Contents lists available at YXpublications

International Journal of Applied Mathematics in Control Engineering

Journal homepage: http://www.ijamce.com

Optimization Research of Production Scheduling in Hybrid Production Process Workshops

Zhonghua Han[a,b], Yu Zhang[a,*], Yuanyuan Liu[c]

[a] School of electrical and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China
[b] Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
[a] College of information and control engineering, Shenyang Urban Construction University, 110167, China

ARTICLE INFO

Article history:
Received 23 December 2023
Accepted 27 February 2024
Available online 3 March 2024

Keywords:
Hybrid production process
Process workshop
Genetic algorithm for compaction
Local assignment rules

ABSTRACT

In order to solve the problem of workshop scheduling optimization of the hybrid production process, a pseudo-reverse learning compact genetic algorithm integrating imulated annealing and adaptive-mutation (IAAMPRLCGA) was proposed. Combine it with local assignment rules to solve the problem. The IAAMPRLCGA algorithm improves the anti-precocious ability of the compact genetic algorithm, improves the quality and diversity of generating new individuals, gives full play to the efficient global search ability of the compact genetic algorithm, solves the problem that the compact genetic algorithm is easy to fall into local optimum, and further enhances the overall performance of the algorithm. The results show that the IAAMPRLCGA algorithm can effectively solve the problem of workshop scheduling optimization of hybrid production process when combined with local assignment rules.

1. Introduction

In today's manufacturing world, it is a crucial goal to complete production tasks at the lowest cost and the fastest speed [1]. In this context, production scheduling optimization has become one of the critical points for enterprises to occupy the market. The production process of ham sausage processing enterprises is a complex and unique hybrid production process that involves both continuous and discrete processes, which leads to the slicing state in the processing of ham sausages. This production process has a high degree of nonlinearity and is subject to multiple disturbances and severe coupling at the same time, and the pure hysteresis phenomenon is also quite prominent [2]. Therefore, for ham and sausage processing enterprises, the problem of scheduling optimization is particularly complicated. The manufacturing workshop of ham sausage processing enterprises belongs to a typical process manufacturing production workshop. In actual production, the production task of a particular specification of ham sausage is sent to the sterilization pot through the ligation process and then transferred to the packaging after being processed by the sterilization. This stage of the ham sausage production process comprises three processes. Because the process characteristics of the three processes are different, the production process of these three processes has the characteristics of mixed manufacturing and has a particular difficulty in solution.

This requires us to find more efficient ways to solve such complex problems. Compared with other swarm intelligence optimization algorithms, the Compact Genetic Algorithm (CGA) has the advantages of high efficiency, fast speed, and small computation. It is very suitable for solving complex NP problems [3]. Therefore, the research on the application of compact genetic algorithms is becoming more and more extensive. Wang Shengyao first introduced the distribution estimation algorithm into the field of scheduling to optimize the Makespan minimization problem [4]. Ha, and Mussetta modified and improved the compact genetic algorithm by implementing multiple probability vectors and adding appropriate learning schemes between these probability vectors and proposed a modified compact genetic algorithm [5], which was applied to the optimization synthesis of linear and planar sparse arrays of different sizes, and achieved good optimization results. Xue et al. proposed a novel method based on a compact genetic algorithm [6] to solve the aggregation problem of optimizing three different basic similarity measures and experimentally proved that the technique can significantly reduce the time and memory consumption while ensuring the correctness and completeness of the alignment.

Through the analysis of relevant literature in recent years, although the current scholars have solved the problem that the CGA algorithm is prone to precociousness, it has yet to significantly improve the optimization effect of the algorithm[7]. Therefore, this

* Corresponding author.
E-mail addresses: 34373988@qq.com (Y. Zhang)
Digital Object Identifiers: https://doi.org/10.62953/IJAMCE.344215

paper proposes a pseudo-reverse learning compaction genetic algorithm that integrates simulated annealing and adaptive variation. The hybrid algorithm has operational feasibility and many advantages, which can better absorb the respective advantages of the two algorithms, complement each other's defects, and better solve the problem of scheduling optimization in the process workshop of ham sausage processing enterprises.

2. Problem description and Mathematical Model

2.1 Problem Description

The production model of the finished product area of the ham sausage production line is shown in Figure 1. The scheduling optimization problem studied in this paper can be described as continuous rolling production in the finished product area of the ham sausage production line: ligation, tumbling, and packaging processes.

First, the ham sausage filling is processed in the previous process and then comes to the finished product area. In the actual production process, the filling of a specific type of ham sausage is sent to the sterilization pot in the sterilization process $Oper_2$ through the ligation process $Oper_l$ and then transferred to the packaging process $Oper_3$ for packaging after the sterilization process $Oper_2$ treatment, the ham sausage production process includes three processes, in the ligation process $Oper_l$, it is produced according to a specific processing speed, and different types of production tasks will be assigned to different ligation line production. The ligation process has typical manufacturing characteristics. In sterilization process $Oper_2$, the ham sausage is sterilized according to the whole pot. After the ligation of the ham sausage accumulates enough, it is transferred to the sterilization pot through the AGV trolley. Therefore, a ham sausage production order is generally split into multiple sub-orders during sterilization. At the same time, when the production line is laid, a ligature line is matched with multiple sterilization pots, which will be arranged into a sterilization unit. The sterilization process has typical discrete manufacturing characteristics, and the ham sausages produced by multiple sterilization pots that complete the same production order are sent to one or more packaging lines to match them in the packaging process $Oper_3$. Due to the different process characteristics of the three processes, the production process of the three processes has hybrid production characteristics. In this multi-process hybrid ham sausage production plant, the ligature line produces different types of ham sausages per unit of time. Therefore, the time taken for different types of ham sausages to fill AGVs is also different. A production order has a continuous processing time on the ligature line, but it is divided into multiple time segments in the sterilization process $Oper_2$. Although the ham sausage of the same production order will enter the matching packaging line after sterilization, the production task of this production order still shows the state of slicing in the production time dimension of packaging process $Oper_3$. Therefore, the production load of each packaging line is also different, and the workload of the packaging line could be more balanced due to the one-to-one or one-to-many relationship between each refractory pot production unit and the packaging line that matches it. Tracing the cause, the load imbalance is greatly affected by the ligature line's different processing speeds when producing different products. Therefore, this problem is difficult to solve. Figure 1 is the production model diagram of the finished product area of the ham sausage production line.

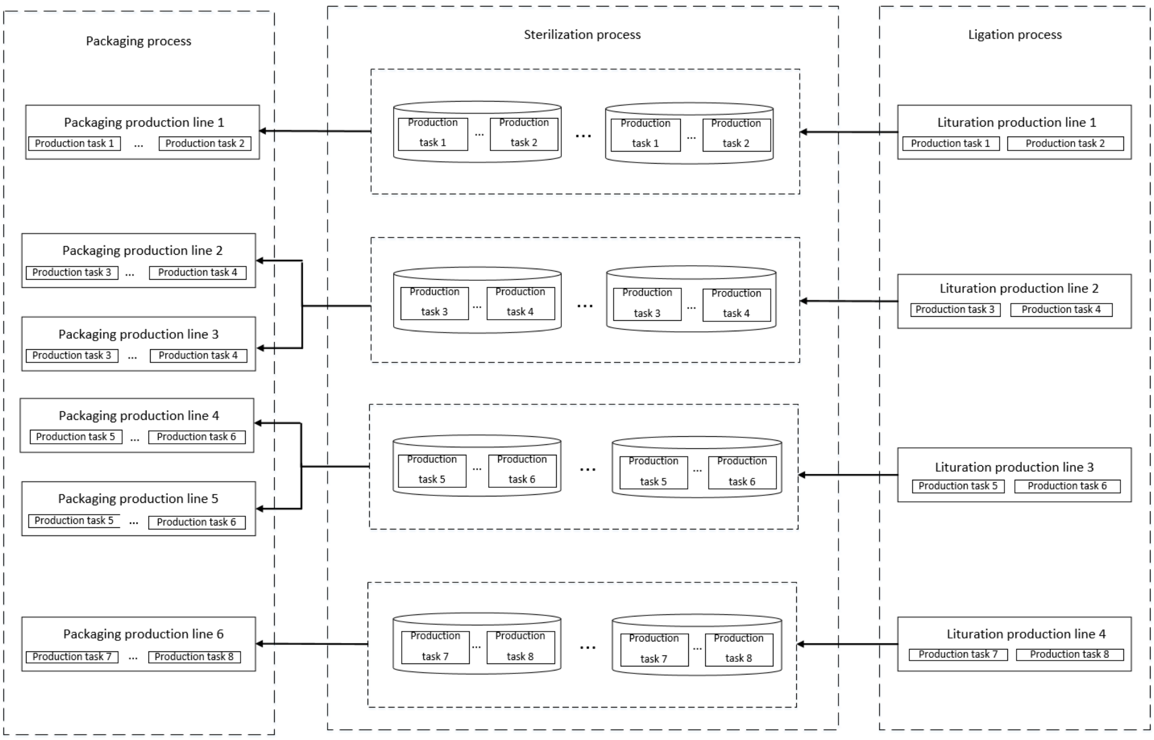

Fig.1 Production model of finished product area of ham sausage production line

2.1 Model Parameters

The process section model of the finished product area of the ham sausage production line includes the following parameters, as shown in Table 1.

Table.1 Model parameters of the production line in the finished ham sausage area

Parameters	Parameter description
$Oper_1$	Ligation process
$Oper_2$	Sterilization process

$Oper_3$	Packaging process
Nl	Number of ligation lines
Nsp	Number of sterilizers
Npk	Number of packaging lines
Wl_k	The k^{th} ligature line of the ligation process, $0 < k \le Nl$
Ws_l	The l^{th} sterilization pot in the sterilization process, $0 < l \le Nsp$
Wpk_m	The m^{th} packaging line of the packaging process, $0 < m \le Npk$
V_l	Standard processing speed for ligature lines
V_{pk}	Standard processing speed of the packaging line
$T_{l,i}$	The processing time required for production task Htc_i in the ligation process
$T_{s,i}$	The theoretical theory of the l^{th} sterilizer started time
T_s	The time it takes for a sterilizer pot to process a pot
$T_{pk,i}$	The total processing time required for production task Htc_i in the packaging process
$T_{r,i}$	The elapsed time of production task Htc_i on the packaging line
m_i	The total amount of processing for the i^{th} production task, $i \in \{1,...,n\}$
$m_{o,i}$	Specifications of the single product of production task Htc_i , $o \in \{1,...,q\}$
m_s	Standard capacity for one sterilizer， $0 < l \le Nsp$
q	The number of specifications and types of individual items for the production task
n	The total number of production tasks processed
Htc_i	Production task i ， $i \in \{1,...,n\}$

2.3 Basic Assumptions and Constraints

2.3.1 Hypothetical Variables

$$Al_{i,k} = \begin{cases} 1 & \textit{Production task } Htc_i \textit{ is assigned to ligature line } Wl_k \textit{ in ligation process } Oper_l \\ 0 & \textit{Production task } Htc_i \textit{ is not set to ligature line } Wl_k \textit{ in ligation process } Oper_l \end{cases} \quad (1)$$

$$As_{i,l} = \begin{cases} 1 & \textit{Production task } Htc_i \textit{ is assigned to sterilization pot } Ws_l \textit{ in sterilization process } Oper_s \\ 0 & \textit{Production task } Htc_i \textit{ is not transferred to sterilization pot } Ws_l \textit{ in sterilization process } Oper_s \end{cases} \quad (2)$$

$$Ap_{i,m} = \begin{cases} 1 & \textit{Production task } Htc_i \textit{ is assigned to packaging line } Wpk_m \textit{ in packaging process } Oper_p \\ 0 & \textit{Production task } Htc_i \textit{ is not set to packaging line } Wpk_m \textit{ in packaging process } Oper_p \end{cases} \quad (3)$$

2.3.2 General constraints on the finished product area of the ham sausage production line

$$\sum_{k=1}^{N_l} Al_{i,k} = 1 \quad (4)$$

$$T_{l,i} = \frac{m_i}{V_l * m_{o,i}} \quad (5)$$

$$T_{s,i} = \frac{m_s}{V_l * m_{o,i}} \quad (6)$$

$$T_{pk,i} = \frac{m_i}{V_{pk}} + T_{r,i} \quad (7)$$

Eq. (4) represents the constraint that production task can only select one ligature line for processing in the ligation process; Eq. (5) represents the processing time required by production task in the ligation process; Eq. (6) represents the theoretical starting time of the sterilizer corresponding to production task; and Eq. (7) represents the total processing time required by production task in the packaging process consists of two parts. One part is the processing time of production task on the packaging line, the other part is the flow time of production task on the packaging line.

2.3.3 Basic Assumptions

Based on the actual production process of the ham sausage processing industry, this paper makes the following assumptions:

(1) The equipment does not need additional switching time

(2) Any production line machine can process the production task, and the running process cannot be interrupted.

(3) The transportation time of semi-finished products between various processes in the running process needn't to be considered.

(4) The buffer between processes is infinite.

2.4 Evaluation Indicators of Production Scheduling Results

Eq. (8) and Eq. (9) take the maximum completion time as the evaluation index:

$$minC_{max} \quad (8)$$

$$C_{max} = \max\{C_{i,3}\}, \ i \in \{1,...,n\} \quad (9)$$

In Eq. (9), C_{max} represents the maximum value of all production tasks in the processing completion time of the last production process (packaging process), the time when the same batch of production tasks is fully processed.

3. Pseudo-reverse Learning Compact Genetic Algorithm That Combines Simulated Annealing and Adaptive Variation

3.1 Standard Compact Genetic Algorithm

A probability vector represents the population in the standard compact genetic algorithm. Two individuals are produced during each generation of evolution, and then the better individual is used to update the probability vector [7]. After multiple generations of evolution, if one of the probability values in probability vector is too large, similar gene fragments will appear in the same position in the newly generated individuals, thus reducing the diversity of the new individuals. At the same time, the developed unique individual, in turn, needs to be updated with the probability vector, which further increases the probability value in the probability vector.

When all probability values in the probability vector are 0 or 1, the compact genetic algorithm terminates evolution. Figure 2 shows the flow chart of the close genetic algorithm.

3.2 Pseudo-reverse Learning Compact Genetic Algorithm That Combines Simulated Annealing and Adaptive Variation

The standard compact genetic algorithm has a robust global search ability, but its local search ability is insufficient, which can easily lead to precocious convergence [8]. The simulated annealing algorithm is a relatively simple algorithm among intelligent algorithms, and although the algorithm is simple, the application results of the algorithm can often obtain an approximate solution of the global optimal solution, and the application performance of the algorithm is good [9]. However, the parallel processing ability of the simulated annealing algorithm could be better, and the computational efficiency could be higher.

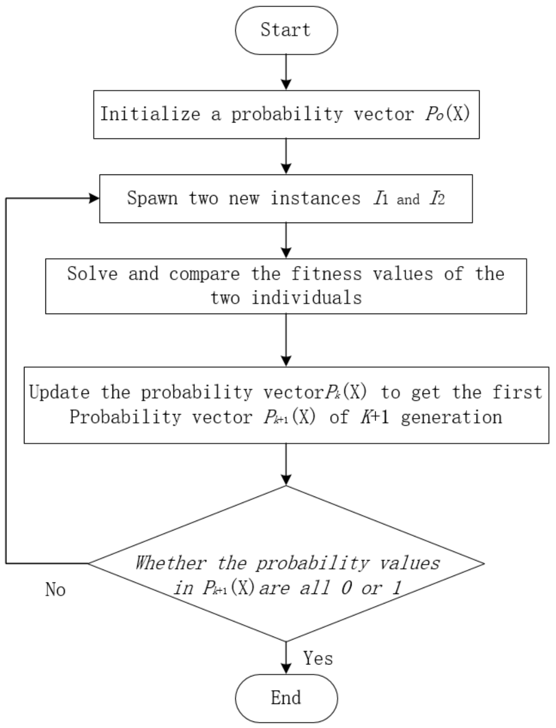

Fig.2 Flow chart of compact genetic algorithm

The defects of the algorithm are particularly prominent when dealing with the scheduling optimization problem of multi-machine, multi-product, and multi-process. Therefore, the fusion of the two algorithms can improve the compact genetic algorithm's anti-precious ability, improve the simulated annealing algorithm's computational efficiency, and reduce the simulated annealing algorithm's parameter dependence. Therefore, a pseudo-reverse learning compact genetic algorithm that combines simulated annealing and adaptive mutation is proposed by combining the close genetic algorithm with the simulated annealing algorithm and the simulated annealing algorithm system. The hybrid algorithm has many advantages, which can better absorb the respective advantages and complementary defects of the two algorithms, give full play to the high-efficiency in global search ability of the compact genetic algorithm and the sudden jump ability of the simulated annealing algorithm, improve the problem that the close genetic algorithm is easy to fall into local optimum, and further enhance the overall performance of the algorithm.

3.2.1 Pseudo-reverse learning initializes populations

The population quality of the initialization of the population algorithm will directly determine the superiority of the algorithm [10]. Therefore, initializing population quality is critical for the algorithm. The standard compact genetic algorithm generally uses the random initialization method to generate the initial population in the initialization stage. However, the method is significant and cannot ensure the diversity of the initialized population. Moreover, the uneven population quality makes the search time to converge to the optimal solution longer, resulting in a lower convergence rate [11]. In order to improve the performance of the compact genetic algorithm, this paper introduces the pseudo-reverse learning strategy into the algorithm.

Ref. [12] has theoretically proved that population initialization based on reverse learning can obtain a better initial solution, accelerating the convergence speed. Therefore, based on the reverse learning strategy, this paper proposes initializing the population using the pseudo-reverse learning strategy. Here is how it works:

Let the Ith individual in the initial population be $X_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,D})$, $x_{i,j} \in [a_j, b_j]$,where D is the dimension and $[a_j, b_j]$ is the range of the value of the jth dimension. The following formula $\breve{X} = (\breve{x}_{i,1}, \breve{x}_{i,2}, \ldots, \breve{x}_{i,D})$ can calculate the pseudo-reverse individual:

$$\breve{x}_{i,j} = a_j + b_j - x_{i,j},\ j = 1, 2, \ldots, D \tag{10}$$

$$\breve{x}_{i,j} = \begin{cases} rand(m_j, \breve{x}_{i,j}), x_{i,j} \le m_j \\ rand(\breve{x}_{i,j}, m_j), x_{i,j} > m_j \end{cases} \tag{11}$$

Where $m_j = \dfrac{b_j - a_j}{2}$, $rand(a,b)$ represent the random number in (a,b).

The steps to initialize the population with a pseudo-reverse learning strategy are as follows:

(1) Randomly initialized population $P = \{X_1, X_2, \ldots, X_N\}$;

(2) Peudo-reverse population $\breve{P} = \{\breve{X}_1, \breve{X}_2, \ldots, \breve{X}_N\}$ is calculated according to Eq. 10 and Eq. 11;

(3) Individuals with better fitness values were selected from the population set as the initial population. Equation 3.1 and Equation 3.2 processes are introduced, and the improvement of the standard reverse learning strategy is completed. The pseudo-reverse learning strategy is used to initialize the population, which helps to obtain a more evenly distributed high-quality solution in the population, thereby promoting the convergence speed of the algorithm.

3.2.2 Simulated annealing operation and adaptive mutation perturbation

Metropolis proposed simulated annealing (SA) in 1953 [13], characterized by retaining inferior populations under certain probability conditions, increasing the diversity of populations, and improving the ability to jump out of local optimizations to a certain extent.

With the gradual increase in the number of iterations, it is a common phenomenon that all individuals in the optimization algorithm population gradually evolve toward the optimal individual, which leads to a gradual decrease in the diversity of the population. If, in this process, the optimal individual happens to be the local optimal solution, then the algorithm may fall into a situation of premature convergence [14]. To avoid this, an adaptive mutation perturbation strategy is proposed, and its formula is as follows:

$$X_{b\_new}^{t} = \frac{t}{Maxiter} X_{b\_Gaussian}^{t} + (1 - \frac{t}{Maxiter}) X_{b\_Cauchy}^{t} \tag{12}$$

Wherein: $X_{b\_new}^{t}$ represents the new individual after the mutation of the optimal individual; t is the current number of iterations; $Maxiter$ is the maximum number of iterations; $X_{b\_Gaussian}^{t}$ denotes the individual after the Gaussian mutation; $X_{b\_Cauchy}^{t}$ means the individual after Cauchy's mutation.

The following conclusions can be drawn through the comparative observation of the density functions of the standard Gaussian and Cauchy distributions: the probability density of the Gaussian distribution is higher in the middle part and lower in the two parts. In contrast, the probability density of Cauchy distribution is relatively balanced in the middle and sides. This suggests that the Gaussian distribution is more inclined to produce smaller random numbers, while the Cauchy distribution is more inclined to make more significant random numbers. Therefore, the Gaussian variant has a strong advantage in local search, while the Cauchy variant is more suitable for global exploration [15].

A comparison of the density functions of the standard Gaussian and Cauchy distributions is shown in Figure 3. Gaussian is a Gaussian variant, and Cauchy is a Cauchy variant.

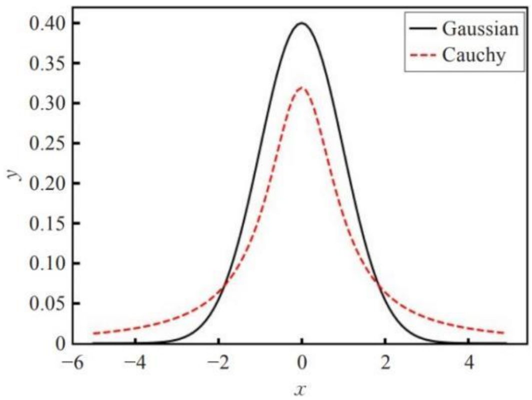

Fig.3 Comparison of standard Gaussian and Cauchy distribution density functions

From Equation 3.3, it can be seen that when the algorithm starts running, the t-value is small and the weight of Cauchy mutation is large. By using Cauchy mutation, a larger step size is obtained to avoid the algorithm falling into local optima. As the algorithm continues to run, the t-value is larger and the weight of Gaussian mutation is larger. Accurate search is performed through Gaussian mutation.

The adaptive mutation perturbation generates a new solution and is combined with the simulated annealing algorithm to accept the poor solution with a certain probability; the local optimal solution can jump out, making up for the lack of local search of the compact genetic algorithm. This strategy combines local search and global exploration to improve the performance and robustness of the algorithm.

3.2.3 Design of pseudo-reverse learning compact genetic algorithm combining simulated annealing and adaptive mutation

In the actual hybrid production process problem, there are often process differences between products, and the same product needs to be processed on different machines at different times. According to the process flow, the constraints are relatively many, and the complexity of the problem is further increased. To adapt to the algorithm's effectiveness under different production plans and product process constraints, this paper adjusts and improves the algorithm appropriately, enhances the scope of application of the algorithm, and simplifies the algorithm's operation.

(1) Algorithm initialization parameter design

① Annealing initial temperature. The initial annealing temperature determines the annealing efficiency and the probability of accepting the inferior solution, and there are differences in the determination of different problems and different constraint data, so the problem data determine the determination according to specific rules

$$T_0 = k * (f_{max} - f_{min}) \tag{13}$$

In Eq. 13:

k ----Generally take 20, 50, 100, etc.;

f_{max} ----The maximum fitness value of an individual in the initial population

f_{min} ----The minimum fitness value of an individual in the initial population.

②Annealing selection probability parameters. The Metropolis criterion is applied to the annealing selection probability, and the likelihood of selecting a relatively inferior solution is:

$$p_i = e^{\frac{f_i - f_b}{t}} \tag{14}$$

In Eq. 14:

p_i indicates the probability of selecting a newly generated individual in the population;

f_i indicates the fitness value of the newly generated individual;

f_b shows the fitness value of the individual selected in the previous iteration.

t indicates the current annealing temperature value.

③ Annealing coefficient. The selection of the annealing coefficient determines the speed of annealing temperature attenuation, generally considering the needs of the algorithm to solve the problem globally; the annealing coefficient is selected as a number close to 1 and is taken in the text.

(2) Coding and decoding Design

The process of generating new individuals according to the set probability model can be briefly described as follows: in the probability model, the value of each gene of the individual (from the 1st to the nth position), that is, the arrangement of the production task at the corresponding position, is determined according to the probability of each production task appearing at that position. Specifically, the number of production tasks is selected using roulette to determine the processing sequence of production tasks, and each gene locus is selected individually.

In the individual decoding process, the production tasks are distributed sequentially according to the information contained in the individual in the first process. The production tasks are selected for processing in the second and subsequent processes based on the First Available Machine First (FAMF) principle.

Through the above process design, the IAAMPRLCGA algorithm makes full use of the solution performance of the compact genetic algorithm. It combines the sudden jump ability of the simulated annealing algorithm, and the following simulation experiments verify the actual effect.

3.2.4 Pseudo-inverse learning compaction genetic algorithm steps that fuse simulated annealing and adaptive mutation

Step 1: Establish a probability matrix. Firstly, the probability matrix is used to replace the probability vector in the standard compact genetic algorithm. The probability matrix contains the distribution information and evolutionary trend of the current population genes, reflecting the core idea of compact genetic algorithm. Based on the individual encoding method in this article, establish a $n \times n$ matrix as a probability model P. In the probability model, rows 1st to n^{th} correspond to the workpiece J_1 to J_n ;Columns 1 to n correspond to the 1 to n genes of the individual, respectively. Elements in a probability model, such as $P_{i,s}^{L}$ representing the probability of a workpiece J_i appearing on an individual's s^{th} gene in the generation L.

Step2: Initialize the probabilistic model P, set the algorithm evolutionary algebra $L=0$.

Step3:Using pseudo reverse learning to initialize the population. Generate two new individuals, and the process of generating each individual is as follows: the 1st to n^{th} genes of the individual are sequentially selected using a roulette wheel method. The probability of the workpiece J_i appearing on the s^{th} gene is $P_{i,s}^{L}$. Once a workpiece is selected on a certain gene, the probability of the workpiece appearing at any position after that position is reset to zero. Unselected jobs continue to participate in the selection until all jobs complete the selection, and a new individual is generated. Using pseudo reverse learning, two individuals with better fitness values are identified as the initial population, denoted as I_1, I_2. Decode the individual I_1, I_2 and select the individual corresponding to the smaller maximum completion time as $BetterI$.

Step 4: A new solution I through adaptive mutation perturbation is generated through the probability model, and the new solution I competes with the optimal individual $BetterI$ obtained in Step 3, and the winning individual is recorded as $BestI$. If $BestI$ is the new solution, continue with Step 5; If $BestI$ is $BetterI$ in Step 3, then introduce the Metropolis criterion of simulated annealing algorithm to reselect new solutions I and $BetterI$, denote the selected individual as $BestI$, and continue with Step 5.

Step5:Guided by $BestI$, the probability model P is updated to evolve in the direction of $BestI$, while the evolutionary algebra $L=L+1$.

Step6:Determine any $P_{i,s}$ is either 1 or 0. If the condition is met, execute Step 8; If not satisfied, execute Step 7.

Step7:Determine whether the evolutionary algebra L has reached the set maximum evolutionary algebra L_{max}. If $L=L_{max}$, output the historical optimal individual and the evolution ends. Otherwise, perform a decay calculation with a decay coefficient of ∂ for the current temperature value $T_0\partial^n$ and return to Step 4.

Step 8: Output the optimal individual and the algorithm ends.

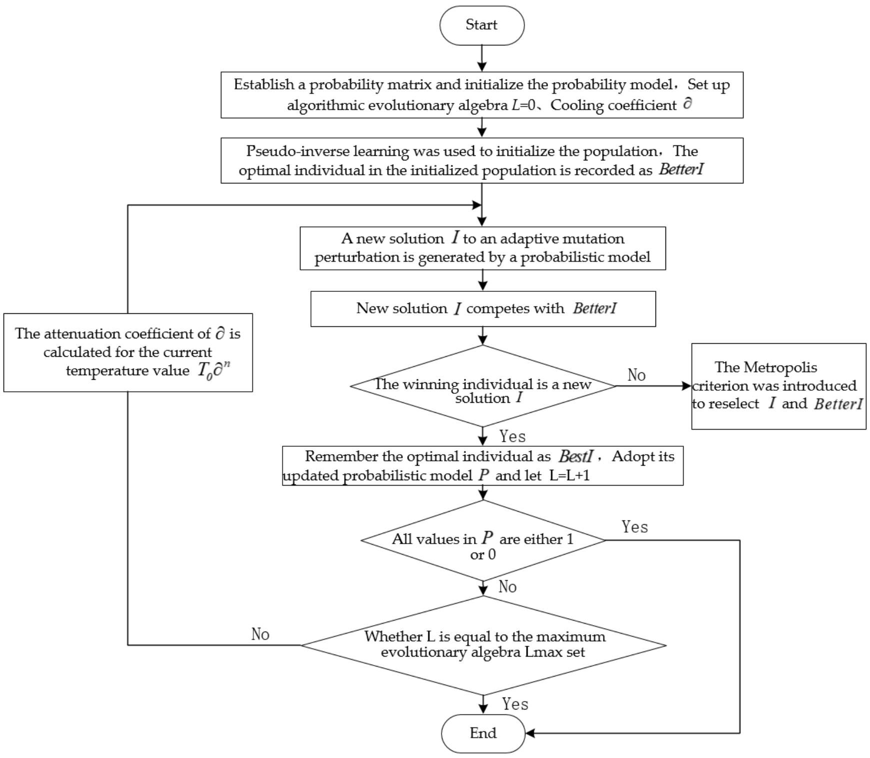

Fig.4 IAAMPRLCGA algorithm flow chart

4. Simulation Experiments

The simulation experiment program is written through Python 3.8 simulation software, the operating system is Windows 11, the processor is Corei5, the CPU is 2.60GHz, and the PC memory is 8GB.

Currently, the research on the optimization of mixed process scheduling is in the initial stage, and there is a lack of standard examples, so this paper uses the actual data of the ham sausage processing workshop to build the simulation data and simulate it.

In order to verify the effectiveness of the improved method, this paper designs multiple sets of simulation schemes under the same data scale. Then this paper test the effect of solving the optimization problem of mixed process workshop scheduling under four swarm intelligent optimization algorithms, namely genetic algorithm (GA), standard compact genetic algorithm (CGA), wolf pack optimization algorithm (WPA) and IAAMPRLCGA algorithm.

4.1 Algorithm Parameter Setting

For the sake of fairness and objectivity in the experiment, the parameter values of genetic algorithm and wolf pack optimization algorithm in this article refer to the settings in reference [15]: in genetic algorithm, the population size $NP = 100$, crossover probability $P_c = 0.8$, and mutation probability $P_m = 0.8$; in the WPA algorithm, the number of individuals in the population $NP = 100$. The number of competitive wolves q is 5, the search direction h is 4, and the movement step size $stepb$ is 0.3. Each experiment runs 20 times and the number of iterations sets to 500.

4.2 Mixed Ham Sausage Production Process Workshop Case Test

4.2.1 Construct simulation data

(1) Workshop model parameters

The production workshop of a ham sausage processing enterprise is a process workshop composed of multiple production lines, and this paper focuses on the finished product area of its production workshop. The production simulation data contain three processes: $\{Oper_1, Oper_2, Oper_3\}$, each ligature line corresponds to a fixed plurality of sterilization pots, and the sterilization pots corresponding to each ligature line are arranged into a sterilization group. Each ligature line corresponds to a different number of sterilizers, and the buffer zone between the sterilization process and the packaging process can be regarded as an infinite buffer.

Table.2 Model parameter table

Model parameters	Parameter description	Parameter value
Nl	The number of production lines corresponding to operation $Oper_1$	4
Nsp_1	The number of sterilizers corresponding to the first production line	10
Nsp_2	The number of sterilizers corresponding to the second production line	6
Nsp_3	The number of sterilizers corresponding to the third production line	8
Nsp_4	The number of sterilization pots corresponding to the fourth production line	14
Npk_1	The number of packaging lines corresponding to the first production line	2
Npk_2	The number of packaging lines corresponding to the second production line	5
Npk_3	The number of packaging lines corresponding to the third production line	6
Npk_4	The number of packaging lines corresponding to the fourth production line	6

(2) Parameters of the processing object

The total number of ham sausage production task attribute information $X = 2$, $Prop_1$ represents the single item specification attribute of the production task, $Prop_2$ represents the entire task attribute of the production task, the single item specification attribute of the production task $PropValue_1 = \{HtcType_1, HtcType_2, HtcType_3\}$, and the total task attribute of the production task $PropValue_2 = \{HtcQuota_1, HtcQuota_2, HtcQuota_3\}$.

Table.3 Item specifications and total quantity attribute information table for production tasks

attributes	Single item specifications $Prop_1$	Total $Prop_2$
Htc_1	$Type_1$	$Quota_1$
Htc_2	$Type_2$	$Quota_2$
Htc_3	$Type_3$	$Quota_3$
Htc_4	$Type_2$	$Quota_4$
Htc_5	$Type_1$	$Quota_4$
Htc_6	$Type_3$	$Quota_4$

Table.4 Standard machining timesheets for production tasks

	$Oper_1$	$Oper_2$	$Oper_3$
Htc_1	111	120	60
Htc_2	238	300	150

Htc_3	267	480	240
Htc_4	143	180	90
Htc_5	167	180	90
Htc_6	100	180	90

4.2.2 Simulation scheme

The IAAMPRLCGA algorithm, GA algorithm, WPA algorithm, and CGA algorithm were combined with the FAMF principle as global optimization algorithms to solve the scheduling optimization problem of a ham sausage processing enterprise, and the effect of each scheme was further analyzed. A total of 4 sets of simulation schemes were designed, and the assignment rules of simulation schemes 1~4 were FAMF, and the information on the four sets of simulation schemes is shown in Table 5.

Table.5 4 groups of simulation scheme information

Simulation scheme	Global optimization algorithms	Assignment rules
Schem1	GA	FAMF
Schem2	WPA	FAMF
Schem 3	CGA	FAMF
Schem4	IAAMPRLCGA	FAMF

4.2.3 Simulation results and analysis

(1) Evaluation indicators

In the simulation experiment, the maximum completion time C_{max} is used as the fitness function value of the optimization algorithm, and the goal is to minimize the full completion time. The four sets of simulation schemes were run 20 times to calculate their respective averages, and the results are shown in Table 6.

Table.6 Comparison table of evaluation indicators of the results of production scheduling in 4 groups

Evaluation indicators		scheme1	scheme2	scheme3	scheme4
C_{max}	Optimum	418	412	408	398
	Worst	425	422	414	404
	average value	421	416.7	410.8	402

Based on the detailed analysis of the data listed in Table 6, it is clear that the IAAMPRLCGA algorithm presents significant advantages in solving the optimization problem of maximum time to completion. Specifically, compared to other optimization algorithms, the total completion time of scheme four is reduced by 4.8%, 3.4%, and 2.5% comparing with scheme one and scheme 3, respectively. This result not only highlights the excellent performance of the IAAMPRLCGA algorithm in the field of scheduling optimization but also shows its significant improvement in terms of maximum completion time.

The box plot is shown in Figure 5. Further observation of the box plot shows the trend of fitness values for GA, WPA, CGA, and IAAMPRLCGA optimization algorithms. It is worth noting that the fitness values of all algorithms show a decreasing trend as a whole, which indicates that the performance of the optimization algorithm gradually improves with its iterative progression. However, compared with other algorithms, the boxplot of the IAAMPRLCGA algorithm shows a more compact data distribution and no outliers, which shows that the proposed algorithm is also excellent in terms of stability.

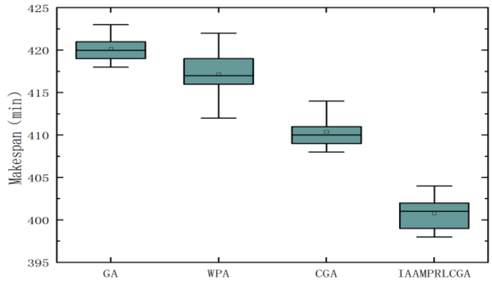

Fig.5 Fitness value box diagram

(2) Gantt chart analysis of production scheduling results

Figure 6 shows the Gantt chart of the 10th experimental results in the IAAMPRLCGA algorithm, which is the optimal Gantt chart. Different colors in the diagram represent different types of production tasks, and the Gantt chart can be used to obtain the online processing sequence of production tasks. Htc_1、Htc_2、Htc_3 and Htc_4 first enter the ligation production lines 1、2、3 and 4, respectively, and then carry out continuous rolling production in the corresponding sterilization and packaging units. At t=398, the ham sausage production line will complete all production tasks.

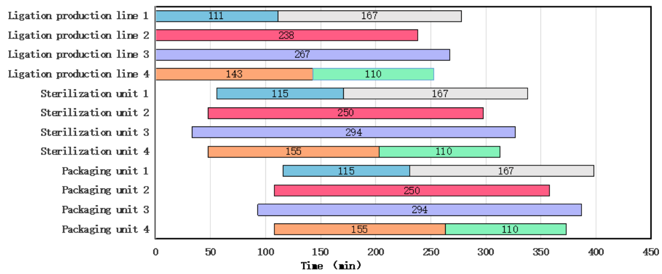

Fig.6 Gantt chart for optimal scheduling

(3) Analysis of the evolutionary process of scheduling

According to the iterative curves of the optimization values of the four optimization algorithms IAAMPRLCGA, GA, CGA, and WPA under the same actual simulation data shown in Figure 7, it can be found that with the increase of training algebra, the final optimization values of the four algorithms show a trend of gradually decreasing and tending to stabilize. In the initial stage of the algorithm, due to the large population size of the GA algorithm and the WPA algorithm, they show a significantly fast convergence speed and demonstrate a strong search ability for the solution space. However, although the GA algorithm shows a fast convergence speed in the search process, its optimization ability is significantly inferior to that of the other three algorithms when the maximum number of iterations in the graph is 500. At the same time, although the WPA algorithm has a fast convergence speed, it is easy to fall into local extremes and stop evolving within 85 generations, resulting in poor optimization results. In the process of evolution, the probability value distribution in the probability model decreases rapidly, which leads to the decline of its search performance, and the evolution stagnates in the 149th generation, falling into local extremums. Although its optimization speed is slightly more potent than the WPA algorithm, it still needs to improve its ability to jump out of local extremes and insufficient evolutionary vitality.

On the contrary, the scheme using the IAAMPRLCGA algorithm shows a fast search performance similar to that of the CGA algorithm in the initial stage. However, in the process of continuous evolution, the distribution of probability values in probability models rapidly decreases. The search performance also declines, and the evolution stagnates at 140 generations, falling into local extremums. At this time, the simulated annealing algorithm is introduced to regain the evolutionary ability of the algorithm. At the same time, the ability of the algorithm to search the solution space is enhanced, and the algorithm finds the optimal solution when it is iterated to the 275th generation. The results show that the IAAMPRLCGA algorithm has significant advantages in dealing with optimization and incredibly complex scheduling problems.

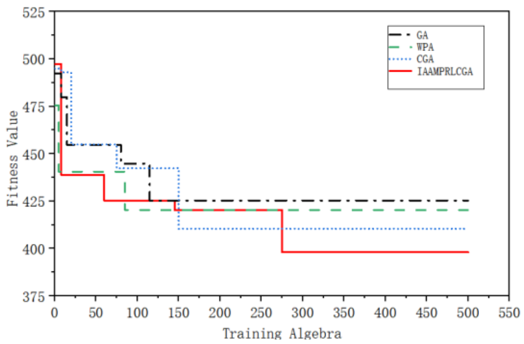

Fig.7 Optimization value iteration graph

5. Conclusion

To solve the hybrid production Process Workshop Scheduling Optimization Problem in a ham sausage processing enterprise, a pseudo-reverse learning compact genetic algorithm combining simulated annealing and adaptive variation was proposed. Through simulation experiments, it is known that the IAAMPRLCGA algorithm can better solve problems such as easily falling into local optima and premature convergence compared to other swarm intelligence optimization algorithms, and more effectively solve the scheduling optimization problem of hybrid production process workshops. Broadening the application scope of the IAAMPRLCGA algorithm in scheduling optimization problems is the next research direction.

References

Zhou Hu. Application analysis of intelligent control technology in food machinery[J]. Modern Food, 2022, 28(18): 99-101.

Fan Jiongming, Hu Shanying, Chen Dingjiang, et al. An essential analysis of the process manufacturing industry[J]. Strategic Study of Chinese Academy of Engineering, 2017, 19(03): 80-88.

Han Zhonghua, Zhu Yixing, Shi Haibo, et al. Research on batch scheduling optimization problem of flexible assembly line workshop based on improved compact genetic algorithm[J]. Systems Engineering - Theory & Practice, 2016, 36(06):1616-1624.

Wang Shengyao, Wang Ling, Xu Ye, et al. Distribution estimation algorithm for solving the scheduling problem of mixed assembly workshop[J]. Acta Automatica Sinica,2012,38(03):437-443.

Ha B V, Mussetta M, Pirinoli P, Zich R E. Modified compact genetic algorithm for thinned array synthesis[J]. IEEE Antennas And Wireless Propagation Letters, 2016, (15):1-6.

Xue X S, Liu J H, Tsai P W, Zhan X Y. Optimizing ontology alignment by using compact genetic algorithm[J]. International Journal Of Computational Intelligence And Security, 2015, (11):231-234.

Han Zhonghua, Zhang Quan, Shi Haibo, et al. Optimization problem of multi-sequence finite buffer scheduling in flexible flow workshop with preparation time[J]. Journal of Mechanical Engineering, 2019, 55(24): 236-252.

Peng Jun, Liu Zhen, Xu Xuewen. Novel compact genetic algorithm and its performance analysis[J]. Computer Technology And Development, 2015, 25(11): 120-124.

Yi Bin, Jing Shuyao. Cargo location optimization of automated three-dimensional warehouse based on improved simulated annealing genetic algorithm[J]. Journal of Packaging Science,2023,15(03):76-84.)

Zhang Bin, Li Yanhui, Guo Hao. Cross-population differential evolution algorithm based on reverse learning[J]. Journal of Computer Applications, 2017, 37(04): 1093-1099.)

Han Zhonghua, Zhu Yixing, Shi Haibo, et al. Research on batch scheduling optimization problem of flexible flow workshop based on improved compact genetic algorithm[J]. Systems Engineering - Theory & Practice, 2016, 36(6): 1617-1624.

Ye Zimeng, Zhang Dabin. Genetic algorithm for solving TSP based on reverse learning and metropolis criterion[J]. Software Guide, 2023, 22(08): 104-110.

Peng Laihu, Fang Liaoliao, Shen Chunya, et al. Software,2023,44(04):6-10.

Chen Xuewen, Chen Ruihan, Li Zhi, et al. Intelligent scheduling system for the service industry based on adaptive simulated annealing algorithm[J]. Computer Knowledge And Technology,2023,19(35):38-42.

Luo Dong, Guo Xiaoxue. Chaotic bat algorithm based on Cauchy-Gaussian variation[J]. Information And Computer (Theoretical Edition), 2023, 35 (22): 72-74+78.

Zhonghua Han is a professor at Shenyang Jianzhu University in Shenyang, Shenyang, China. He is mainly engaged in production operation management, enterprise automation system integration technology, enterprise scheduling and production scheduling algorithm engineering application.

Zhang Yu is currently pursuing a master's degree in Electrical and Control Engineering at Shenyang Jianzhu University in China. He obtained a bachelor's degree in automation from Shenyang Jianzhu University. His research interests include production optimization and intelligent optimization algorithms.

Yuanyuan Liu is an associate professor at Shenyang Urban Construction University in Shenyang. China. She is mainly engaged in production operation, production -scheduling algorithm engineering application.