Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7452638, 11 pages
https://doi.org/10.1155/2022/7452638

Research Article

The Implementation of Multiobjective Flexible Workshop Scheduling Based on Genetic Simulated Annealing-Inspired Clustering Algorithm

Ming Huang,[1] Fei Wang,[2,3] and Si Wu[4]

[1]College of Software, Dalian Jiaotong University, Dalian, 116028 Liaoning, China
[2]College of Mechanical Engineering, Dalian Jiaotong University, Dalian, 116028 Liaoning, China
[3]Department of Basics, Dalian University of Science and Technology, Dalian, 116052 Liaoning, China
[4]College of Foreign Languages, Dalian University of Science and Technology, Dalian, 116052 Liaoning, China

Correspondence should be addressed to Si Wu; wusi@dlust.edu.cn

Received 18 March 2022; Revised 23 April 2022; Accepted 6 May 2022; Published 20 May 2022

Academic Editor: Jun Ye

Copyright © 2022 Ming Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multiobjective flexible workshop scheduling is an important subject to improve resource utilization and production efficiency and enhance the competitiveness of enterprises. As the situation of resource constraints becomes more and more severe, the problem of companies rationally allocating limited resources in production is becoming more and more serious. Today, the manufacturing industry widely adopts advanced manufacturing modes such as computer-integrated manufacturing and intelligent manufacturing, but in these semi-intelligent manufacturing modes with a high degree of uncertainty and a high degree of personnel dependence, it is difficult to adapt to the work of large-scale production. Therefore, suitable clustering algorithms are urgently needed to help solve these problems, and this paper selects a clustering algorithm based on the genetic simulation annealing algorithm. This article is aimed at studying the problem of efficiency improvement in the production process of large-scale manufacturing and at finding a stronger and more effective production mode for the manufacturing industry. Firstly, this paper introduces the basic principles of simulated annealing genetic algorithm and regularized clustering algorithm. These algorithms have excellent performance in searching for global optimal solutions. They can be constantly tested and computed to keep the calculation results close to the global optimal solution. In this paper, the K-means clustering algorithm is used to select the shortest completion time to represent the clustering target. According to the minimum distance principle, the machine, workpiece, and other objects are input into the clustering of the algorithm, and the K-means algorithm will send out the sorting plan. Therefore, a multiobjective flexible job shop scheduling model based on genetic simulated annealing algorithm and clustering algorithm is established. Then, by using hypothetical production data to simulate the operation of the workshop, the scheduling model was applied to conduct a deduction and empirical comparative study. The experimental results showed that the model shortened the completion time of the workpiece by 4.4% and increased the average load rate of the machine by 10%.

1. Introduction

Simulated annealing algorithm, genetic algorithm, and clustering algorithm have been applied in the fields of grouping technology, job division and scheduling, equipment layout, vehicle routing, etc., and the application scope of various algorithms is continuously expanding. In recent years, the domestic market environment has undergone earth-shaking changes, and market trends are often the direction of change that the manufacturing industry needs to adapt. Changes in the market structure directly lead to changes in the production structure. This is mainly because the corresponding supply and demand relationship has changed. With the continuous advancement of information technology applications, people's ability to obtain information has been greatly improved, the people's cognitive abilities have continued to improve, and their living standards have continued to improve. With their full understanding of the market, the individual needs of the people will play a production-oriented role, leading to the diversification of product demand in the market. In the market competition of economic globalization and multilayered demand, enterprises are under increasing pressure to survive, and production methods tend to be diversified and small batches of customized production routes. How to enhance the core competitiveness of enterprises is a long-term and severe challenge faced by many enterprises.

With the rapid development of information technology and the rapid development of Internet of Things technology, the combination of Internet of Things technology and logistics technology has created many new opportunities for the manufacturing industry and also presented many challenges. Extract valuable information from a large amount of rapidly generated production data, thereby promoting the development of logistics data big data mining, providing decision support for logistics managers, and optimizing the structure of the logistics industry. This study found that the improved genetic simulation-based clustering algorithm is helpful to solve the multiobjective flexible workshop scheduling problem. Therefore, this paper has some practical significance for the related clustering algorithm based on the improved genetic simulation-based annealing algorithm. The case empirical evidence presented in this paper provides theoretical and practical references for promoting the reform of the manufacturing workshop.

Many scholars have conducted research on the use of advanced algorithms to improve the efficiency of workshop scheduling. Aiming at the problem of the lack of conformational design and weight training algorithms in neural network applications, Zhang et al. proposed a backpropagation neural network learning algorithm combined with simulated annealing genetic algorithm. The simulated annealing mechanism is built into the genetic algorithm to optimize the neural network at the same time. For the conformation and network weight design and optimization, this algorithm increases the amount of network weight calculation, thereby greatly increasing the loss, only suitable for laboratory research; it is difficult to apply in practical work [1]. Cen et al. proposed the genetic simulated annealing algorithm, which is an algorithm that combines genetic algorithm and simulated annealing algorithm, which is used in soil slope operations to quickly and accurately locate the critical slip surface of the soil slope to calculate the safety factor. They believe that compared with the golden section method based on the finite element calculation of the same stress field, the genetic simulated annealing algorithm is more accurate and efficient in searching for the critical slip surface of the slope of the earth-rock dam. The measurement is not in place [2]. Reddy et al. adopt a new method based on the hybrid technology of genetic algorithm and simulated annealing. This method is called the clustering algorithm concept of additive and split hierarchical clustering. They described a method by classifying the unit into various clusters which is a new method to solve the unit combination problem. Its purpose is to save fuel costs, reduce overall operating costs, and easily meet the minimum upper and lower constraints through the proper use of generator sets. However, the premise of this algorithm is that the generator set will never shut down. There is no failure, once an unexpected situation occurs. It will cause the algorithm to crash [3]. Aiming at the problem of high energy consumption of the spectral clustering algorithm in the construction of similarity matrix and feature decomposition of the overall data set, Wang et al. proposed a clustering algorithm based on grid division and decision graph to reduce the amount of calculation and improve the efficiency of clustering. In addition, they also added a decision graph method to quickly identify the clustering center and improve the stability of the algorithm. Numerical experiments show that the algorithm can effectively improve the efficiency of the spectral clustering algorithm. However, many scholars do not need any prior knowledge of the algorithm. Will this behavior cause the machine to learn every time it runs, resulting in repeated labor [4]. Aiming at the problem that the positioning system of the ship automatic identification system cannot find the ship navigation characteristics in real time, Li et al. propose an adaptive time interval clustering algorithm based on density grid. The algorithm can cluster the adaptive time interval according to the size of the real-time ship trajectory data. In order to efficiently and real-time discover the information of ship hot spots, the areas discovered by clustering algorithm are generally marine transportation hubs and existing hot issues, and their support for decision-making is very limited [5]. Wu et al. proposed a clustering algorithm based on adaptive multiresolution map, which can not only improve the efficiency of the calculation process but also obtain stable propagation results. Their research results have been applied to log facies analysis and improved the old method. The in-depth analysis of multidimensional logging curves based on the clustering of multiresolution graphs is very time-consuming and highly dependent on initial parameters. However, the research only focuses on the field of logging facies, and the research results can only be applied to this reference application in other fields [6]. Liu et al. have studied a model called flexible workshop. From the perspective of practical application, they use energy-sensitive production scheduling technology to minimize energy consumption. In order to deal with large-scale problems and further improve its energy efficiency, they designed another one. A special genetic algorithm makes the operation sequence of the model work change within certain constraints. The experimental numerical results show that the flexible workshop proposed by them has good effectiveness and efficiency, but the research optimizes energy efficiency, and there are few studies on machine efficiency and personnel efficiency [7]. In view of the unsustainable use of machines in the workshop, the preventive maintenance and transportation process of machines are considered. Wang et al. put forward the problem of flexible job shop scheduling considering the preventive maintenance activities and the transportation process and established numbers of ways to reduce the total energy consumption. The target flexible job shop scheduling model has less human demand, which is not in line with the actual situation, and more consideration should be given to the configuration of functional personnel [8].

The research innovation of this paper is mainly reflected in the improvement and practical application of the genetic simulated annealing algorithm. The main innovations are as follows: use the Metropolis criterion of the simulated annealing algorithm to improve the genetic algorithm selection operator, enhance the genetic algorithm while maintaining the good performance of the genetic algorithm selection scheme, and then, make the performance of the clustering algorithm based on the genetic simulation annealing algorithm better played in the application process. This paper uses genetic simulated annealing algorithm to solve the optimal scheduling problem of workshop production.

2. Multiobjective Flexible Workshop Scheduling Model Establishment Method

2.1. The Basic Principles of Simulated Annealing Genetic Algorithm. Scholars such as Metropolis proposed the simulated annealing algorithm in 1953. The algorithm is inspired by the similarities between many combinatorial optimization problems in mathematics and the solid-state cooling annealing process in physics. The proposed algorithm is a stochastic optimization algorithm based on Monte Carlo iterative solutions, which combines both SA and GA algorithms with good probabilistic optimization capability. Combining the probability jump ability of the algorithm, the solution of the objective function is randomly searched in the executable solution space of the problem. This iterative process gradually continues to find the global optimal solution of the problem and decides whether to accept the new solution according to the Metropolis criterion [9]. The purpose of designing the simulated annealing algorithm is to achieve the optimal value of the objective function. During the annealing process, the probability of accepting the optimized solution should be greater than the probability of accepting the degraded solution. As the temperature drops, it is more likely to accept the optimized solution. Close to zero, the probability of accepting the degraded solution also approaches zero [10]. Among them, the Metropolis criterion is the core theory of the simulated annealing algorithm. The criterion is defined according to the above theory, and the expression is

$$f(\Delta n, x) = \begin{cases} 1 & \Delta n \leq 0, \\ e^{-\Delta n/x} & \Delta n > 0. \end{cases} \tag{1}$$

In the formula, $f(x)$ represents the acceptance probability of the current new solution; Δn represents the difference of the objective function corresponding to the current new solution and the previous solution; and x represents the temperature control parameter during the iteration of the simulated annealing algorithm [11]. The Metropolis acceptance standard allows the algorithm to accept degraded solutions with a certain probability and prevents the algorithm from falling into a local optimum. A larger value of x is more likely to accept a degraded solution, but as the iteration progresses, x gradually decreases and approaches 0 [12]. The simulated annealing algorithm is different from the random algorithm that accepts all new solutions and the local search characteristics of the local search algorithm, which makes the algorithm have better global search capabilities and greater probability to obtain the best global solution [13].

The value of the fitness function is the only criterion for evaluating individual interests in the combination. The design of the fitness function is usually related to the objective function in question. In flexible workshop production scheduling problems, the fitness function is often related to the production time of the workshop. In the fastest scheduling problem, the scheduling goal is the shortest time at the end of the entire production process. At this time, the fitness function is generally designed as the reciprocal of the total scheduling processing time as the fitness function [14]. Assuming that there are n machines processing x workpieces, the processing completion time of the machine m_n to the workpiece g_x is $y(g_x, m_n)$. Since the processing time of the workpiece g_x on the machine tool m_n is $t(g_x, m_n)$ and the scheduling plan of the workpiece is $\{g_1, g_2, \cdots, g_x\}$, the time model until the completion of all workpieces is as follows:

$$\begin{gathered} y(g_x, 1) = t(g_1, m_1), \\ y(g_1, m_n) = y(g_1, m_{n-1}) + t(g_1, m_n), \\ y(g_x, 1) = y(g_{x-1}, 1) + t(g_x, m_1), \\ y(g_x, m_n) = \mathrm{MAX}\{y(g_{x-1}, m_n)\,; y(g_x, m_{n-1})\} + t(g_x, m_n), \\ y_{\max} = y(g_x, n). \end{gathered} \tag{2}$$

$x = 2, 3 \cdots, +\infty$ and $n = 2, 3 \cdots, +\infty$. In this scheduling problem, the fitness function is $f(y) = 1/y_{\max}$.

The coding methods currently used in genetic algorithms mainly include binary coding, decimal coding, and real number coding [15]. The orderly coding list (chromosome) of the process completion time is expressed as *[y1 y2 y3]*; the workpiece chromosome is expressed as *[g1 g2 g3]*; the machine is expressed as *[m1 m2 m3]*; thus, the machine list corresponding to this chromosome is *[y1m1 y2m2 y3m3]*. Accordingly, the processing sequence scheduling scheme of each workpiece on the machine is shown in Figure 1.

According to the design criteria of the fitness function of the genetic algorithm, the following conditions are mainly considered in the design of the fitness function: first, make the fitness function monotonic, continuous, and nonnegative, so that the optimization algorithm can find the maximum value. The goodness-of-fit value should reflect the strength of the solution, which is usually difficult to implement [16]. The function design should be as simple as possible to reduce the calculation time and improve the calculation efficiency. The design should improve the robustness of the fitness function as much as possible. The goal of this research is to find the minimum value of the time

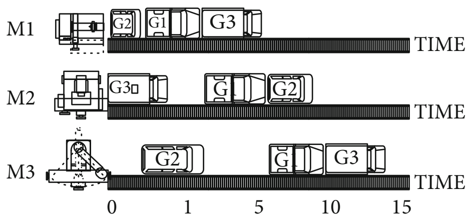

Figure 1: A feasible scheduling scheme based on process coding.

required to complete the workpiece, so it is assumed that the fitness function is

$$f(x) = \exp\{(ab)\cdot E(x)\}^{-1}, \tag{3}$$

where $f(x)$ represents the energy value of the individual, x represents the fitness of the individual chromosome, and a and b are variable parameters. The values are selected according to the duration of the algorithm and various actual environments to facilitate the annealing operation and the population. Adjusted in the diversity, there is greater flexibility [17].

Suppose G is the population size, g is the current population generation, and g_i is the maximum genetic generation. First, save the two individuals with the smallest and the largest fitness in the current population in the result set. When $g \le g_i/2$, randomly select $2g$ individuals from the current population for screening, arrange them in the order of the strength of individual fitness, and save the selection in the result set. The number of outstanding individuals is g. The probability of individual fitness value is selected in a proportional manner, and the fitness of individual i is obtained as S_i; then, the probability p_i of the individual being selected is

$$p_i = \frac{S_i}{\sum_{i\in[1,g]}^{G} S_i}. \tag{4}$$

This paper combines simulated annealing algorithm and genetic algorithm; combining genetic simulated annealing algorithm and simulated annealing algorithm can effectively avoid the premature phenomenon of traditional genetic algorithm, and at the same time, according to the specific situation, the fitness function and genetic code can be designed to make the algorithm more effective. Accelerate convergence [18]. Figure 2 shows a specific method that combines genetic algorithm and simulated annealing algorithm to obtain the global optimal solution.

The specific algorithm process is as follows: initialize algorithm control parameters, such as initial annealing temperature, temperature cooling rate, end iteration threshold, maximum iteration limit, and result probability [19]. Random initialization generates executable determinants and initial data sets for each set of executable solutions. Calculate the fitness value; perform genetic operations in the initial data set, such as selective crossmutation; apply the fitness function to new individuals to calculate the fitness value; and select new individuals using simulated annealing algo-

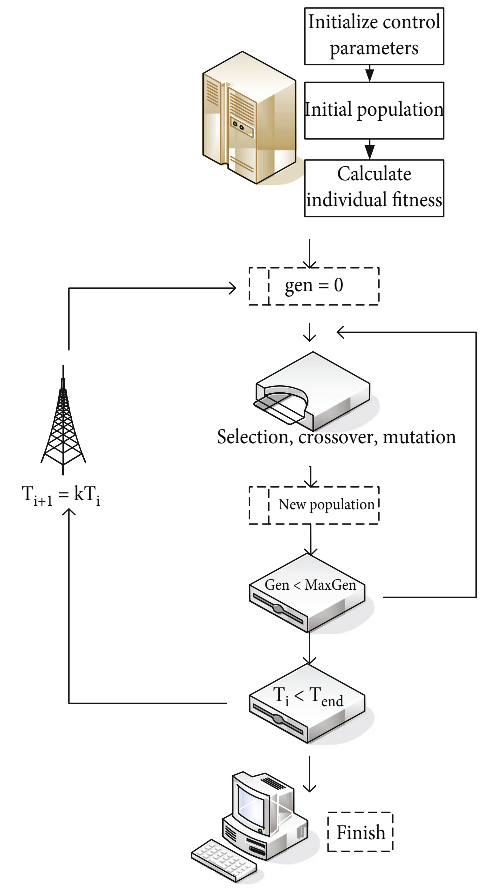

Figure 2: Genetic simulated annealing algorithm solution flow chart.

rithm [20], design the maximum number of loop iterations [21]. Determine whether the current temperature is lower than the end temperature. If it is lower than the end temperature, the algorithm ends and returns to the global optimal solution. If not, the temperature attenuation operation is cycled [22].

2.2. Regularized Clustering Algorithm. Everitt proposed the definition of clustering in 1974. Clustering is the process of dividing a collection of physical objects into multiple classes consisting of similar objects. The purpose of clustering is to divide a sample data set that has not been split into clusters and divide it into different clusters according to their unique similarities, so that the sample data with similar characteristics can be clustered into a single cluster, and determine that the interclass spacing of the sample data is greater than the intraclass spacing. Measuring similarity is a method used to measure the similarity between sample data. When measuring similarity, it is necessary to combine the components of the vector being measured, but there is no unified method to combine them. It should be decided according to the actual situation [23]. Therefore, various distance measurement formulas have appeared.

The method of Euclidean distance is to set *X1X2* as two n -dimensional model samples. It is necessary to note that the physical quantity of each feature vector and the unit of the physical quantity must be consistent in the corresponding dimension. The Euclidean distance measure is to measure the similarity of unit-consistent physical quantities. This is the use of Euclidean distance as a measure of similarity. The Euclidean distance formula is as follows:

$$D(X_1, X_2) = (X_1 - X_2)^T (X_1 - X_2) = (X_1 - X_2)^2 + \cdots + (X_n - X_{n-1})^2. \tag{5}$$

The accuracy of the Euclidean distance algorithm will significantly improve with the number of iterations of the model sample, which manifested by the concentration of the population distribution [24]. Figure 3 shows the distribution of the population in the search space during the iteration.

Overall, there are five poles in the figure, and the size of the poles gradually decreases [25]. When $(X_1 - X_2)^T = 1$, it can be seen that the population is more evenly distributed in the search space. As the number of iterations increases or decreases, the population gradually gathers towards the peak point. When $(X_1 - X_2)^T = 10$, the population gathers near the five peak points. Therefore, the Euclidean distance method is helpful for the clustering algorithm to search for multipeak extreme values and develop them in detail. The reason is that the Euclidean distance uses neighborhood information for local search. Once the neighborhood range of the peak point is determined, the algorithm will search for a better solution in the neighborhood and only find a better solution to replace the current solution [26]. Therefore, it is possible to always save nearby peak points in the iterative process and gradually reduce the neighborhood.

The Mahalanobis distance assumes two n-dimensional vectors X and Y, and its square expression is

$$D^2 = (\mathrm{X} - Y)^T C^{-1} (X - Y). \tag{6}$$

In the Mahalanobis distance equation, X represents the pattern vector, Y represents the average vector, and C represents the population covariance matrix [27]. The overall covariance matrix of each type of sample is expressed as

$$C = E\left\{(X - Y)(X - Y)^T\right\} = E\left\{\begin{matrix} (x1 - y1) \\ \vdots \\ (xn - yn) \end{matrix}\right\} [(x1 - y1) \cdots (xm - yn)]. \tag{7}$$

The advantage of Mahalanobis distance is that it can eliminate the correlation effect between model samples [28]. When $C = 1$, the distance can be regarded as Euclidean distance.

Set X1X2 as two n-dimensional pattern sample vectors; the formula is

$$D_m(X_i, X_j) = \left[\sum_{k=1}^{n} x_{ik} - {x_{jk}}^m\right]^{1m}. \tag{8}$$

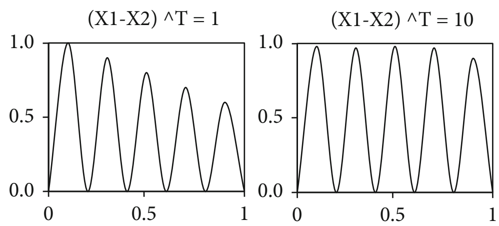

Figure 3: The distribution of the search space of the population with the evaluation times of the function.

Among them, the kth component of i and j is represented by X_{ik} and X_{jk}. When m is not equal to 1, $D_1(X_i, X_j) = \sum_{k=1}^{n} x_{ik} - {x_{jk}}^1$. When m is equal to 1, this distance represents the Minnesota distance.

The Hamming distance method is to set X_i and X_j as two n-dimensional binary sample vectors, and the formula is as follows, where X_{ik} and X_{jk} represent the kth component of i and j, respectively:

$$D_m(X_i, X_j) = \frac{1}{2}\left(n - \sum_{k=1}^{n} x_{ik} \cdot x_{jk}\right). \tag{9}$$

The Tanimoto measure is applicable to the case of binary features from 0 to 1, and the specific formula is as follows:

$$S(x_i, x_j) = \frac{X_i^T X_j}{X_i^T X_i + X_j^T X_j - X_i^T X_j} = \frac{x_i, x_j \text{ feature number in common}}{x_i, x_j \text{ total number of features}}. \tag{10}$$

The K-means clustering algorithm believes that the closer the distance between samples, the higher the similarity, and each cluster is composed of samples with similar distances [29]. The algorithm uses Euclidean distance to measure the similarity, the clustering criterion uses the square error sum function, and the reference function of the K-means clustering algorithm for the jth cluster is expressed as follows:

$$Y_y = \sum_{i=1}^{N_y} \left\| X_i - Z_y \right\|^2, \quad X_i \in S_y. \tag{11}$$

This equation solves the clustering problem by solving the extreme value optimization problem of the reference function. The clustering criterion is used to measure the difference or similarity function between the model sample data sets $\{S_j, j = 1, 2 \cdots, n\}$. Z_y represents the average value of the yth class of samples, which is the cluster center of the class; N_y represents the number of samples in S_y, and S_y represents the yth cluster. The algorithm has all Q modes:

$$Y_y = \sum_{y=1}^{Q} \sum_{i=1}^{N_y} \left\| X_i - Z_y \right\|^2, \quad X_i \in S_y. \tag{12}$$

Among them, Z_y represents the cluster center; N_y represents the number of samples in S_y, and S_y represents the yth cluster set (domain).

This paper uses the K-means clustering algorithm to randomly select n objects to represent the initial cluster center of each cluster: $Y1, Y2, \cdots, Yn$, and divide the remaining objects into the minimum distance between the clusters of the n cluster centers according to the principle. The formula is as follows:

$$Y_y(n) = \min \{X - Y_i(n), i = 1, 2 \cdots n\}, \quad X_i \in S_y(n). \quad (13)$$

Calculate the mean value of each cluster center again:

$$Y_y(n+1) = \frac{1}{N} \sum_{x \in S_y(n)} X, \quad y \in [1, n]. \quad (14)$$

Among them, N_y is the number of samples in the yth category. If $Y_y(n+1) \neq Y_y(n), y \in [1, n]$, then repeat formula (13) for an iterative loop until $Y_y(n+1) = Y_y(n), y \in [1, n]$, the algorithm converges, and stop the iterative loop [30]. In the double crescent data set experiment, two clustered data points were represented by two colors. In the two cases, multiple experiments were carried out, and the single-cluster data y was 60 and 120, respectively. Figure 4 is a schematic diagram of the original data point clustering results of the K-means clustering algorithm.

2.3. Flexible Workshop Production Scheduling Model. According to the method of describing uncertainty, there are three corresponding flexible scheduling modeling methods. One is to analyze and summarize the probability distribution of uncertain parameters and use random variables to describe uncertain parameters for analysis and modeling. Determine the parameters, and establish a production scheduling model; the third is to use fuzzy numbers to represent uncertain parameters, that is, parameter fuzzy scheduling model.

(1) The random distribution describes the scheduling model with uncertain parameters

$$\min_{j,x,l} \quad E\{f(j,x,l,\theta)\}$$
$$\text{s.t.} \quad \left\{ \begin{array}{c} g(j,x,l,\theta) \geq 0 \\ h(j,x,l,\theta) = 0 \\ j \in J, x \in X, l \in L, \theta \in \theta \end{array} \right\}, \quad (15)$$

where j is the decision coefficient representing the time required for product production and machine utilization; X is the decision variable representing the processing time, processing times, and quantity; and the objective function $f()$ represents the discrete variables in the production process, θ. The objective function $f()$ contains the shortest completion time, manufacturing cost, and customer satisfaction. The ultimate goal of scheduling is the expected value of the objective function, so the uncertain parameters in the model were represented by random variables

(2) Scheduling model for interval description of uncertain factors

$$\min_{j,x,l} \quad \tilde{f}\left(j,x,l,\tilde{\theta}\right)$$
$$\text{s.t.} \quad \begin{cases} \tilde{g}\left(j,x,l,\tilde{\theta}\right) \geq 0 \\ \tilde{h}\left(j,x,l,\tilde{\theta}\right) = 0 \\ j \in J, x \in X, l \in L, \tilde{\theta} \in \theta \end{cases} \quad (16)$$

If the distribution probability of the uncertain parameter is difficult to obtain, the interval of the uncertain parameter can obtained or the range of the uncertain parameter is relatively small; the interval $\theta \in [\theta_i \min, \theta_i \max]$ is used to obtain the uncertain parameter and establish the schedule interval planning model.

(3) Scheduling model of fuzzy parameters

$$\min_{j,x,l} \quad E\left\{\tilde{f}\left(j,x,l,\tilde{\theta}\right)\right\}$$
$$\text{s.t.} \quad \begin{cases} \tilde{g}\left(j,x,l,\tilde{\theta}\right) \geq 0 \\ \tilde{h}\left(j,x,l,\tilde{\theta}\right) = 0 \\ j \in J, x \in X, l \in L, \tilde{\theta} \in \theta \end{cases} \quad (17)$$

There are many uncertain parameters in the manufacturing process, and the probability distribution and interval range cannot obtained. Only the probability distribution of these uncertain parameters can be obtained. There are multiple distribution intervals and different distribution probabilities. Then, fuzzy numbers can be used to express uncertainty. Parameterize, establish, and solve the fuzzy scheduling model.

In order to verify the performance of the four algorithms mentioned above, this article uses these algorithms to simulate the shortest completion time of processes, machines, and workpieces in typical workshop scheduling. After comparing the calculation results, a comparison chart is obtained. It can be seen that the simulated annealing algorithm not only has fast convergence speed but also has good quality. It not only overcomes the premature phenomenon of genetic algorithms but also avoids the shortcomings of slow maturity of improved algorithms. Figure 5 shows the comparison of the convergence speed of each algorithm.

3. Workshop Production Scheduling Case Experiment and Analysis

3.1. Multiobjective of Flexible Workshop. The K-means algorithm optimizes the processing sequence and maximizes the

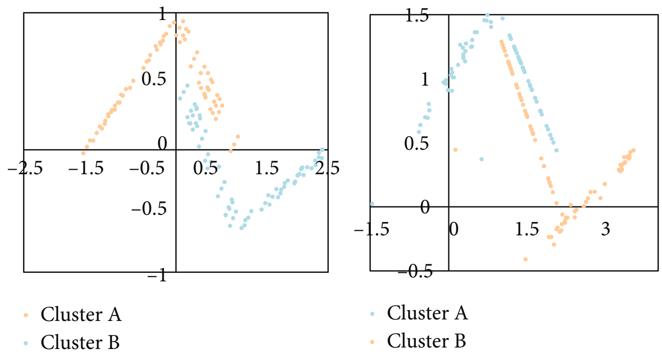

Figure 4: Clustering results of double crescent synthetic data.

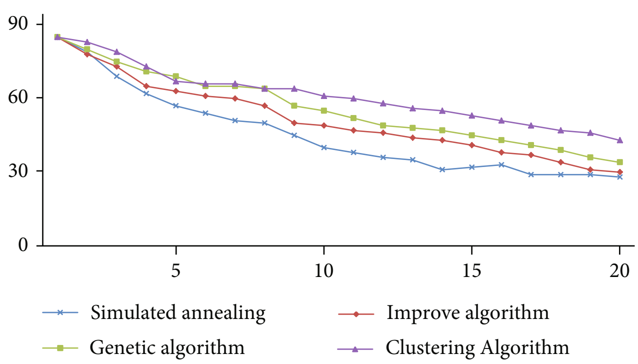

Figure 5: Comparison of algorithm convergence curves.

Table 1: Product processing schedule.

Product name	Processing time	Product name	Processing time
1	32	11	14
2	22	12	30
3	25	13	24
4	24	14	17
5	16	15	20
6	18	16	21
7	13	17	20
8	24	18	25
9	24	19	31
10	13	20	30

processing performance index. Flexible workshop production scheduling needs to determine all job statuses such as each machine and process constraints. The constraints are the processing time of each process and the processing sequence constraints of each job on each machine, processing start time, or completion time. Job shop scheduling is related to many factors, including equipment operating efficiency and employee work ability. As a multiobjective combinatorial optimization problem, predecessors proposed 27 scheduling schemes for shop scheduling. These goals can be divided into two categories: time-based goals and resource-based goals. The main characteristics of time-based goals are quick response to demand goals and satisfaction of customer needs, which are mainly reflected in the control of the completion time. It is uneconomical to complete too early and too late. If it is too early, accumulation time will be generated. It consumes storage space and deteriorates product liquidity. If it is too slow, it will not be able to meet customer needs. Improving resource utilization is also an important consideration in system research. The resource-based goal is to be able to use as few resources as possible to complete tasks and achieve better cost utilization. Taking cost as an evaluation value, it is not difficult to find today's society for energy conservation and higher requirements for emission reduction.

In actual production, many engineering problems are multiobjective optimization problems. In terms of workshop production scheduling, different departments of the enterprise also have different goals for scheduling decision-making. For example, the equipment department wants to maximize the utilization of machines and reduce the failure rate, the manufacturing department wants to increase the productivity of the workshop, and the sales department wants to hand over the goods as soon as possible to better meet customer needs. Therefore, when determining the production schedule of an enterprise, it is necessary to make a reasonable compromise between the interests of multiple parties.

Table 2: Processing scheduling and corresponding processing schedule.

Machine number	Corresponding product serial number	Total machine processing time
1	3, 7, 9	62
2	10, 13, 18	62
3	11, 17, 20	64
4	4, 5, 8	64
5	6, 12, 14	65
6	2, 15, 16	63
7	1, 19	63

3.2. Flexible Workshop Scheduling Model. The simulated annealing genetic algorithm has a strong global search

Table 3: Processing schedule for each calculation.

Calculation times	Minimum time	Calculation times	Minimum time
1	63	6	65
2	64	7	65
3	62	8	62
4	63	9	62
5	63	10	61
Average minimum processing time			63

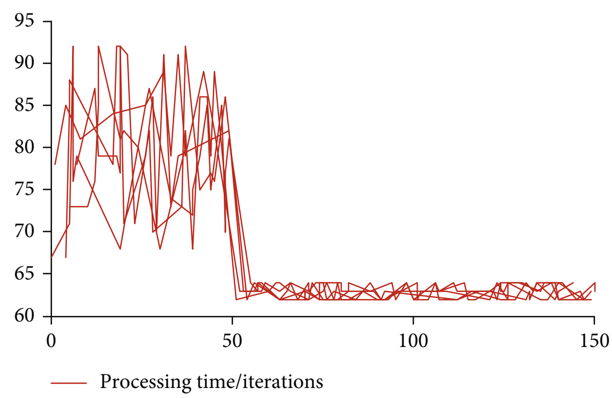

Figure 6: Graph of the results of 10 consecutive calculations.

function, so it is used by many researchers to solve other combinatorial optimization problems, including production scheduling problems. In order to observe the effect of applying simulated annealing algorithm in production scheduling problem, a simple scheduling case model is established here. The case is explained below: given seven machines with the same function, a total of 20 products must be produced. The production plan will be solved to minimize the total production time. The processing time of each product is shown in Table 1.

For the above cases, the simulated annealing genetic algorithm is applied to solve the problem. The designed scheduling system consists of six parts: system management, algorithm management, data entry, and output management. Among them, the scheduling algorithm management module can be used for algorithm selection, comparison, and configuration. Data import or input and scheduling process modules are the core of the entire system. The scheduling progress can be viewed in real time, and the output results can be output or formatted as needed. Then, carry on the simulation calculation to the above-mentioned problem; one of the results is shown in Table 2.

The shortest total processing time in the above table is 62. As a result of the solution, the processing time on each machine is balanced. In order to verify the stability of the algorithm, this paper uses the K-means clustering algorithm to solve the above problem 10 times, and the shortest processing time obtained is shown in Table 3.

It can be concluded from the above table that the algorithm is stable and the error generated is small. During 10 consecutive iterations of the K-means clustering algorithm, the corresponding results fluctuate as shown in Figure 6.

It can be observed that the simulation calculation results of the algorithm fluctuate greatly in the early stage and gradually fluctuate in the later stage. This is also in line with the characteristics of the algorithm, indicating that the algorithm is more stable in applications with simple scheduling problems.

In order to fully verify the performance and effectiveness of the genetic simulated annealing algorithm proposed in this paper, here is a simulation of a more complex situation.

That is, add different workpieces and processes to simulate a more real workshop situation, assuming that the relevant parameter data is shown in Table 4.

Table 4: Process and processing schedule.

Workpiece		Working procedure				
		1	2	3	4	5
Machine	W1	M5	M4	M3	M2	M1
	W2	M1	M2	M3	M4	M5
	W3	M4	M3	M2	M1	M5
	W4	M3	M4	M5	M1	M2
	W5	M2	M3	M4	M5	M1
Working procedure		Workpiece				
		M1	M2	M3	M4	M5
Processing time	1	1	2	2	2	8
	2	7	10	5	6	3
	3	5	8	5	3	10
	4	8	4	1	10	8
	5	10	3	2	5	9

Table 5: The machine's optimal processing sequence table.

		Workpiece				
Machine	M1	W2	W3	W4	W1	W5
	M2	W5	W3	W2	W1	W4
	M3	W4	W3	W5	W1	W2
	M4	W3	W4	W5	W1	W2
	M5	W1	W4	W5	W3	W2

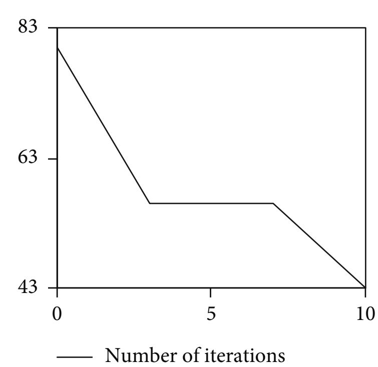

Figure 7: Curve graph of processing time and number of iterations.

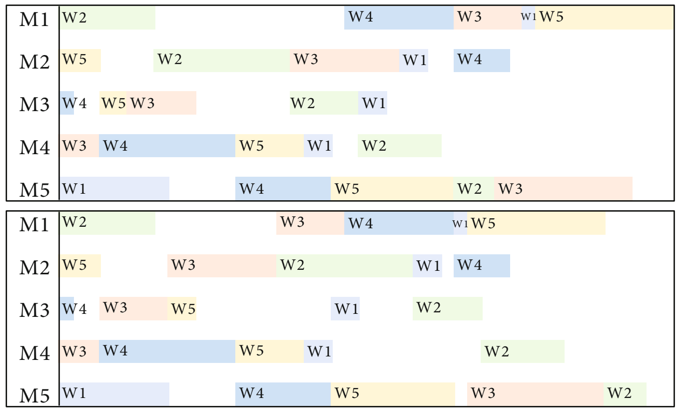

Figure 8: Workshop scheduling plan before and after optimization.

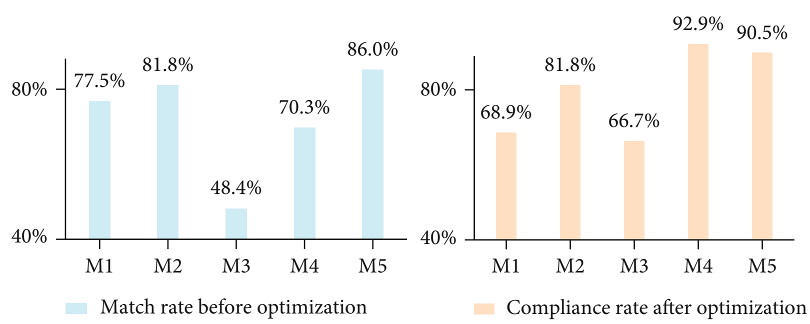

Figure 9: Machine load rate before and after optimization.

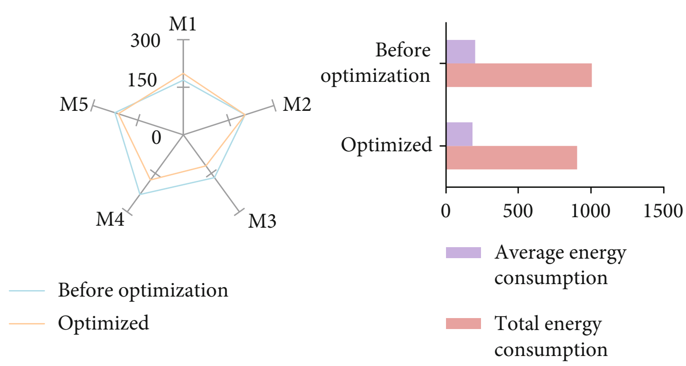

Figure 10: Comparison of energy consumption of each group of machines (kW·h).

In view of the above situation, this paper sets the initial parameter values according to the annealing algorithm and sets the machine procedures and processing time into a matrix arrangement. The optimal processing sequence obtained after applying the genetic simulated annealing algorithm proposed in this paper is recorded in Table 5.

It is calculated that the shortest processing time is 43, which has reached the optimal solution of the problem. Figure 7 is a graph showing the relationship between the processing time of the algorithm and the number of iterations.

3.3. Empirical Comparison of Production Scheduling. Based on the hypothetical production conditions in Table 4, randomly select a flexible workshop scheduling plan with an iteration number of 5, and then, take a plan with an iteration number of 10, which is, respectively, defined as before and after optimization, and make their Gantt. The plan is shown in Figure 8.

According to the data in Figure 8 and then according to the scheduling Gantt chart, the maximum processing completion time is observed. It is easy to find that the completion time before optimization is 45, and the optimization is 43. The production time efficiency is increased by 4.4%, because the system scheduling goal should be to make the machine load factor as high as possible to improve machine utilization even if the idle time of the machine is as short as possible; set the load rate = 1 − idle time/total start − up time. The comparison of the machine load rate calculated according to the Gantt chart is shown in Figure 9.

It can be observed that the optimized machine load rate is more averaged, the extreme value is higher, and the fluctuation is smaller. The average load rate before optimization is 72.8%, and that after optimization is 80.1%, and the machine load rate is significantly improved.

Then, calculate the power of each machine before and after the optimization, and get Figure 10.

Under the premise of the same workload, the optimized machine's startup time is shortened, and the energy consumption is also significantly reduced.

4. Discussion

This article has completed two aspects of method introduction and experiment, but limited by the author's level, some research is basic, and some algorithms proposed there need to be further improved. The following is the work that this paper thinks can be further improved: This paper studies the K-means clustering algorithm for multiobjective flexible job shop scheduling problem, but there are more and more nonnumerical data in practical work. According to the specific methods and design methods, developing an efficient and improved K-means big data clustering method to study these nonnumerical data is a very promising work. This paper mainly uses the hypothetical production data to simulate the operation of the workshop and trains a large number of logistics data to accelerate. How to improve the accuracy needs further research, such as using workpiece handover time, error-driven method to optimize the results, and the random storage of correctly classified data. This paper uses a simple model, a logistics data cloud computing scheduling research model considering only virtual machine. However, real-world cloud computing scheduling algorithms need to consider various aspects such as memory and maximum completion time. In this complex situation, how to comprehensively consider these resources to achieve this scheduling model and the ability to be applied to actual cloud computing task scheduling needs further research.

5. Conclusions

Relying on the K-means algorithm, this paper fully considers the multiple goals of workshop production, proposes a flexible workshop scheduling model, and conducts an empirical comparison study. Based on some production data assumed in the previous article, the two preoptimization and postoptimization models are deduced. As a result, the model proposed in this paper shortens the production time by 4.4%, increases the average load rate of the machine by 10%, and improves the efficiency of the machine. However, the study in this paper also has some shortcomings in some aspects; for example, the experimental method is not innovative enough, and the whole experimental process is still somewhat complicated. It is hoped that improvements can be made in future research to contribute more to improving the machine efficiency in the production workshop.

Data Availability

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Conflicts of Interest

The authors state that this article has no conflict of interest.

References

[1] D. Zhang, W. Li, X. Wu, and X. Lv, "Application of simulated annealing genetic algorithm optimized back propagation(BP) neural network in fault diagnosis," *International Journal of Modeling Simulation & Scientific Computing*, vol. 10, no. 4, pp. 1950024–1950049, 2019.

[2] W. Cen, J. Luo, J. Yu, and M. Shamin Rahman, "Slope stability analysis using genetic simulated annealing algorithm in conjunction with finite element method," *KSCE Journal of Civil Engineering*, vol. 24, no. 1, pp. 30–37, 2020.

[3] G. V. Reddy, V. Ganesh, and C. Srinivasarao, "Cost reduction in clustering based unit commitment employing hybrid genetic-simulated annealing technique," *Journal of Electrical Engineering & Technology*, vol. 14, no. 1, pp. 27–35, 2019.

[4] L. Wang, S. Ding, Y. Wang, and L. Ding, "A robust spectral clustering algorithm based on grid-partition and decision-graph," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 5, pp. 1243–1254, 2021.

[5] J. Li, H. Jiao, J. Wang, Z. Liu, and J. Wu, "Online real-time trajectory analysis based on adaptive time interval clustering algorithm," *Big Data Mining and Analytics*, vol. 3, no. 2, pp. 131–142, 2020.

[6] H. Wu, C. Wang, Z. Feng, Y. Yuan, H. F. Wang, and B. S. Xu, "Adaptive multi-resolution graph-based clustering algorithm for electrofacies analysis," *Applied Geophysics*, vol. 17, no. 1, pp. 13–25, 2020.

[7] N. Liu, Y. F. Zhang, and W. F. Lu, "Improving energy efficiency in discrete parts manufacturing system using an ultra-flexible job shop scheduling algorithm," *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 6, no. 2, pp. 349–365, 2019.

[8] H. Wang, B. Sheng, Q. Lu et al., "A novel multi-objective optimization algorithm for the integrated scheduling of flexible job shops considering preventive maintenance activities and transportation processes," *Soft Computing*, vol. 25, no. 4, pp. 2863–2889, 2021.

[9] A. Y. Prasad and B. Rayanki, "A generic algorithmic protocol approaches to improve network life time and energy efficient using combined genetic algorithm with simulated annealing in MANET," *International journal of intelligent unmanned systems*, vol. 8, no. 1, pp. 23–42, 2020.

[10] Y. Zhu, L. Zhou, and H. Xu, "Application of improved genetic algorithm in ultrasonic location of transformer partial discharge," *Neural Computing and Applications*, vol. 32, no. 6, pp. 1755–1764, 2020.

[11] M. F. Shirjini, S. Farzi, and A. Nikanjam, "MDPCluster: a swarm-based community detection algorithm in large-scale graphs," *Computing*, vol. 102, no. 4, pp. 893–922, 2020.

[12] X. Dong and Y. Cai, "A novel genetic algorithm for large scale colored balanced traveling salesman problem," *Future Generation Computer Systems*, vol. 95, no. JUN., pp. 727–742, 2019.

[13] Y. J. Yoo, "Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches," *Knowledge-Based Systems*, vol. 178, no. AUG.15, pp. 74–83, 2019.

[14] M. Liu, B. Liu, C. Zhang, and W. Sun, "Spectral Nonlinearly Embedded Clustering Algorithm," *Mathematical Problems in Engineering*, vol. 2016, Article ID 9264561, 9 pages, 2016.

[15] S. Rhim, C. Kim, and Y. Choi, "Coupled GA-SA optimization algorithm and application for compressive curved damper springs of automotive lockup clutches," *International Journal of Automotive Technology*, vol. 21, no. 3, pp. 641–647, 2020.

[16] X. Qin, J. Li, W. Hu, and J. Yang, "Machine learning K-means clustering algorithm for interpolative separable density fitting to accelerate hybrid functional calculations with numerical atomic orbitals," *The Journal of Physical Chemistry A*, vol. 124, no. 48, pp. 10066–10074, 2020.

[17] C. Tang, "A clustering algorithm based on nonuniform partition for WSNs," *Open Physics*, vol. 18, no. 1, pp. 1154–1160, 2020.

[18] Y. Wang, X. Liu, and L. Xiang, "GA-based membrane evolutionary algorithm for ensemble clustering," *Computational Intelligence and Neuroscience*, vol. 2017, Article ID 4367342, 11 pages, 2017.

[19] C. Shi, B. Wei, S. Wei, W. Wang, H. Liu, and J. Liu, "A quantitative discriminant method of elbow point for the optimal number of clusters in clustering algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 31, pp. 1–16, 2021.

[20] Y. Peng, B. Li, X. Mao, C. Li, H. Liu, and F. Peng, "Partition of the workspace for machine tool based on position-dependent modal energy distribution and clustering algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 108, no. 3, pp. 943–955, 2020.

[21] S. Loganathan and J. Arumugam, "Energy centroid clustering algorithm to enhance the network lifetime of wireless sensor networks," *Multidimensional Systems and Signal Processing*, vol. 31, no. 3, pp. 829–856, 2020.

[22] V. N. Kuchuganov, A. V. Kuchuganov, and D. R. Kasimov, "Clustering algorithm for a set of machine parts on the basis of engineering drawings," *Programming and Computer Software*, vol. 46, no. 1, pp. 25–34, 2020.

[23] A. Hamdi, N. Monmarché, M. Slimane, and A. M. Alimi, "Fuzzy rules for ant based clustering algorithm," *Advances in Fuzzy Systems*, vol. 2016, Article ID 8198915, 16 pages, 2016.

[24] S. Samji and H. Cattermole, "Promoting less-than-full-time (LTFT) training in your local trust: a workshop in conjunction with your champion of flexible training," *Future healthcare journal*, vol. 6, Suppl 2, pp. 12-13, 2019.

[25] M. Ramezani Mayiami, M. Hajimirsadeghi, K. Skretting, X. Dong, R. S. Blum, and H. V. Poor, "Bayesian topology learning and noise removal from network data," *Discover Internet of Things*, vol. 1, no. 1, p. 11, 2021.

[26] F. Mosquera, P. Smet, and G. V. Berghe, "Flexible home care scheduling," *Omega*, vol. 83, pp. 80–95, 2019.

[27] J. L. Andrade-Pineda, D. Canca, P. L. Gonzalez-R, and M. Calle, "Scheduling a dual-resource flexible job shop with makespan and due date-related criteria," *Annals of Operations Research*, vol. 291, no. 1-2, pp. 5–35, 2020.

[28] A. E. Ezugwu, A. K. Shukla, M. B. Agbaje, O. N. Oyelade, A. José-García, and J. O. Agushaka, "Automatic clustering algorithms: a systematic review and bibliometric analysis of relevant literature," *Neural Computing and Applications*, vol. 33, no. 11, pp. 6247–6306, 2021.

[29] B. Moore, P. Chair, J. Real, and W. Chair, "Summary of the 19th International Real-Time Ada Workshop," *Ada Letters*, vol. 38, no. 1, pp. 9–13, 2018.

[30] S. Bock, "Finding optimal tour schedules on transportation paths under extended time window constraints," *Journal of Scheduling*, vol. 19, no. 5, pp. 527–546, 2016.