ab}\) |
| \(F_c\) | The completing time of machine \(c\) |
| \(E_c\) | Total energy consumption of machine \(c\) |
| \(E^{(c)}\) | Processing energy consumption of machine \(c\) |
| \(E_{\text{work}}(c)\) | Idle energy consumption of machine \(c\) |
| \(P^{(c)}\) | Processing power of machine \(c\) |
| \(P_{\text{work}}^{(c)}\) | Idle power of machine \(c\) |
| \(C_{\text{max}}\) | Makespan |
| \(E\) | Total energy consumption of all machines |
| \(R_d\) | Ergonomic risks of worker \(d\) |
| \(R_{\text{max}}\) | The greatest ergonomic risk of all workers |
| \(X_{abcd}\) | \( \begin{cases} 1, & \text{operation } O_{ab} \text{ is processed by worker } d \text{ in machine } c \\ 0, & \text{otherwise} \end{cases} \) |

\[ C_{\text{max}} = \max(F_c) \ c = 1, 2, \ldots, m \] (1)

(2) The model for calculating the energy consumption

The total energy consumption on all machines is calculated with Equation (2), where the total energy consumption \(E_c\) of machine \(c\) is composed of the processing energy consumption and the idle energy consumption, as defined in Equation (3). The processing energy consumption \(E^{(c)}_{\text{work}}\) and the idle energy consumption \(E^{(c)}_{\text{idle}}\) on machine \(c\) are calculated with Equation (4) and Equation (5), respectively. The processing time \(T^{(c)}_{\text{work}}\) and the idle time \(T^{(c)}_{\text{idle}}\) of machine \(c\) are calculated with Equation (6) and Equation (7), respectively. The completing time of machine \(c\) is calculated by Equation (8).

\[ E = \sum_{c=1}^{m} E_c \ c = 1, 2, \ldots, m \] (2)
\[ E_c = E^{(c)}_{\text{work}} + E^{(c)}_{\text{idle}} \ c = 1, 2, \ldots, m \] (3)
\[ E^{(c)}_{\text{work}} = P^{(c)}_{\text{work}}T^{(c)}_{\text{work}} \] (4)
The model of the sustainable flexible job shop scheduling problem considering dual resource can be formulated. According to Equation (9), the ergonomic risk of worker points. According to Equation (9), the ergonomic risk of worker represents the manual material handling index; the total risk value indicates the range of ergonomics risk and the maximum ergonomic risk $R_{max}$. The makespan $C_{max}$ is calculated with Equation (11). Equation (13) indicates that the next operation of one job can only be operated by one machine at a time to process one operation. Equation (15) indicates that each operation can only be operated by one combination of one machine and one worker. Equation (16) indicates the range of $X_{abcd}$.

$$\sum_{a=1}^{m} \sum_{b=1}^{w} X_{abcd} = 1, a = 1, 2, \ldots, n, b = 1, 2, \ldots, n_a, c = 1, 2, \ldots, m d = 1, 2, \ldots, w$$

(3) The mode for calculating the ergonomic risk

The ergonomic risk for a worker varies when the workers operates different machines. According to [21, 22], the EAWS (European Assembly Worksheet) method can be employed to evaluate the ergonomic risk for workers. EAWS is generated by car manufacturers such as Volkswagen and Fiat, and is an effective method to evaluate the ergonomic risk. The EAWS method uses three risk levels to evaluate risk:

- The low-risk level: risk value $\in [0-25]$;
- The feasible level: risk value $\in [26-50]$;
- The high-risk level: risk value $>50$.

The risk points (RP) of the whole body are composed of four parts:

$$RP = PI + MMHI + FI + EP$$

where $PI$ represents the sum of the pose index; $MMHI$ represents the manual material handling index; $FI$ represents the action strength index; $EP$ represents other unconsidered risk points. According to Equation (9), the ergonomic risk of worker $c$ when operating different machine can be obtained, the total risk value $R_d$ of each worker is calculated by Equation (10), and the maximum ergonomic risk of all workers is calculated by Equation (11).

$$R_d = \sum_{c=1}^{m} \sum_{b=1}^{w} X_{abcd} R_{abcd}$$

$$R_{max} = \max(R_d) \quad d = 1, 2, \ldots, w$$

After the models for calculating all objectives are built, the model of the sustainable flexible job shop scheduling problem considering dual resource can be formulated.

$$f = \min(C_{max}, E, R_{max})$$

subject to

$$ST_{ab} - ST_{a(b-1)} \geq \sum_{d=1}^{w} \sum_{c=1}^{m} X_{abcd} T_{a(b-1)cd}$$

where Equation (12) indicates the optimisation objectives, the makespan $C_{max}$, the energy consumption $E$, and the maximum ergonomic risk $R_{max}$. The makespan is calculated with Equation (1), and the total energy consumption $E$ of all machines is calculated with Equation (2). The maximum ergonomic risk of all workers $R_{max}$ is calculated using Equation (11). Equation (13) indicates that the next operation of one job can start only after the immediately precedent operation has been finished. Equation (14) indicates that a worker can only operate one machine at a time to process one operation. Equation (15) indicates that each operation can only be operated by one combination of one machine and one worker. Equation (16) indicates the range of $X_{abcd}$.

4 | SURVIVAL DURATION-GUIDED NSGA-III ALGORITHM

The NSGA-III algorithm has been proved to have a good performance for solving multi-objective problems [23]. It not only inherits the framework and advantages of NSGA-II, but also integrates a selection strategy based on reference points, which guarantees the convergence and diversity of the next generation of individuals. It has been proved to be one of the best algorithms currently applied to multi-objective optimisation problems [23].

Among the three objectives, the makespan is contrary to the energy consumption, while the maximal risk of workers has no obvious relationship with the other two objectives. To balance the three objectives and overcome the shortcomings of the NSGA-III algorithm, we propose an improved survival duration-guided NSGA-III algorithm (SDG-NSGA-III).

The survival duration refers to the survival time of an individual during evolving. In the first generation, the survival duration of all individuals is initialised to be 0. After each iteration, compare each individual with any of its parents in the previous iteration, if they are the same, the survival duration of the individual is increased by 1. As shown in Figure 1, the survival duration of all individuals is 0 in the first generation, in the second generation, the survival duration of individual5, individual7, individual19, ..., individual189, individual200 is increased by 1, while the survival durations of individual201, individual202, ..., individual300 are all still 0. In the third generation, the
survival duration of individual5, individual9, and individual20 is increased by 1 again, and the survival duration of individual230 and individual289 is increased by 1.

The flowchart of the SDG-NSGA-III is shown in Figure 2. The detailed steps of this algorithm are as follows.

**Step 1:** Generate an initial population and initialise the survival time of all individuals. The initial population is generated randomly and the survival duration of all initial individuals is set to be zero.

**Step 2:** Calculate the average survival duration of all individuals and the survival duration of the best individual on each objective (please refer to Section 4.1.2 for details).

**Step 3:** If the average duration is smaller than $t_1$, generate the offspring population with crossover1 and mutation operator; generate offspring population by crossover2 and mutation, otherwise.

**Step 4:** Combine the parent population and the offspring population, perform the decoding algorithm and evaluate the population.

**Step 5:** Sort the population with non-dominated ranking strategy and select the offspring population with reference point-based selection.

**Step 6:** If the average survival duration is greater than $t_2$, the algorithm ends; return to step2, otherwise.

### 4.1 | Details of the SDG-NSGA-III

#### 4.1.1 | Encoding

The operation-based encoding method is adopted, that is, the $j$-th occurrence of job $i$ is the $j$-th operation of job $i$. As shown in Figure 3, the chromosome and the corresponding operations are given. The number in the last allele of the chromosome indicates which decoding method to choose. In order to balance the three objectives, two decoding methods are proposed. One is the low-energy-risk decoding method and the other is the low-risk-energy decoding method. The detailed steps for the two decoding methods are described in Section 4.1.4.

#### 4.1.2 | Calculate the survival duration of all individuals and the best individual of each objective

(1) The survival duration of all individuals

As mentioned in Section 4, each individual has survival duration, and the average survival duration of all individuals is calculated according to Equation (17).

$$average_i = \frac{\sum_{j=1}^{\text{popsize}} D_{ij}}{\text{popsize}} \quad i = 1, 2, \ldots, \text{popsize}$$

where $i$ is the generation index, $average_i$ is the average survival duration of all individuals in the $i$-th generation, $j$ is the individual index, $D_{ij}$ is the survival duration of the $j$-th individual.
generate an offspring population with only one crossover method. Therefore, two crossover methods, that is, crossover 1 and crossover 2, are employed in this study to explore and exploit the solution space.

When the average duration is smaller than a threshold value $t_1$, crossover 1 is chosen to perform the crossover operation. During the early iterations, all the individuals are almost new, there is no obvious dominant relationship between individuals, so two parent individuals are randomly selected from the population. When the average duration is greater than the threshold value $t_1$, it shows that some individuals have survived long enough and have obviously dominated other individuals in the population, the offspring population should learn from these individuals. Hence, in crossover 2, there are three parent individuals, one is selected from the Pareto individuals of the population, one is selected from the population randomly, and one is selected from the best individuals of each objective. The offspring individual learns from both the Pareto individuals and the best individual of each objective to obtain a better individual.

(1) Crossover 1

Two individuals in the population are randomly selected as parent $f_1, f_2$. Not more than half of all jobs are randomly selected as cross-jobs, and the other jobs are non-cross-jobs. The steps are as follows:

**Step 1**: The offspring individual $s1$ inherits the position of the cross-job in $f_2$, the offspring individual $s2$ inherits the position of the cross-job in $f_1$;

**Step 2**: The other positions of the offspring individual $s1$ are filled in order according to the non-cross-jobs of the parent individual $f_1$. The method to fill the other position of $s2$ is similar to that of $s1$.

The obtained offspring individuals $s1$ and $s2$ are shown in Figure 5.

(2) Crossover 2

There are three parents individuals $f1, f2, f3$. $f1$ is selected from the Pareto set of the population. $f2$ is selected from the population randomly. Among all individuals, there must be one or more best individuals for each objective, hence, $f_3$ is selected from the best individuals for each objective with roulette selection. If the survival duration of one individual is greater than that of others, which indicates the best individual for this objective has not been updated for iterations, the offspring population should have more possibility to learn from this individual in order to optimise this objective, so it is more likely to be selected as $f_3$. The steps are as follows:

**Step 1**: One job is randomly selected as a cross-job $t_1$, the positions of cross-job $t_1$ in $f_1$ are named as set $t_1$, these positions in set $t_1$ of $s$ are filled with cross-job $t_1$.
Step 2: The positions in set 1 of \( f_3 \) and other positions with the same genes in these positions are deleted.

Step 3: The cross-job 2 is selected from the remaining genes of \( f_3 \), the positions of cross-job 2 in \( f_3 \) are named as set 2.

Step 4: The positions in set 2 of \( s \) are filled with cross-job 2, the other positions of \( s \) are filled in order according to the non-cross-jobs of the \( f_2 \).

As shown in Figure 6 the selected cross-job 1 is 1, set 1 is \{1, 2, 4\}, position_1, position_2, position_4 of \( s \) is filled with the cross-job 1. According to step 2, position_1, position_2, position_4 of \( f_3 \) are deleted. The genes in position_1, position_2, position_4 of \( f_3 \) are \{3, 1, 1\}, so the other positions are deleted because the genes in these positions are the same as 3 or 1. According to step 3, the selected cross-job 2 is 2, set 2 is \{3, 7, 9\}.

The population often converges prematurely in the classical NSGA-III, therefore, in the early stage of the algorithm, crossover 1 is adopted to explore more space. When the average survival duration is greater than \( t_1 \), which means it is time to exploit the solution space, crossover 2 is adopted to exploit all individuals.

4.1.4 | Mutation

Mutation can exploit the neighbourhood, and searching the neighbourhood of the original individual may identify a more satisfied individual. The two-point swap mutation operator is employed. The steps are as follows. An example of how the mutation works is shown in Figure 7.

**Step 1:** Select two positions randomly.

**Step 2:** Swap the genes in the two positions.

4.1.5 | Decoding

Because the encoding method does not assign a machine and a worker to each operation, it is necessary to design a decoding algorithm to select an available machine and worker for each operation. Under the basic principle of the gap extrusion method [24], a double-low decoding algorithm with low energy consumption and low risk was designed. The double-low decoding comprehensively considers the makespan, the energy consumption, and the ergonomic risk. Its core idea is to select machines with low energy consumption and workers with low risk as much as possible without affecting the makespan in order to achieve the sustainable development.

The double-low decoding algorithm is composed of two decoding methods: a low-energy-risk decoding algorithm, and a low-risk-energy decoding algorithm. The difference between the two decoding methods lies in whether to choose a machine with lower energy consumption (low-energy) firstly and then choose a worker with lower risk (low-risk), or choose a worker with low-risk firstly and then choose a machine with low-energy on the premise of ensuring the makespan. Economic indicators are critical for all manufacturing companies. Both decoding methods first consider minimising the makespan. Companies have different choices for environmental indicators and social indicators. In order to consider the environmental and social indicators more evenly, two decoding algorithms are designed. The low-energy-risk decoding algorithm gives priority to the energy consumption of machines, and the low-risk-energy decoding algorithm gives priority to the risk of workers.

(1) The low-energy-risk decoding algorithm

**Step 1:** For each gene in a chromosome, translate it into a corresponding operation.

**Step 2:** Find the completing time \( T_s \) of the previous operation (if it is the first operation, set \( T_s = 0 \)). Find out all available machines that can operate the operation \( M_{ab} \), and there is a corresponding set of workers...
$W_{d,c}$ for each machine $c$ in $M_{ab}$. Find out the earliest starting time $T_f$ under the combination of machine $c$ and worker $d$ [c, d]. If $T_f$ is greater than or equal to $T_f$, go to step 2-1; go to step 2-2 otherwise.

**Step 2-1:** The earliest starting time of the combination [c, d] is $T_f$, record the starting and completing times of the combination, and go to step 3.

**Step 2-2:** For the combination [c, d], determine whether there is a gap between $T_s$ and $T_f$ in which the current operation can be inserted. If there is a gap, record the starting and completing times of the gap and go to step 3; go to step 2-3 otherwise.

**Step 2-3:** The earliest starting time of combination [c, d] is $T_f$. Record the starting and completing times and go to step 3.

**Step 3:** Select the best worker for each machine $c$. That is, find the best worker combination $d_{best}$ with the earliest completing time among all combinations of machine $c$. If there is more than one, go to step 3-1; go to step 4 otherwise.

**Step 3-1:** Select the worker in all combinations with the smallest processing energy consumption as the best worker $d_{best}$ of machine $c$. If there is more than one, go to step 3-2; go to step 4 otherwise.

**Step 3-2:** Select the worker with the lowest ergonomic risk among all combinations. If there is not only one, randomly select a worker. Then go to step 4.

**Step 4:** Choose the best combination among all combinations. Select the combination with the earliest completing time among all combinations as the best combination $[c_{best}, d_{best}]$. If the combination is not unique, go to step 4-1; go to step 5 otherwise.

**Step 4-1:** Select the combination with the lowest energy consumption among all the combinations, if there is not only one, go to step 4-2; go to step 5 otherwise.

**Step 4-2:** Select the combination with the smallest risk of the worker before processing this operation among all combinations as the best combination, if there is more than one, randomly select a combination. Then go to step 5.

**Step 5:** Finish selecting a machine and worker for this operation. Return to step 1 until the selection of machines and workers for each gene in the chromosome is finished.

In order to explain much more clearly, we give an example. Operation $O_{22}$ is to be scheduled now. Operations $O_{31}$, $O_{21}$ and $O_{11}$ have been processed. The scheduling Gantt chart before scheduling operation $O_{22}$ is shown in Figure 8.

According to step 2, the starting and completing times of all combinations that can process operation $O_{22}$ are shown in Table 2.

According to step 3, select the best worker combination for each machine. Among them, the best worker combination of machine $M_2$ is $[M_{22}, W_{22}]$, and starting time and completing times are 3 and 6, respectively; the best worker combination of machine $M_4$ is $[M_{41}, W_{41}]$, and its corresponding starting and completing times are 3 and 6, respectively; The best worker combination for machine $M_5$ is $[M_{51}, W_{52}]$, and its corresponding starting and completing times are 3 and 7, respectively.

According to step 4, select the best combination from all of the combinations, that is, $[M_{52}, W_{52}]$, $[M_{41}, W_{41}]$, and $[M_{51}, W_{52}]$. With reference to the machine processing power in Table 3, it is concluded that the processing energy

![Diagram](image)

**FIGURE 8** Gantt chart before scheduling operation $O_{22}$

**TABLE 2** Different combinations for operation $O_{22}$

| Machine | Worker | Start time | End time |
|---------|--------|------------|----------|
| $M_2$   | $W_1$  | 3          | 6        |
| $W_1$   | 3      | 6          |
| $M_4$   | $W_1$  | 3          | 6        |
| $W_3$   | 12     | 17         |
| $M_5$   | $W_1$  | 3          | 8        |
| $W_5$   | 3      | 7          |
| $W_3$   | 12     | 18         |
consumption of $M_2$ is smallest, so $[M_2, W_2]$ is selected as the best combination.

(2) Low-risk-energy decoding algorithm

The low-risk-energy decoding algorithm is similar to the low-energy-risk decoding algorithm in most steps, except steps 3 and 4. Since the other steps are the same as those in the previous section, only steps 3 and 4 are given as follows:

Step 3: Choose the best worker combination for each machine $c$. That is, find the best worker $d_{best}$ in all combinations with the earliest completing time among all combinations of machine $c$. If there are not only one, go to step 3-1; go to step 4 otherwise.

Step 3-1: Select the worker in the combination with the lowest ergonomic risk as the best worker combination $d_{best}$ of machine $c$. If there is more than one, go to step 3-2; go to step 4 otherwise.

Step 3-2: Select the worker in the combination with the smallest processing energy consumption among all combinations. If there is more than one, randomly select a worker. Then go to step 4.

Step 4: Choose the best combination among all combinations. Select the combination with the earliest completing time among all combinations as the best combination $[c_{best}, d_{best}]$. If the combination is not unique, go to step 4-1; go to step 5 otherwise.

Step 4-1: Select the combination with the smallest risk of the worker before processing this operation among all combinations as the best combination, if there is not only one, go to step 4-2; go to step 5 otherwise.

Step 4-2: Select the combination with the lowest energy consumption among all the combinations, if there is not only one, randomly select a combination. Then go to step 5.

### Table 3

| Machine | Processing power |
|---------|-----------------|
| $M_2$   | 1820            |
| $M_4$   | 2340            |
| $M_5$   | 2220            |

### Table 4

| Machine No. | Processing power | Idle power |
|-------------|------------------|------------|
| 1           | 2270             | 370        |
| 2           | 1820             | 350        |
| 3           | 1880             | 350        |
| 4           | 2340             | 390        |
| 5           | 2220             | 380        |
| 6           | 2260             | 370        |
| 7           | 2160             | 400        |
| 8           | 1600             | 350        |
| 9           | 2510             | 400        |
| 10          | 2510             | 380        |
| 11          | 2440             | 390        |
| 12          | 1920             | 320        |
| 13          | 2290             | 390        |
| 14          | 1960             | 400        |
| 15          | 2970             | 400        |

### Table 5

| Parameters | Values |
|------------|--------|
| $p_c$      | 0.8    |
| $p_m$      | 0.1    |
| $p_r$      | 0.5    |
| $t_1$      | 200    |
| $t_2$      | 2000   |
| Popsize    | 100    |

4.1.6 Non-dominated ranking and reference point-based selection strategy

With the decoding algorithm, three objectives of each individual can be calculated, and all individuals can be sorted with the non-dominated ranking to determine the individual dominance level. All individuals are selected according to the reference points.

4.1.7 Restart operator

During evolving, some individuals may be similar to or even the same as others, leading to a premature population. Therefore, a restart operator is designed. If the amount of the same individuals exceeds the restart threshold $P_r$, some new individuals are generated randomly to replace those same individuals.
5 | CASE STUDY

5.1 | Design of experiments

5.1.1 | Experiment setting

The proposed algorithm is coded with Matlab2017a and all the experiments were carried out on a desktop computer with Intel Core i5-3240, 3.40 GHz CPU, 4.00 G RAM, and Win7 64 OS.

5.1.2 | Data

Take the instances in [25] as the experiment data. The processing time in [25] is taken as that of the different combinations of machines and workers. The processing power and the idle power of machines are shown in Table 4. Due to the lack of data, the ergonomic risk value of workers is calculated by summing $PI$, $MMHI$, $FI$, and $EP$, which are all assumed to follow a uniform distribution of $U(0, 2.5)$, respectively. For convenience, they are named as SFJSPCDR01-10 in order.

5.1.3 | Parameters setting

Parameters are set in Table 5.

5.1.4 | The aims

The aims of the experiments are as follows:

(1) Analyse the effectiveness of the proposed decoding algorithm.
(2) Verify the performance of the improved survival duration-guided NSGA-III algorithm.

5.2 | Analyse the effectiveness of the proposed double-low decoding algorithm

In order to explore the effectiveness of the proposed double-low decoding algorithm, the time-based decoding algorithm, the energy-based decoding algorithm, and the risk-based decoding algorithm are compared. They are all used to solve instance SFJSPCDR01. The comparisons in terms of the final Pareto solutions of the three decoding and the double-low decoding algorithms are shown in Figures 9–11, respectively. The comparison of the hypervolume of the final Pareto solution obtained by the three decoding algorithms and the double-low decoding algorithm is shown in Table 6.

It can be seen from Figures 9–11 and Table 6, in which the bold values mean that the best value in that run, that the solutions obtained by the three decoding algorithms are inferior to the double-low decoding algorithm, which proves the effectiveness of the double-low decoding algorithm. Since the
three decoding algorithms based on time, energy, or risk only consider the better solution in a single objective and ignore others, these three decoding algorithms cannot find a better solution, and the double-low decoding proposed considers three indicators at the same time and so can obtain a better solution.

5.3 | Performance analysis of improved survival duration-guided NSGA-III

In order to verify the effectiveness of the improved survival duration-guided NSGA-III algorithm (SDG-NSGA-III), the original NSGA-III algorithm and NSGA-II algorithm are employed for comparison. These algorithms are used to solve instance SFJSPCDR01-10, and each instance is run 10 times. The hypervolumes of the solutions obtained with the three algorithms are compared, as shown in Table 7.

As shown in Table 7, the SDG-NSGA-III algorithm outperforms the original NSGA-III and NSGA-II in solving all instances except instance SFJSPCDR 03. The results prove the effectiveness of the proposed algorithm. The reason is that SDG-NSGA-III algorithm employs the framework of NSGA-III, retains its advantages, and guides the crossover by survival duration, which can avoid converging prematurely and overcome the shortcomings of the original NSGA-III.

6 | CONCLUSIONS

With the increasing concerns about sustainability, manufacturing enterprises are paying increasing attention to sustainable development. The sustainable flexible job shop scheduling problem has been studied considering dual resources. First, the makespan, energy consumption, and the ergonomic risk are considered as objectives to build an optimisation model of this problem. Then, an improved survival duration-guided NSGA-III algorithm is proposed. In order to optimise three objectives jointly, a double-low decoding algorithm is designed to solve the problem.

However, there remain some limitations to this study. In future research, the model of the social indicators should consider more practical factors in formulating the optimisation model. Moreover, more effective algorithms should be developed to solve many-objective scheduling problems.

ACKNOWLEDGEMENTS

Supported in part by the National Science Foundation of China under Grant (51,305,024) and in part by the National Key R & D plan under Grant (No. 2020YFB1712902)

REFERENCES

1. Jeurissen, R., John, E., Cannibals, W.: The triple bottom line of 21st century business. J. Bus Ethics. 23(2), 229–231 (2000)
2. Egle, S., Zivile, S.: Social sustainability measurement framework: the case of employee perspective in a CSR-committed organization. J. Clean Prod. 188, 708–719 (2018)
3. Gong, G.L., et al.: A new double flexible job-shop scheduling problem integrating processing time, green production, and human factor indicators. J. Clean Prod. 174, 560–576 (2018)
4. Neumann, W.P., Dul, J.: Human factors: spanning the gap between OM and HRM. Int. J. Oper. Prod. Manag. 30(9), 923–950 (2010)
5. Ryan, B., et al.: Integrating human factors and operational research in a multidisciplinary investigation of road maintenance. Ergonomics. 54(5), 436–452 (2011)
6. Neumann, W.P., Village, J.: Ergonomics action research II: a framework for integrating HF into work system design. Ergonomics. 55(10), 1140–1156 (2012)
7. Neumann, W.P., Medbo, P.: Integrating human factors into discrete event simulations of parallel flow strategies. Prod. Plann. Contr. 20(1), 3–16 (2009)
8. Gao, K., et al.: A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. IEEE/CAA J. Autom. Sin. 6(04), 904–916 (2019)
9. Gao, K.Z., et al.: Flexible Job-Shop Rescheduling for New Job Insertion by Using Discrete Jaya Algorithm[J]. IEEE Transactions on Cybernetics. 49(5), 1944–1955 (2019)
10. Lei, D.M., Guo, X.P.: Variable neighbourhood search for dual-resource constrained flexible job shop scheduling. Int. J. Prod. Res. 52(9), 2519–2529 (2014)
11. Lei, D.M., Tan, X.F.: Local search with controlled deterioration for multi-objective scheduling in dual-resource constrained flexible job shop[C]. In: Proceedings of the 28th Chinese Control and Decision Conference. IEEE Press, Hefei (2016)
12. Li, J.Y., Huang, Y., Niu, X.W.: A branch population genetic algorithm for dual-resource constrained job shop scheduling problem. Comput. Ind. Eng. 102, 113–131 (2016)
13. Gao, L., Pan, Q.K.: A shuffled multi-swarm micro-migrating birds optimiser for a multi-resource-constrained flexible job shop scheduling problem. Inf. Sci. 372, 655–676 (2016)
14. Zheng, X.L., Wang, L.: A knowledge-guided fruit fly optimization algorithm for dual resource constrained flexible job-shop scheduling problem. Int. J. Prod. Res. 54(18), 5554–5566 (2016)
15. Yazdani, M., Zandieh, M., Tavakkoli-Moghaddam, R.: Evolutionary algorithms for multi-objective dual-resource constrained flexible job-shop scheduling problem. Opsearch. 56, 983–1006 (2019)
16. Wu, X.L., Xiao, X., Zhao, N.: Dual resource scheduling problem of flexible job shop considering loading and unloading. Control. Decis. 35(10), 2475–2485 (2020)
17. Muhammad, A., Takashi, I.: Scheduling for sustainable manufacturing: a review. J. Clean Prod. 205, 866–883 (2018)
18. Coca, G., et al.: Sustainable evaluation of environmental and occupational risks scheduling flexible job shop manufacturing systems. J. Clean Prod. 209, 146–168 (2019)
19. Gong, X., et al.: Energy- and labour-aware flexible job shop scheduling under dynamic electricity pricing: a many-objective optimization investigation. J. Clean Prod. 209, 1078–1094 (2018)
20. 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)
21. Tang, J., Gong, B.G., Fei, Z.M.: Research on job rotation of production line based on ergonomic risk measurement. J. Anhui. Univ. Technol. (Nat. Sci.) 31(3), 32–35 (2016)
22. Schaub, K., et al.: The European assembly worksheet. Theor. Issues Ergon. Sci. 14(6), 616–639 (2013)
23. Gu, Z.M., Wang, G.G.: Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization. Future Generat. Comput. Syst. 107, 49–69 (2020)
24. Wu, X.L., et al.: Research on multi-objective flexible job shop scheduling optimization. Comput. Integr. Manuf. Syst. 12(5), 731–736 (2006)
25. Gong, X.R.: Research on Intelligent algorithms for dual flexible multi-objective job shop scheduling. Hunan University, Changsha (2018)

How to cite this article: Hongyu L., Xiuli W. An survival duration-guided NSGA-III for sustainable flexible job shop scheduling problem considering dual resources. IET Collabor. Intell. Manuf. 2021;3:119–130. https://doi.org/10.1049/cim2.12003