Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 3816574, 14 pages
https://doi.org/10.1155/2023/3816574

Research Article

Two-Stage Hybrid Optimization Algorithm for Silicon Single Crystal Batch Scheduling Problem under Fuzzy Processing Time

Lu Kang,[1,2,3] **Ding Liu**,[1,2,3] **Yali Wu**,[1,2,3] **and Guozheng Ping**[1,2,3]

[1]*School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China*
[2]*National and Local Joint Engineering Research Center of Crystal Growth Equipment and System Integration, Xi'an University of Technology, Xi'an 710048, China*
[3]*Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an 710048, China*

Correspondence should be addressed to Ding Liu; liud@xaut.edu.cn

Received 23 September 2022; Revised 12 December 2022; Accepted 16 December 2022; Published 17 January 2023

Academic Editor: Erik Cuevas

Considering the widely existing processing time uncertainty in the real-world production process, this paper constructs a fuzzy mathematical model for the silicon single crystal production batch scheduling problem to minimize the maximum completion time. In this paper, a two-stage hybrid optimization algorithm (TSHOA) is proposed for solving the scheduling model. Firstly, the improved differential evolution algorithm (IDE) is used to solve the order quantity allocation problem of silicon single crystal with different sizes to obtain the quantity of silicon single crystal rods with different sizes produced by different types of single crystal furnaces. Secondly, the variable neighborhood search (VNS) algorithm is adopted to optimize the order quantity sequencingof batch production processes. Finally, simulations and comparisons demonstrate the feasibility of the model and the effectiveness of TSHOA.

1. Introduction

As an important basic semiconductor material, silicon single crystal is widely used in semiconductor integrated circuit chips and other fields. With the rapid development of the information industry, several types of electronic equipment are increasing, and the replacement is rapid, resulting in a sharp increase in the number and types of semiconductor chips. China's industrial demand for silicon single crystals accounts for about one-third of the global total demand, but its production capacity lags far behind that of developed countries and mainly depends on import supply. To break the import restriction, get rid of the industrial monopoly and promote the development of silicon single crystal production in the direction of mass production and industrialization; China has carried out a series of scientific and technological reform policies [1]. In view of the demand for large-scale and multiple variety orders, how to improve production efficiency and reduce production costs is the core for semiconductor manufacturing enterprises to improve their competitiveness and the key to further realize scale and industrialization.

The Czochralski method is the main technical method for growing silicon single crystals. Its manufacturing process for growing silicon single crystals is complex and changeable, with many complex growth processes and long production cycles. It is recognized as one of the most complex manufacturing systems [2]. To break the bottleneck of foreign monopoly technology and have its own core technology, many domestic scholars are committed to improving the growth quality of silicon single crystals [3–5]. In recent years, many enterprises have conquered large-scale silicon single crystals and entered the preliminary mass production stage. Foreign silicon single crystal production enterprises developed earlier than domestic silicon single crystal enterprises. To carry out large-scale production and improve production efficiency, foreign silicon single crystal production enterprises have begun to study and practice the issue of silicon single crystal order allocation, shorten the production cycle, improve production efficiency, and reduce

production costs through reasonable production scheduling [6, 7]. However, domestic silicon single crystal enterprises still use manual scheduling methods for production planning. Therefore, it has become a major research hotspot to study advanced and practical algorithms to solve the production scheduling problem for domestic production enterprises.

However, the influence of uncertain processing time will cause a huge social and economic impact on enterprise production, which has attracted more and more researchers' attention [8–11]. For example, Sakaw and Mori [8] first proposed a triangular fuzzy number ranking criterion to compare the size of fuzzy numbers and applied this criterion to solve the fuzzy job shop scheduling problem. Engin and Yilmaz [9] used a fuzzy processing time and a fuzzy due date to formulate the multiobjective hybrid flow-shop scheduling with multiprocessor tasks problem. Alharbi and El-Wahed Khalifa [10] adopted pentagonal fuzzy numbers to represent the processing time to solve the flow-shop scheduling problem under fuzzy environment. Chen [11] used triangular fuzzy number to represent the uncertainty of job processing time to minimize the maximum fuzzy completion time for the fuzzy distributed flexible job shop scheduling problem. The silicon single crystal production batch scheduling problem is a constrained combinatorial optimization problem, which belongs to NP-hard problem; it is difficult to solve it by traditional optimization methods. In recent years, most researchers use intelligent optimization algorithm to solve the NP-hard problem and have achieved good results. For example, Lei et al. [12] improved the coding method and proposed a hybrid frog leaping algorithm (SFLA) to solve the green flexible job shop scheduling problem with the objective function of minimizing workload and total energy consumption. Lei and Yang [13] took the energy consumption constraint as the objective function in the first stage, and then sent the solution set obtained in the first stage to the second stage. In the second stage, a certain energy consumption value was taken as the constraint for fine solution to obtain a scheduling scheme meeting the energy consumption constraint. Guo and Zhong [14] put forward many requirements on the scheduling model of remanufacturing jobs, established a four objective scheduling model, and introduced multiple population coevolution based on artificial fish swarm algorithm to solve the model. Li et al. [15] proposed a new improved artificial immune algorithm (IAIS) for the second type of fuzzy flexible job shop scheduling problem under the fuzzy processing time. Rui and Gong [16] adopted the improved decomposition based multiobjective evolutionary algorithm (IMOEA/D) and designed a variable neighborhood search combining five local search strategies to optimize the maximum fuzzy completion time. There is no related literature about the existing fuzzy scheduling research on the batch scheduling problem of silicon single crystal production.

To sum up, the production scheduling problem generally focuses on two aspects, that is, how to establish an appropriate model to describe the actual scheduling and how to solve the model to find the best feasible solution that meets the objective. In view of the solution ideas and methods of above scheduling problems, this paper establishes a minimum fuzzy maximum completion time optimization model for silicon single crystal production batch scheduling problem and proposes an intelligent optimization algorithm to solve the model.

2. Silicon Single Crystal Batch Scheduling Problem with Multiple Size under Fuzzy Processing Time

2.1. The Model of the Silicon Single Crystal Batch Scheduling Problem. This paper studies the optimal scheduling according to the production equipment matching situation and production task demand of a silicon single crystal production plant. Among them, the corresponding thermal field dimensions of the single crystal furnace equipped by the plant are 22-inch thermal field, 24-inch thermal field, 28-inch thermal field, and 32-inch thermal field.

The types of silicon single crystal required for the order task are 6-inch, 8-inch, 10-inch, and 12-inch. According to the production conditions, the types of silicon single crystal that can be grown by different types of single crystal furnaces are shown in Table 1. Where "Yes" means that this type of silicon single crystal can be grown by this type of single crystal furnace and "No" means that this type of silicon single crystal cannot be grown by this type of single crystal furnace.

Because the production process is often accompanied by many uncertainties, the processing time of each silicon single crystal can only be determined as a rough range. Therefore, this article uses triangular fuzzy number (TFN) to represent the processing time of each process with $\mathrm{TFN} = (t_1,\ t_2,\ t_3)$. The membership function graph is shown in Figure 1. The abscissa represents the processing time, and the ordinate represents the membership value. The closer the membership value is to 1, the higher the degree of belonging to the fuzzy set is. Among them, t_1 closest to the origin represents the most optimistic processing time, t_2 that in the middle represents the most probable processing time, and t_3 farthest from the origin represents the most conservative processing time. The membership expression of TFN is shown in formula (1) [17] and that will have several special cases as shown in formula (2)–(4).

$$\mu_{TFN}(x) = \begin{cases} 0, & x \le t_1, \\ \dfrac{x - t_1}{t_2 - t_1}, & t_1 < x \le t_2, \\ \dfrac{t_3 - x}{t_3 - t_2}, & t_2 < x < t_3, \\ 0, & x \ge t_3. \end{cases} \tag{1}$$

(1) If $t_2 = t_3$,

Table 1: Corresponding production type representation.

The type of silicon single crystal	The type of single crystal furnace			
	22 inch	24 inch	28 inch	32 inch
6 inch	Yes	Yes	No	No
8 inch	Yes	Yes	Yes	No
10 inch	Yes	Yes	Yes	No
12 inch	No	No	Yes	Yes

$$\mu_{TEN}(x) = \begin{cases} 0, & x \le t_1, \\ \dfrac{x - t_1}{t_2 - t_1}, & t_1 < x \le t_2 = t_3, \\ 0, & x > t_2 = t. \end{cases} \tag{2}$$

(2) If $t_1 = t_2$,

$$\mu_{TEN}(x) = \begin{cases} 0, & x < t_1 = t_2, \\ \dfrac{t_3 - x}{t_3 - t_2}, & t_1 = t_2 \le x < t_3, \\ 0, & x \ge t_3. \end{cases} \tag{3}$$

(3) If $t_1 = t_2 = t_3$,

$$\mu_{TEN}(x) = \begin{cases} 1, & x = t_1 = t_2 = t_3, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Because the processing time is expressed in TFN, the fuzzy completion time C_i is also TFN, expressed as $C_i = (c_1, c_2, c_3)$; c_1, c_2, and c_3 represent the most optimistic completion time (i.e., the lower bound of C_i), the most likely completion time, and the most conservative completion time (i.e., the upper bound of C_i) of the i-th size order job, respectively.

The model of the scheduling problem in this paper is described as follows. The order quantity of L-type silicon wafers with different sizes is converted into N jobs and allocated to M single crystal furnaces with different types. Each job has multiple processes, and each process can only have one optional single crystal furnace. The processing sequence of the processes is fixed, and the performance of each single crystal furnace equipment is different. The purpose of scheduling is to determine the jobs allocated on different types of single crystal furnace equipment, so as to minimize the maximum fuzzy completion time of the whole production system. The following constraints must be met [18]:

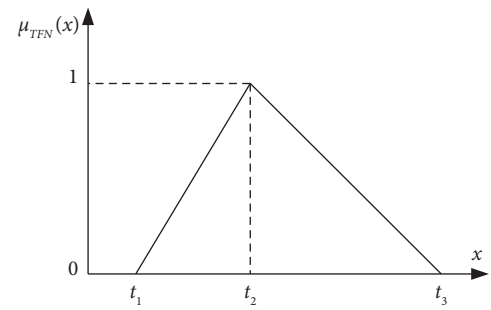

Figure 1: The membership function of triangular fuzzy number.

(1) The same job can only be processed on one equipment at any time

(2) Only one job can be processed by the same equipment at any time

(3) The single crystal furnace cannot be interrupted during the production process, but can wait at a specific process

(4) The same job can only be processed in the next operation after the previous operation is completed

(5) Assuming that the failure factor is not considered, all single crystal furnaces are able to work normally

The symbols and definitions involved in this study are shown in Table 2.

According to the symbol definition, the objective function in this paper can be expressed as follows:

Objective function:

$$f = \min\left(\widetilde{T_{\text{END}}}\right) = \max\left(C_1, C_2, \cdots, C_i\right) i \in [1, L]. \tag{5}$$

Subject to:

$$C_i = \max\left(\widetilde{et}_{i,j,z}\right), i \in [1, L], j \in \left[1, M_s\right], z \in [1, Z], s \in [1, 4], \tag{6}$$

$$\widetilde{et}_{i,j,z} = \widetilde{st}_{i,j,z} + \widetilde{t_{i,j}}, i \in [1, L], j \in \left[1, M_s\right], z \in [1, Z], s \in [1, 4], \tag{7}$$

$$\sum_{j=1}^{M_s} o_{k,j} = 1, k \in [1, N], s \in [1, 4], \tag{8}$$

$$\sum_{k=1}^{N} o_{k,j} = 1, j \in \left[1, M_s\right], s \in [1, 4], \tag{9}$$

Table 2: Symbol definition.

Symbol	Expression
M	Number of machines
N	Number of jobs
L	Dimension type
Z	Maximum of production batch
i	Dimension sequence
j	Machine sequence
k	Job sequence
z	Production batch
s	Machine type
Y_j	Loading capacity of the j-th machine
N_i	The order quantity of the i-th size order silicon single crystal
D_i	Actual production quantity of the i-th size order silicon single crystal
$o_{k,j}$	The k-th job is operated on the j-th machine
$U_{i,j}$	The number of the i-th size silicon single crystal grown on the j-th machine
$\widetilde{t_{i,j}}$	Fuzzy processing time of the i-th size silicon single crystal on the j-th machine
$\widetilde{C_i}$	Fuzzy completion time of the i-th size silicon single crystal order
$\widetilde{T_{\mathrm{END}}}$	Fuzzy maximum completion time
$\widetilde{st}_{i,j,z}$	Fuzzy starting time of the i-th size silicon single crystal in z-batch on the j-th machine
$\widetilde{et}_{i,j,z}$	Fuzzy completion time of the i-th size silicon single crystal in z-batch on the j-th machine

$$\widetilde{st}_{i,j,z} = \widetilde{et}_{i,j,z-1} + 2, i \in [1, L], j \in \left[1, M_s\right], s \in [1, 4], \tag{10}$$

$$N_i \le D_i, D_i = \sum_{j=1}^{M_s} Y_j * U_{i,s}, i \in [1, L], j \in \left[1, M_s\right], s \in [1, 4], \tag{11}$$

$$\widetilde{st}_{i,j,z} \ge 0, \widetilde{t_{i,j}} \ge 0, i \in [1, L], j \in \left[1, M_s\right], z \in [1, Z], s \in [1, 4], \tag{12}$$

$$0 \le U_{i,s} \le Q_{i,s}, Q_{i,s} = \mathrm{Ceiling}\left(\frac{N_i}{Y_s}\right), i \in [1, L], s \in [1, 4], \tag{13}$$

where equation (5) represents the objective function. Equations (6) and (7) represent the calculation of fuzzy completion time. Equations (8)–(10) are equality constraints, where equation (8) indicates that each job can only be processed by one furnace, equation (9) indicates that each furnace can only process one job at the same time, and equation (10) indicates that the fuzzy starting time of the next batch is equal to the fuzzy completion time of the previous batch plus the time to replace the quartz crucible, and the replacement time is 2 h. Equations (11)–(13) are inequality constraints, where equation (11) indicates that the actual production quantity must be greater than the order required quantity. Equation (12) indicates that the starting time variables and running time variables must be greater than or equal to 0. Equation (13) represents the upper and lower limits of the number of the i-th size silicon single crystal produced on the s-type single crystal furnace, and *Ceiling* means rounding up.

2.2. Fuzzy Number and Its Operation. When calculating the objective function corresponding to different individuals or solutions of the fuzzy production scheduling problem, it is necessary to add, subtract, multiply, divide, and take the largest of the fuzzy number (the processing time). At the same time, when evaluating the advantages and disadvantages of different solutions, it is necessary to carry out comparative operations. The definition for two triangular fuzzy numbers, TEN$\widetilde{s} = (s_1, s_2, s_3)$ and $\widetilde{t} = (t_1, t_2, t_3)$, is as follows:

(1) Addition, subtraction, and multiplication operations:

$$\begin{aligned} \widetilde{s} + \widetilde{t} &= \left(s_1 + t_1, s_2 + t_2, s_3 + t_3\right), \\ \widetilde{s} - \widetilde{t} &= \left(s_1 - t_1, s_2 - t_2, s_3 - t_3\right), \\ \lambda * \widetilde{s} &= \left(\lambda s_1, \lambda s_2, \lambda s_3\right). \end{aligned} \tag{14}$$

(2) Comparison operations: for triangular fuzzy numbers $\widetilde{s} = (s_1, s_2, s_3)$ and $\widetilde{t} = (t_1, t_2, t_3)$, the following criteria are usually used for comparison [13]:

Criteria 1:
If $L_0(\widetilde{t}) = (t_1 + 2t_2 + t_3)/4 > L_0(\widetilde{s}) = (s_1 + 2s_2 + s_3)/4$, then $\widetilde{t} > \widetilde{s}$, otherwise, $\widetilde{t} < \widetilde{s}$
Criteria 2:
If $L_0(\widetilde{t}) = L_0(\widetilde{s})$, then compare $L_1(\widetilde{t}) = t_2$ and $L_1(\widetilde{s}) = s_2$, when $L_1(\widetilde{t}) > L_1(\widetilde{s})$, then $\widetilde{t} > \widetilde{s}$
Criteria 3:

If $L_0(\tilde{t}) = L_0(\tilde{s})$ and $L_1(\tilde{t}) = L_1(\tilde{s})$, then compare $L_2(\tilde{t}) = t_3 - t_1$ and $L_2(\tilde{s}) = s_3 - s_1$; when $L_2(\tilde{t}) > L_2(\tilde{s})$, then $\tilde{t} > \tilde{s}$

(3) Larger operations: at present, the following two methods are commonly used to calculate the larger operation of triangular fuzzy numbers, namely, the S method and the L method [8]. The definitions are as follows:

S method definition:

$$\tilde{s} \vee \tilde{t} = (s_1 \vee t_1, s_2 \vee t_2, s_3 \vee t_3). \tag{15}$$

L method definition:

$$\tilde{s} \vee \tilde{t} = \begin{cases} \tilde{s}, & \text{if } \tilde{s} > \tilde{t}, \\ \tilde{t}, & \text{otherwise.} \end{cases} \tag{16}$$

Taking $\tilde{s}$ and $\tilde{t}$ as examples, the results, respectively, obtained by S method and L method for fuzzy larger operation are shown in Figure 2.

According to the two operations above, the result obtained by L method is $\tilde{s}$ or $\tilde{t}$ and the result obtained by S method is the set combination of two triangular fuzzy numbers. Therefore, this paper adopts the L method as larger operation of triangular fuzzy numbers.

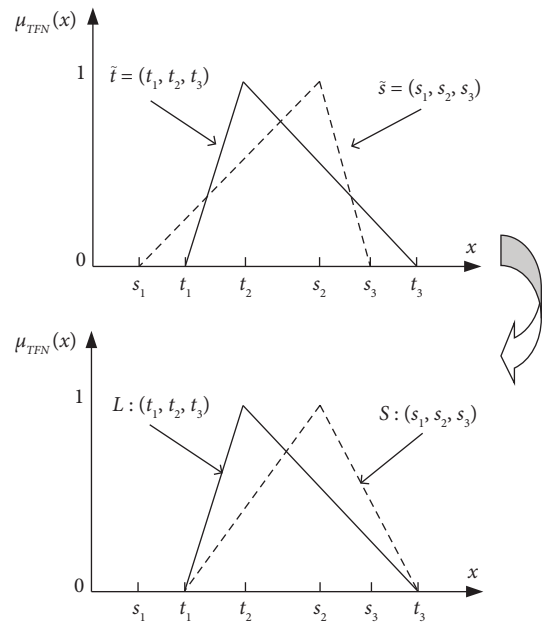

Figure 2: Larger operation of triangular fuzzy number.

3. Framework of a Two-Stage Hybrid Optimization Algorithm (TSHOA)

3.1. Encoding and Decoding. According to the problem description of silicon single crystal batch scheduling problem with multiple size under fuzzy processing time, the final scheduling result is composed of two parts. The first part is how to allocate the order numbers of silicon single crystal rod for different types of single crystal furnaces, and the other part is how to allocate the batch plans for produced order quantity on the single crystal furnaces. Therefore, algorithm coding consists of two stages.

3.1.1. The First Stage. The coding in the first stage is represented by a vector composed of the number of silicon single crystal with different sizes produced by different types of single crystal furnaces as shown in Table 1. The vector dimension is 10 dimensions, and the vector composition form is shown in the following formula:

$$x = [x_{i1}, x_{i2}, x_{i3}, x_{i4}, x_{i5}, x_{i6}, x_{i7}, x_{i8}, x_{i9}, x_{i10}]. \tag{17}$$

where x_{i1} represents the number of 6-inch silicon single crystal produced by the single crystal furnace corresponding to the 22-inch thermal field. x_{i2} represents the number of 8-inch silicon single crystal produced by the single crystal furnace corresponding to the 22-inch thermal field. x_{i3} represents the number of 10-inch silicon single crystal produced by the single crystal furnace corresponding to the 22-inch thermal field. x_{i4} represents the number of 6-inch silicon single crystal produced by the single crystal furnace corresponding to the 24-inch thermal field, and the following variables meaning can be analogized in the same way.

According to the variable solution results of the first stage, it is analyzed and judged whether it is necessary to enter the second stage. To describe the assignments more clearly, a visual representation of variable assignments is presented in Table 3. When the sum of the order quantity exceeds the total number of single crystal furnaces, it is necessary to enter the second stage to obtain the batch allocation scheme to solve the objective function. Otherwise, it means that one batch can complete the production. The maximum fuzzy processing time is taken as the fuzzy completion time of this type of single crystal furnace, and the final completion time is the maximum fuzzy completion time of all types of single crystal furnaces, that is, in Table 3, the total completion time of each column of variables to complete the production is summed, and the fuzzy completion time of the column with the maximum completion time is the maximum fuzzy completion time of the total factory. For example, if $x_{i1} + x_{i2} + x_{i3} = 10$ and $M_1 = 5$, the number of 22-inch thermal field single crystal furnaces is 5, and it is necessary to enter the second stage at this time.

3.1.2. The Second Stage. Suppose the first stage solution result is $x_{i1} = 4$, $x_{i2} = 3$, and $x_{i3} = 3$, that is, the result of the first stage solution is that 6-inch silicon single crystal rods should be produced 4, 8-inch and 10-inch silicon single crystal rods should be produced 10, respectively; all types of this silicon single crystal rods need to be produced on a 22-inch single crystal furnace. Because the processing time from the same type of single crystal furnace that processes different sizes of silicon single crystals is different, so the main purpose of the second stage is how to distribute the production process, so that the total completion time of the distribution results is the shortest. The dimension of the

Table 3: The first stage encoding variable representation.

Silicon single crystal size	Single crystal furnace type (thermal field size)			
	22 inch	24 inch	28 inch	32 inch
6 inch	x_{i1}	x_{i4}	—	—
8 inch	x_{i2}	x_{i5}	x_{i7}	—
10 inch	x_{i3}	x_{i6}	x_{i8}	—
12 inch	—	—	x_{i9}	x_{i10}

variables in the second stage is determined by the summation of the corresponding columns in the first stage. For the convenience of recording, the variables in the second stage are composed of s ($s = 4$, it represents 4 different types of single crystal furnaces) cell arrays. Suppose that, each array ceils consist of 1, 2, 3, and 4, which represent 6-inch, 8-inch, 10-inch, and 12-inch silicon single crystal rods, respectively. Therefore, each digit in the coding string of the second stage represents a job number, respectively. For the same job number, the number of occurrences represents the number of the processing of the same size silicon single crystal rod. And the second stage variable encoding representation is shown in equation (18). Take Table 3 as example, in the first stage, the first column corresponds to the column summation $x_{i1} + x_{i2} + x_{i3} = 10$, then the variable dimension is 10 dimensions. Assume that there are 5 single crystal furnaces corresponding to the 22-inch thermal field, i.e., if 10 silicon single crystal rods need to be produced, the initial variables are randomly generated as shown in formula (19).

$$p = \left[\left\{p^1\right\}, \left\{p^2\right\}, \left\{p^3\right\}, \left\{p^4\right\}\right], \tag{18}$$

$$\begin{aligned} p^1 &= \left[\, p_1 \;\; p_2 \;\; p_3 \;\; p_4 \;\; p_5 \;\; p_6 \;\; p_7 \;\; p_8 \;\; p_9 \;\; p_{10} \,\right] \\ &= \left[\, 2 \;\; 3 \;\; 2 \;\; 1 \;\; 1 \;\; 3 \;\; 2 \;\; 1 \;\; 1 \;\; 3 \,\right], \end{aligned} \tag{19}$$

where p represents the second stage variable representation. p^1 represents the cell array variable representation of the first type of single crystal furnace, and the s type of single crystal furnace models are arranged in order. It can be seen from formula (19) that the p^1 code length is 10 in which $p_1 = 2$ means that the first 8-inch silicon single crystal rod is produced on the $M_{1,1}$, $p_2 = 3$ means that the first 10-inch silicon single crystal rod is produced on the $M_{1,2}$, $p_3 = 2$ means that the second 8-inch silicon single crystal rod is produced on the $M_{1,3}$, and the following is the same, but $p_6 = 3$ does not represent the sixth single crystal furnace, but means that the second 10-inch silicon single crystal rod is produced on the first completed single crystal furnace in the first batch of 5 sets single crystal furnaces; it can be noted for the virtual sixth single crystal furnace. By analogy, the fuzzy completion time of the last single crystal furnace in the last batch can be obtained by solving the variables. Similarly, the fuzzy completion time of different types of single crystal furnaces can be obtained. The maximum fuzzy completion time in s type of single crystal furnace is obtained as the objective function through L method comparison criterion.

3.2. Improved Differential Algorithm (IDE)

3.2.1. Improved Mutation Operations. Standard differential evolution (DE) is a swarm evolution-based algorithm proposed by Storn and Price [19]. The algorithm has the characteristics of memorizing individual optimal solutions and sharing information within the population, that is, optimizing the solutions to the problem through cooperation and competition among individuals in the population. The mutation operation is the core of the DE algorithm, which affects the performance of the algorithm to a large extent. The five types of typical mutation strategies commonly used are as follows:

$$v_{i,j}^{G+1} = x_{i,j}^{G} + F \times \left(x_{r1,j}^{G} - x_{r2,j}^{G}\right), \tag{20}$$

$$v_{i,j}^{G+1} = x_{r3,j}^{G} + F \times \left(x_{r1,j}^{G} - x_{r2,j}^{G}\right), \tag{21}$$

$$v_{i,j}^{G+1} = x\text{best}_{j}^{G} + F \times \left(x_{r1,j}^{G} - x_{r2,j}^{G}\right), \tag{22}$$

$$v_{i,j}^{G+1} = x\text{best}_{j}^{G} + F \times \left(x_{r1,j}^{G} - x_{r2,j}^{G}\right) + F \times \left(x_{r3,j}^{G} - x_{r4,j}^{G}\right), \tag{23}$$

$$v_{i,j}^{G+1} = x_{r1,j}^{G} + F \times \left(x_{r2,j}^{G} - x_{r3,j}^{G}\right) + F \times \left(x_{r4,j}^{G} - x_{r5,j}^{G}\right), \tag{24}$$

where $r1$, $r2$, $r3$, $r4$, and $r5$ are integers that are different from each other, representing different individuals in the population; j denotes the dimension of individuals in the population; $x_{i,j}^{G}$ denotes the parent individual of the G-th generation; $x\text{best}_{j}^{G}$ denotes the optimal individual of the G-th generation, and F is the variation factor.

According to formula (20), in the standard DE, the mutation vector $v_{i,j}^{G+1}$, basis vector $x_{i,j}^{G}$, and difference vector $x_{r1,j}^{G} - x_{r2,j}^{G}$ are all taken from the same dimension, then for the entire population, the mutation, crossover, or selection operations between each dimension are mutually independent, that is, the evolutionary process between each dimension is unrelated. If a certain dimension or several dimensions in the population cause the algorithm to fall into a precocious state due to the high degree of aggregation of individuals, it is difficult to jump out of the local optimal solution by its own dimension alone. Therefore, this paper introduces the idea of different-dimensional mutation proposed in the literature [20], and at the same time, this paper adopts a variety of mutation strategies. The specific expression is as follows [21]:

Mutation strategy 1:

$$\begin{aligned} v_{i,j}^{G+1} &= x_{r3,j}^{G} + F \times \left(x_{r1,j}^{G} - x_{r2,j}^{G}\right) \\ 1 &\le i \le \frac{Np}{3}. \end{aligned} \tag{25}$$

Mutation strategy 2:

$$v_{i,j}^{G+1} = x\text{best}_j^G + F \times \left(x_{r1,j}^G - x_{r2,j}^G\right) \quad \frac{Np}{3} < i \le \frac{2Np}{3}. \tag{26}$$

Mutation strategy 3:

$$v_{i,j}^{G+1} = x_{r1,n}^G + F \times \left(x_{r2,m}^G - x_{r3,m}^G\right) \quad \frac{2Np}{3} < i \le Np, \tag{27}$$

where Np represents the number of individuals in the population, $i \neq r1 \neq r2 \neq r3$, and $j \neq n \neq m$. It can be seen from formulas (25)–(27) that the whole population first performs a random global search mutation strategy, such as formula (25). Then, the mutation strategy is transformed, and the global best individual is used as the base vector for local optimization, as shown in formula (26). Finally, a different-dimensional mutation strategy such as formula (27) is adopted. When the evolutionary solution of one-dimension falls into a local optimum, the particles of other dimensions can be used to jump out of the current local optimum to avoid the algorithm from entering a premature state. This multistrategy different-dimensional mutation method combines the advantages of each mutation strategy, effectively balancing the exploration and exploitation capabilities of the algorithm.

It can be seen from the mutation strategy that the mutation factor F has a greater impact on the performance of the algorithm. In the early stage of the algorithm, mutation factor F is a larger value, which improves the global search ability of the algorithm; in the later stage of the algorithm, a smaller mutation factor F can make the algorithm more refined to improve the local search ability of the algorithm. Therefore, dynamically adjusting the value of the variation factor F is crucial to the balance the exploration ability and optimization ability of the algorithm. Therefore, this article proposes a new dynamic adjustment variation factor, which determines the variation factor of the current individual according to the fitness value ranking of the individual in population. The specific expression is shown in the following formulas:

$$F(i) = F_{\min} + \frac{s(i)}{Np} * \left(F_{\max}^G - F_{\min}\right), \tag{28}$$

$$F_{\max}^G = F_{\max} - \frac{\text{Gen}}{G_{\max}} \left(F_{\max} - F_{\min}\right), \tag{29}$$

where $F_{\max}^G$ is the maximum variation factor of the G-th generation population, which is generated by formula (29) [22], $F_{\max}$ and $F_{\min}$, respectively, represent the maximum and minimum values of the variation factor, Gen and $G_{\max}$, respectively, represent the current algebra and the maximum algebra, and $s(i)$ represent the ranking of individual i in the entire population Np, which is sorted according to the individual fitness value. In this way, each individual in the population has an independent variation factor, and selects the desired search method according to its own characteristics, that is, if the variation factor is large, it means that the ranking is high in the entire population, and the corresponding individual fitness value is large, and the individual should focus on the global search, otherwise, the individual should focus on the local fine search.

3.2.2. Crossover Improvements. The crossover operation recombines the mutant individual generated by the mutation operation with the original individual according to the crossover probability CR to generate a new individual. The traditional crossover operation is obtained by synthesizing the mutant individual and the original individual, but the conventional crossover operation has a slow evolution speed and produces better results. The probability of the individual is small, so according to the idea of crossover operation between the optimal individual and the original individual in the improved particle swarm algorithm [23], the improved crossover operation is shown in the following formula [21]:

$$u_{i,j}^{G+1} = \begin{cases} v_{i,j}^{G+1}, & \text{rand}() \le CR \text{or} j = j_{\text{rand}}, \\ x\text{best}_j^G, & \text{otherwise}, \end{cases} \tag{30}$$

$$CR = CR_{\max} - \frac{\text{Gen}}{G_{\max}} \left(CR_{\max} - CR_{\min}\right), \tag{31}$$

where CR is adaptively adjusted by formula (31), $x\text{best}^G$ is the optimal individual of the G-th generation, and the new individual is synthesized from the optimal individual and the mutant individual. In the early stage of the algorithm, the mutation individual $v_{i,j}^{G+1}$ accounts for a large proportion of the newly generated crossover individuals, which is helpful for the global optimization of the algorithm. In the later stage of the algorithm, the optimal individual $x\text{best}_j^G$ accounts for a large proportion of the newly generated crossover individuals, which is helpful for the local optimization of the algorithm.

3.2.3. Selection Improvements. The individual obtained by the crossover operation and the original individual will retain the individual with good fitness value to the next generation, so as to ensure that the algorithm is constantly approaching the optimal solution. The traditional selection operation ignores the excellent characteristics of the mutant individual. This paper adopts the selection strategy idea of the traditional GA algorithm [24], the generation of new individuals is selected among the original individuals, mutant individuals, and crossover individuals. Take the minimum problem as an example, as shown in the following formula [21]:

$$x_i^{G+1} = \begin{cases} x_i^G, & f\left(x_i^G\right) == \min\left(f\left(x_i^G\right), f\left(v_i^{G+1}\right), f\left(u_i^{G+1}\right)\right), \\ v_i^{G+1}, & f\left(v_i^{G+1}\right) == \min\left(f\left(x_i^G\right), f\left(v_i^{G+1}\right), f\left(u_i^{G+1}\right)\right), \\ u_i^{G+1}, & f\left(u_i^{G+1}\right) == \min\left(f\left(x_i^G\right), f\left(v_i^{G+1}\right), f\left(u_i^{G+1}\right)\right). \end{cases} \tag{32}$$

3.3. Variable Neighborhood Search Algorithm. The variable neighborhood search (VNS) algorithm [25] is an improved local search algorithm. It is mainly composed of two parts, namely, neighborhood action and variable neighborhood descent. The neighborhood action can generate multiple different neighborhoods of the current solution, and the variable neighborhood descent can obtain the optimal value by searching alternately with the neighborhood search structure composed of different actions. The basic steps of the VNS algorithm are as follows:

Step 1: we generate the initial solution x according to the upper and lower limits of the variables and set the number of neighborhood search structures $k_{\max}$, the maximum number of searches $d_{\max}$, and the maximum number of iterations $G_{\max}$.

Step 2: we randomly select a feasible solution x' according to the k-th neighborhood structure $N_k(x)$ of the current solution x and then search for x' in the first neighborhood search structure; at this time, $k = 1$, $d = 1$, and $g = 1$.

Step 3: if the fitness value of the newly generated solution x'' is better than the current solution, update the optimal solution to x'', we continue to use the current neighborhood search structure to search, otherwise, let $d = d + 1$, if $d > d_{\max}$. It shows that the neighborhood search structure has been unable to obtain a better solution, that is, the neighborhood search structure has been obtained the local optimal solution of the neighborhood, let $k = k + 1$.

Step 4: if $k \le k_{\max}$ is not reached, we go to step 3, otherwise, we go to step 5.

Step 5: let $g = g + 1$; if $g \le G_{\max}$, we go to step 2; otherwise, we end the iteration and output the optimal solution.

This article mainly adopts the following three neighborhood structures:

Neighborhood search structure, N_1: we randomly select two processes with different process codes in the processes to exchange.

Neighborhood search structure, N_2: we randomly select a process in the processes and exchange this process with its subsequent process, if the process is the same, we reselect it backward until the process is different, if this process is the last process in the processes, then we rerandomize.

Neighborhood search structure N_3: we randomly select a process in the processes and exchange this process with its previous process; if the process is the same, we reselect it forward until the process is different; if this process is the first in the processes, then we rerandomize.

3.4. The TSHOA Algorithm. In this article, a two-stage optimization algorithm (TSHOA) is used to solve the fuzzy silicon single crystal scheduling problem under uncertain time with multiple sizes. First, the silicon wafer orders are converted into the number of silicon single crystal rods after obtaining orders with different types of single crystal furnaces and silicon wafers of different sizes, and the IDE algorithm is used to obtain the allocation of silicon single crystal rods of different sizes produced by different types of single crystal furnaces to complete the first stage of problem solving. Then, the VNS algorithm is used to solve the second stage, and the batch allocation solution are carried out for the silicon single crystal rods that need to be produced in multiple batches, so as to minimize the final fuzzy completion time.

The TSHOA proposed in this article combines the advantages of the DE algorithm and the VNS algorithm. The DE algorithm has fast convergence performance, simple parameters, and easy operation, but the algorithm is very easy to fall into local optimization. The VNS algorithm is an improved local search algorithm, which has strong local search ability and can effectively help the DE algorithm jump out of the local optimum. Therefore, the proposed algorithm has better global and local search capabilities and has high solution accuracy. But because of the variable neighborhood search feature of the VNS algorithm, when the problem scale is large, the space of the neighborhood solution will also grow, which will consume a lot of search time. Specifically, the steps are as follows:

Step 1: we set the initialization parameters of the algorithm, the population size Np, the maximum iteration termination number $G_{\max}$, the value range of the independent variable, and let the initial iteration number Gen = 1.

Step 2: we initialize the population. The first stage initialization is performed. The initial population is obtained by random initialization according to the upper and lower limits of the variables as shown in formula (17).

Step 3: we calculate and analyze whether it is necessary to enter the second stage according to the individual results. As shown in Table 3, when the sum of the corresponding columns of single crystal furnaces with different thermal fields exceeds the number of single crystal furnaces, enter the second stage to find out the allocation plan and obtain the objective function. If entering the second stage, use the neighborhood search

Table 4: Fuzzy processing time of each process.

Process and processing time	Size (Charging amount)	22 inch (120 kg)	24 inch (150 kg)	28 inch (250 kg)	32 inch (400 kg)
Charging fuzzy processing time	6	[0.6, 0.9, 1.2]	[0.7, 1, 1.2]	—	—
	8	[0.6, 0.9, 1.2]	[0.7, 1, 1.2]	[1.2, 1.5, 1.7]	—
	10	[0.6, 0.9, 1.2]	[0.7, 1, 1.2]	[1.2, 1.5, 1.7]	—
	12	—	—	[1.2, 1.5, 1.7]	[1.7, 2, 2.2]
Melting fuzzy processing time	6	[6, 7, 8]	[8, 9, 10]	—	—
	8	[6, 7, 8]	[8, 9, 10]	[14, 15, 16]	—
	10	[6, 7, 8]	[8, 9, 10]	[14, 15, 16]	—
	12	—	—	[14, 15, 16]	[19, 20, 21]
Seeding fuzzy processing time	6	[2, 3, 6]	[2, 3, 6]	—	—
	8	[2, 3, 6]	[2, 3, 6]	[2, 3, 6]	—
	10	[2, 3, 6]	[2, 3, 6]	[2, 3, 6]	—
	12	—	—	[2, 3, 6]	[2, 3, 6]
Shouldering fuzzy processing time	6	[1.1, 1.2, 1.3]	[1.1, 1.2, 1.3]	—	—
	8	[1.4, 1.5, 1.6]	[1.4, 1.5, 1.6]	[1.4, 1.5, 1.6]	—
	10	[1.7, 1.8, 1.9]	[1.7, 1.8, 1.9]	[1.7, 1.8, 1.9]	—
	12	—	—	[2.2, 2.3, 2.4]	[2.2, 2.3, 2.4]
Equal diameter growth fuzzy processing time	6	[51, 53, 55]	[64, 66, 68]	—	—
	8	[28, 30, 32]	[35, 37, 39]	[59.9, 61.9, 63.9]	—
	10	[30, 32, 34]	[38.5, 40.5, 42.5]	[65.5, 67.5, 69.5]	—
	12	—	—	[78, 80, 82]	[126, 128, 130]
Ending fuzzy processing time	6	[2.5, 3, 3.5]	[2.5, 3, 3.5]	—	—
	8	[3.5, 4, 4.5]	[3.5, 4, 4.5]	[3.5, 4, 4.5]	—
	10	[5, 6, 7]	[5, 6, 7]	[5, 6, 7]	—
	12	—	—	[9, 10, 11]	[9, 10, 11]
Cooling down the furnace fuzzy processing time	6	[5, 5.5, 6]	[5, 5.5, 6]	—	—
	8	[6, 7, 8]	[6, 7, 8]	[6, 7, 8]	—
	10	[8, 9, 10]	[8, 9, 10]	[8, 9, 10]	—
	12	—	—	[13, 14, 15]	[13, 14, 15]
Total fuzzy processing time	6	[68.2, 73.6, 81]	[83.3, 88.7, 96]	—	—
	8	[47.5, 53.4, 61.3]	[56.6, 62.5, 70.3]	[88, 93.9, 101.7]	—
	10	[53.3, 59.7, 68.1]	[63.9, 70.3, 78.6]	[97.4, 103.8, 112.1]	—
	12	—	—	[119.4, 125.8, 134.1]	[172.9, 179.3, 187.6]

algorithm under the three criteria to schedule and optimize the process. If it does not exceed, it means that the production can be completed in one batch, and the longest fuzzy processing time is regarded as the fuzzy completion time of this type of single crystal furnace.

Step 4: judging $\mathrm{Gen} \le G_{\max}$, if satisfied, let $Gen = Gen + 1$, otherwise, output the optimal allocation plan for silicon single crystal production according to the result with the smallest fuzzy completion time.

Step 5: we calculate the fitness value according to the fitness value to obtain the optimal individual $x$best and the ranking of the fitness value of the individual in the whole population.

Step 6: we obtain the adaptive mutation factor according to the ranking of fitness values and perform mutation operation according to formulas (25)–(28).

Step 7: we perform the crossover operation according to formula (30).

Step 8: we perform the selection operation according to formula (32), select individuals with lower fitness values to form a new population, and then go to step 3.

4. Simulation Test

The research in this article is based on the order task of a single crystal silicon manufacturer. The size of the silicon single crystal rods, charging amount of single crystal furnace and the corresponding processes production time are shown in Table 4. The scheduling results obtained by the proposed THSOA algorithm are compared to the results obtained by the manual scheduling. All simulations are implemented in MATLAB R2016a, Intel(R) Core (™) i7-7700 CPU @3.60 GHz processor with 8.00 GB memory programming.

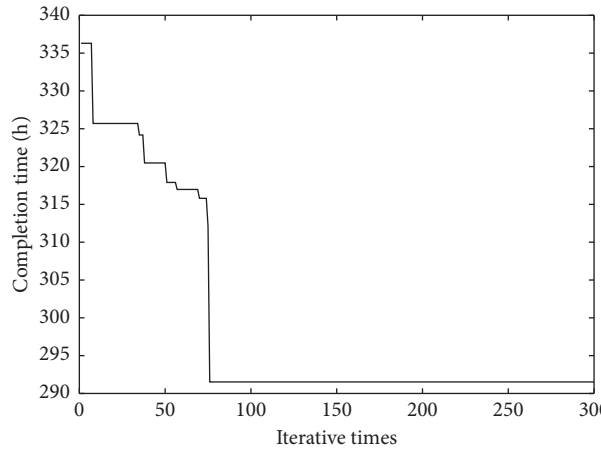

Figure 3: The convergence curve of optimization objective.

Table 5: The allocation results obtained by the TSHOA scheduling.

Silicon single crystal rod size	Thermal fields' size				Actual production quantity (kg)	Order quantity (kg)
	22 inch	24 inch	28 inch	32 inch		
6 inch	3	11	—	—	2010	2000
8 inch	11	1	7	—	3220	3200
10 inch	3	2	1	—	910	910
12 inch	—	—	2	6	2900	2778

Table 6: The allocation results obtained by the manual scheduling.

Silicon single crystal rod size	Thermal fields' size				Actual production quantity (kg)	Order quantity (kg)
	22 inch	24 inch	28 inch	32 inch		
6 inch	0	14	—	—	2100	2000
8 inch	0	0	13	—	3250	3200
10 inch	0	0	4	—	1000	910
12 inch	—	—	0	7	2800	2778

4.1. Parameter Settings. In the first stage, the IDE algorithm parameters are set as follows: the maximum variation factor is Fmax = 1, the minimum variation factor is Fmin = 0.1, crossover probability, the maximum crossover probability is $CR_{\max} = 0.9$, the minimum crossover probability is $CR_{min} = 0.1$, the population size is $Np = 60$, and the maximum number of iterations is 300. The VNS algorithm is used in the second stage, and the population size $Np = 50$, the maximum number of iterations is 100, the maximum number of searches is 10 times, and three kinds of neighborhood search structures in Section 3.3 are used.

The number of 6-inch, 8-inch, 10-inch, and 12-inch silicon single crystals is 30, 25, 22, and 18 pieces per kilogram, respectively. The order demand number of 6-inch, 8-inch, 10-inch, and 12-inch silicon single crystals is 60,000 pieces, 80,000 pieces, 20,000 pieces, and 50,000 pieces, respectively. And the number of 22-inch, 24-inch, 28-inch, and 32-inch single crystal furnaces is 5.

4.2. Experimental Results' Comparison and Analysis. According to the order demand, the objective of this study is that the fuzzy completion time is the shortest. Figure 3 is the objective convergence curve, the abscissa represents the number of iterations times, and the ordinate represents the convergent completion time. It can be seen from Figure 3 that the TSHOA has reached convergence nearly 80 iterative times and its convergence speed is fast, so the solution efficiency can be greatly improved in the process of solving the problem.

At present, the single crystal furnace of larger thermal fields is used to produce the corresponding size silicon single crystal rods by using the manual scheduling method. And the order allocation results obtained by the proposed algorithm and manual scheduling method are shown in Tables 5 and 6. When the quantity of silicon single crystals with different sizes is determined, it will be more wasteful to use manual methods, and the use of optimization-based algorithm will not only meet the production needs but also try to avoid resource waste. For example, it can be seen from Tables 5 and 6 that if the

Table 7: The production batch allocation results obtained by the TSHOA scheduling.

Silicon single crystal furnace size	Device ID	The first batch	The second batch	The third batch	The fourth batch
22-inch thermal field	M_{11}	2	3	3	—
	M_{12}	2	3	1	—
	M_{13}	1	1	2	—
	M_{14}	2	2	2	2
	M_{15}	2	2	2	2
24-inch thermal field	M_{21}	1	1	1	—
	M_{22}	1	1	1	—
	M_{23}	1	1	3	—
	M_{24}	1	1	—	—
	M_{25}	2	1	3	—
28-inch thermal field	M_{31}	1	1	—	—
	M_{32}	2	1	—	—
	M_{33}	3	1	—	—
	M_{34}	1	3	—	—
	M_{35}	1	1	—	—
32 inches thermal field	M_{41}	4	4	—	—
	M_{42}	4	—	—	—
	M_{43}	4	—	—	—
	M_{44}	4	—	—	—
	M_{45}	4	—	—	—

Table 8: The production batch allocation results obtained by the manual scheduling.

Silicon single crystal furnace size	Device ID	The first batch	The second batch	The third batch	The fourth batch
22-inch thermal field	M_{11}	—	—	—	—
	M_{12}	—	—	—	—
	M_{13}	—	—	—	—
	M_{14}	—	—	—	—
	M_{15}	—	—	—	—
24-inch thermal field	M_{21}	1	1	1	—
	M_{22}	1	1	1	—
	M_{23}	1	1	1	—
	M_{24}	1	1	1	—
	M_{25}	1	1	—	—
28-inch thermal field	M_{31}	1	2	1	—
	M_{32}	1	2	2	—
	M_{33}	1	1	1	1
	M_{34}	1	1	1	1
	M_{35}	1	2	1	—
32-inch thermal field	M_{41}	4	4	—	—
	M_{42}	4	4	—	—
	M_{43}	4	—	—	—
	M_{44}	4	—	—	—
	M_{45}	4	—	—	—

output quantity of a 6-inch silicon single crystal is 2000 kg, the optimal production result will also produce 2100 kg obtained by the manual method production, while only 2010 kg will be produced based on the optimization-based method, which will reduce the waste of 90 kg output resources.

The allocation results obtained by the TSHOA algorithm and the manual scheduling method are given in Tables 7 and 8, and the production batches are allocated in silicon single crystal furnace of different thermal fields size, among them, 1, 2, 3, and 4 stand for numerical index, which means silicon single crystal rods of 6 inch, 8 inch, 10 inch, and 12 inch, respectively.

From the batch allocation results of Tables 7 and 8, the solution results obtained by the optimization-based algorithm can be used for production in a 22-inch single crystal furnace. However, if the production cycle is shortened through the manual scheduling method, a large size silicon single crystal furnace can be used to produce small size silicon single crystal rods to improve production efficiency, but it will undoubtedly increase the burden of production energy consumption, material consumption, etc. Table 8 shows that the 22-inch silicon single crystal furnace is not used, which undoubtedly increases the waste of resources. From the batch allocation results of Tables 7 and 8 and the fuzzy processing times of Table 4, the fuzzy completion time

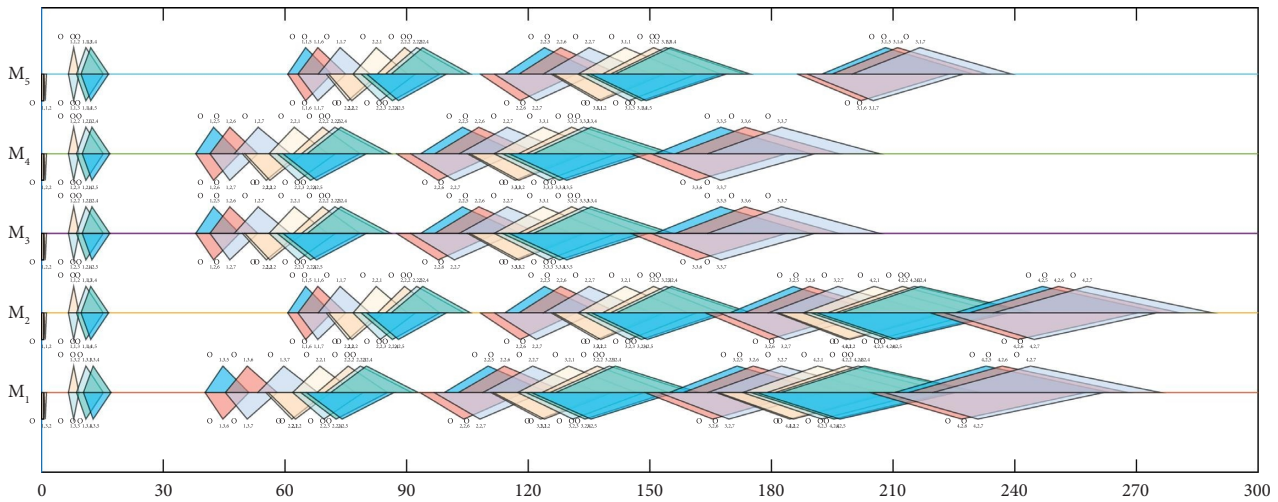

Figure 4: Fuzzy scheduling Gantt chart obtained by the 22-inch thermal field single crystal furnace.

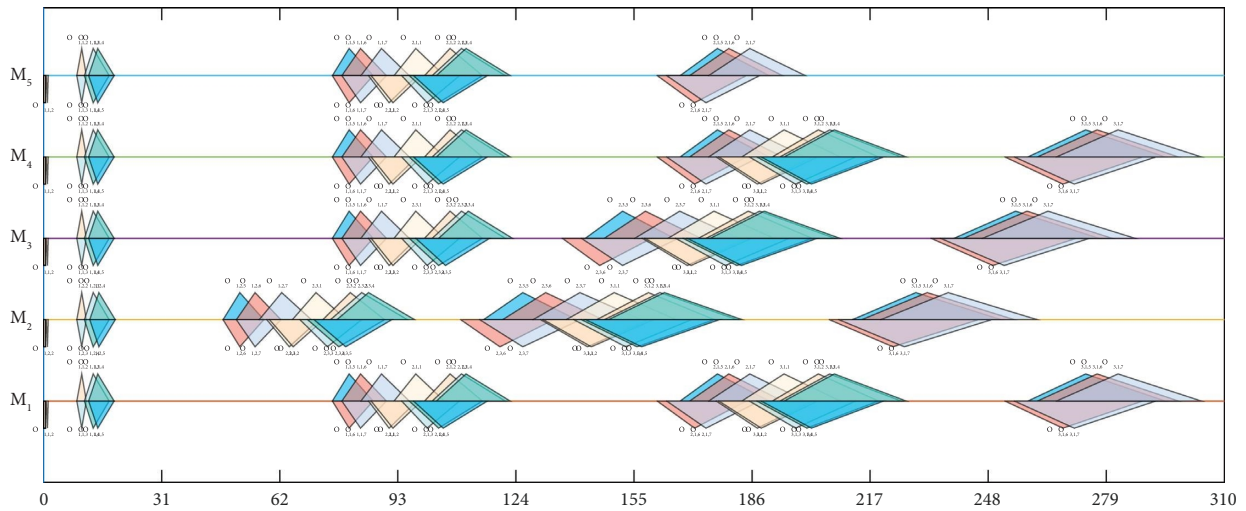

Figure 5: Fuzzy scheduling Gantt chart obtained by the 24-inch thermal field single crystal furnace.

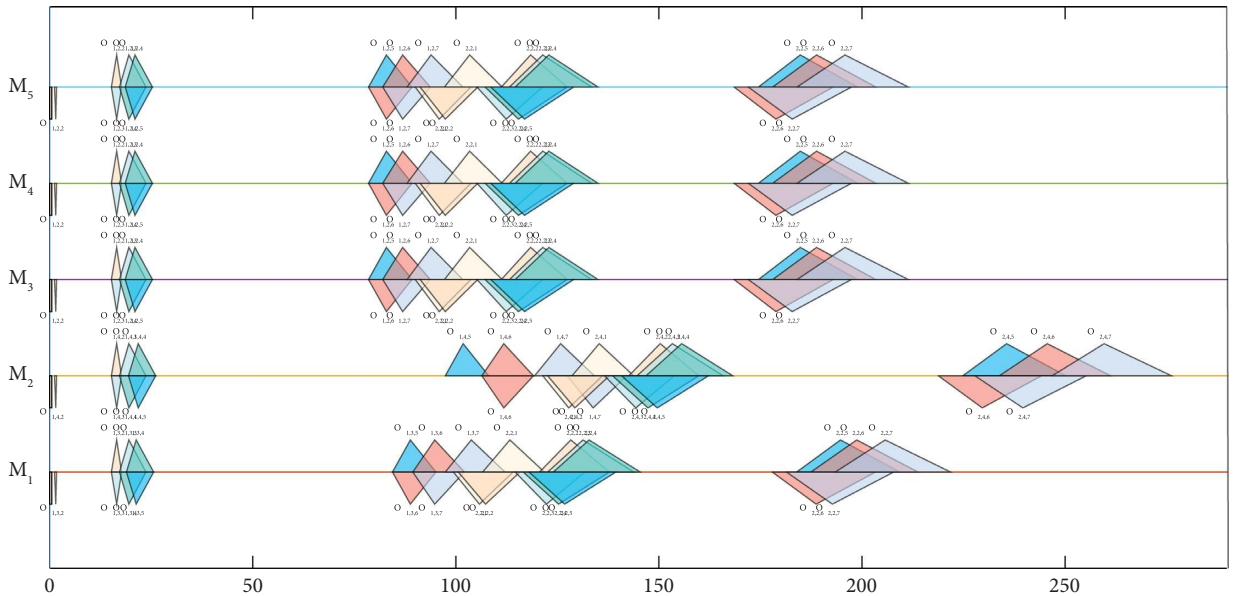

Figure 6: Fuzzy scheduling Gantt chart obtained by the 28-inch thermal field single crystal furnace.

of every size single crystal furnace can be obtained. The maximum fuzzy completion time obtained by the TSHOA and manual scheduling method is (273.9, 290.1, 312) and (384.4, 407.6, 438.8), respectively. So, the proposed TSHOA algorithm can greatly reduce the processing time, improve production efficiency, and thus reduce production cost.

Figures 4–6 are the fuzzy scheduling Gantt charts obtained by the single crystal furnaces corresponding to the 22-inch thermal field, the 24-inch thermal field, and the 28-inch thermal field, respectively. The lower triangle represents the fuzzy opening time of the process, and the upper triangle represents the fuzzy completion time of the process. There are three subscripts on the triangle, which represent the batch of processing, the size of processing, and the current process.

5. Conclusion

In this paper, the silicon single crystal production batch scheduling problem with multiple size under uncertain processing time is studied. Taking minimizing the fuzzy maximum completion time as the optimization objective, a two-stage hybrid optimization algorithm (TSHOA) is designed.

The main conclusions are as follows:

(1) The scheduling problem faced in the actual production process is transformed into a mathematical problem, and a scheduling mathematical model with the objective function of minimizing the maximum completion time is established.

(2) The algorithm mainly combines the respective advantages of the improved differential evolution (IDE) algorithm and the variable neighborhood search algorithm (VNS) to solve the scheduling mathematical model.

(3) The IDE was used to solve the problem of order quantity allocation for silicon single crystals with different sizes, so as to obtain the quantity of silicon single crystal rods with different sizes produced by different types of single crystal furnaces.

(4) The VNS algorithm is used to perform batch scheduling on the order size ranking problem to determine the optimal batch production process. The experimental results obtained by the optimization-based algorithm are compared with the manual method analysis to save production time and improve production efficiency.

(5) The article adopts fuzzy processing method to deal with the widely existing processing time uncertainty production scheduling problems, which is more in line with production requirements.

The limitations of the study are as follows:

This article is a preliminary study of fuzzy scheduling problem on multiple size silicon single crystal production using intelligent optimization algorithm. Firstly, this article only considers the completion time as a production index of the optimization objective. Although it can be recognized in theoretical simulation, it is more ideal in the actual production process. The actual production environment should also consider many factors such as energy consumption, economic cost, machine breakdown, and other objective. Secondly, the proposed intelligent optimization algorithm should be more universal, not just for solving the problems studied in this paper. Therefore, in the future research work, the proposed algorithm needs to be improved to solve more practical production scheduling problems.

Data Availability

The data supporting the findings of this study can be obtained from the corresponding author upon reasonable request.

Additional Points

The code run during the current study is available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Lu Kang wrote the article based on the experiment conducted by Guozheng Ping and the ideas conceived by YAli Wu and Ding Liu revised and improved the structure of article. All authors reviewed the manuscript.

Acknowledgments

The authors are particularly grateful to the National and Local Joint Engineering Research Center of Crystal Growth Equipment and System Integration for providing a part of the equipment to develop this research work. This work was supported by the Major Instrument Project of National Natural Science Foundation of China under Grant 62127809 and the Key Program of National Natural Science Foundation of China under Grant 61533014.

References

[1] L. P. Mark, "Analysis of the trend of China's semiconductor wafer plants," *Integrated Circuit Applications*, vol. 34, no. 5, pp. 22–26, 2017.

[2] L. Hans-Peter Wiendahl, *Load-Oriented Manufacturing Control*, Tsinghua University Press, Beijing, China, 1999.

[3] J. Zhang, L. Ding, and Y. Du, "Optimal control for heterogeneous phase transition temperature field of Czochralski monocrystalline silicon," *Control Theory and Applications*, vol. 38, no. 1, pp. 44–52, 2021.

[4] J. C. Ren, D. Liu, and Y. Wan, "Model-free adaptive iterative learning control method for the Czochralski silicon monocrystalline batch process," *IEEE Transactions on Semiconductor Manufacturing*, vol. 34, no. 3, pp. 398–407, 2021.

[5] X. Zhang, D. Liu, Y. Yang, and J. Liang, "An intelligent particle filter with adaptive M-H resampling for liquid-level estimation during silicon crystal growth," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[6] P.-S. Wang, T. Yang, and L. C. Yu, "Lean-pull strategy for order scheduling problem in a multi-site semiconductor crystal ingot-pulling manufacturing company," *Computers and Industrial Engineering*, vol. 125, pp. 545–562, 2018.

[7] Z. Wang, Q. Wu, and F. Qiao, "A lot dispatching strategy integrating WIP management and wafer start control," *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 4, pp. 579–583, 2007.

[8] M. Sakawa and T. Mori, "An efficient genetic algorithm for job-shop scheduling problems with fuzzy processing time and fuzzy duedate," *Computers and Industrial Engineering*, vol. 36, no. 2, pp. 325–341, 1999.

[9] O. Engin and M. K. Yilmaz, "A fuzzy logic based methodology for multi-objective hybrid flow shop scheduling with multi-processor tasks problems and solving with an efficient genetic algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 42, no. 1, pp. 451–463, 2021.

[10] M. G. Alharbi and H. A. El-Wahed Khalifa, "On a flow-shop scheduling problem with fuzzy pentagonal processing time," *Journal of Mathematics*, vol. 2021, Article ID 6695174, 7 pages, 2021.

[11] W. Chen, "Optimization of fuzzy distributed flexible job shop scheduling problem," *Modern Manufacturing Engineering*, no. 1, pp. 10–18, 2022.

[12] D. Lei, Y. Zheng, and X. Guo, "A shuffled frog-leaping algorithm for flexible job shop scheduling with the consideration of energy consumption," *International Journal of Production Research*, vol. 55, no. 11, pp. 3126–3140, 2017.

[13] D. Lei and D. Yang, "Research on flexible job shop scheduling problem with total energy consumption constraint," *Acta Automatica Sinica*, vol. 44, no. 11, pp. 2083–2091, 2018.

[14] J. Guo and J. Zhong, "Multi-objective optimization method of remanufacturing reprocessing shop scheduling with fuzzy processing time," *Control and Decision*, vol. 35, no. 06, pp. 1497–1504, 2020.

[15] J. Li, Z. Liu, C. Li, and Z. Zheng, "Improved artificial immune system algorithm for type-2 fuzzy flexible job shop scheduling problem," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 11, pp. 3234–3248, 2021.

[16] L. Rui and W. Gong, "An improved multi-objective evolutionary algorithm based on decomposition for bi-objective fuzzy flexible job-shop scheduling problem," *Control Theory and Applications*, vol. 39, no. 1, pp. 31–40, 2022.

[17] S. Li, R. Hu, and B. Qian, "Hyper-heuristic genetic algorithm for solving fuzzy flexible job shop scheduling problem," *Control Theory and Applications*, vol. 37, no. 2, pp. 316–330, 2020.

[18] L. Kang, D. Liu, Y. Wu, Y. Zhao, and G. Ping, "Multi-furnace optimization in silicon single crystal production plants by power load scheduling," *Journal of Process Control*, vol. 117, pp. 1–13, 2022.

[19] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[20] P. Xiao, D. X. Zou, and Z. L. Xia, "Application of improved differential evolution algorithm to dynamic economic dispatch," *Control Engineering China*, vol. 28, no. 2, pp. 275–283, 2021.

[21] L. Kang, D. Liu, Y. Wu, and G. Ping, "An improved DE algorithm for solving multi-furnace optimal scheduling of single crystal silicon production," *International Journal of Pattern Recognition and Artificial Intelligence*, 2022.

[22] Y. Zhang, Y. F. Tao, and J. M. Wang, "An improved DE algorithm for solving hybrid flow-shop scheduling problems," *China Mechanical Engineering*, vol. 32, no. 6, pp. 714–720, 2021.

[23] Q. Niu, B. Jiao, and X. Gu, "Particle swarm optimization combined with genetic operators for job shop scheduling problem with fuzzy processing time," *Applied Mathematics and Computation*, vol. 205, no. 1, pp. 148–158, 2008.

[24] D. E. Goldberg, *Optimization and Machine Learning*, Addison-Wesley, Boston, MA, USA, 1989.

[25] P. Hansen and N. Mladenović, "Variable neighborhood search: principles and applications," *European Journal of Operational Research*, vol. 130, no. 3, pp. 449–467, 2001.