

Improved Genetic Algorithm for Solving the Order Scheduling Problem of Structural Parts Workshop

Jiajun Zhang, Yinghua Liao

School of Mechanical Engineering, Sichuan University of Science & Engineering, Yibin Sichuan 644000, China

Abstract: In view of the problem that the workshop of intelligent terminal structural parts is faced with the disturbance caused by urgent orders affecting the production efficiency and delivery punctuality of the workshop, combined with the initial scheme minimization index and the order insertion rescheduling delay minimization index, a mathematical model of order insertion scheduling in the production workshop of intelligent terminal structural parts is established; And genetic algorithm is used to solve it, dynamically adjusting the probability of non-uniform crossover and non-uniform mutation to improve the coverage of the search space and avoid getting stuck in local optimal solutions. The results show that compared with the traditional manual decision-making scheduling, the proposed scheduling scheme improves by 7.67%, 3.4%, 10.89% and 15.5% respectively on different orders. The scheduling scheme can effectively cope with the disturbance impact caused by order insertion and shorten the production cycle.

Keywords: Intelligent terminal structure, Non-uniform crossing, Non-uniform mutation, Genetic algorithms.

1. Introduction

With the rapid development of the demand market for intelligent terminal products, digital information, the Internet, advanced manufacturing, virtual simulation and other technologies are widely used in the field of modern manufacturing. As a key component of the overall consumer market, the NC machining and manufacturing of intelligent terminal product structures, how to combine the above modern technologies to carry out the transformation and upgrading of NC machining and manufacturing process, is an important research topic to improve production efficiency and quickly occupy the market. In the manufacturing industry, the production workshop is the core of enterprise production. Mature production workshops have very high requirements for order planning, space and time, and control of process logic.

The insertion of urgent orders is one of the common disturbances in the production workshop of intelligent terminal structures. The common methods of inserting orders include refund and postponement. In addition, Yan Haoyun [1] and others proposed the method of rescheduling and inserting orders, established the mixed integer programming model of the order rescheduling problem, and used a heuristic solution method based on the idea of dynamic programming to solve it, and verified its superiority. In recent years, more and more scholars have paid attention to the emergency order insertion scheduling problem. Zakaria [2] and others proposed an improved genetic algorithm to adapt to the dynamic insertion of emergency orders by reordering the available idle time of devices and matching the rescheduling operation with the reorganization and non-reorganization strategies. Lichongbo [3] and others studied an energy-saving optimization method for flexible job shop rescheduling considering emergency order insertion and dynamic events of machine tool failure, established an energy-saving Dynamic Rescheduling optimization model, and proposed an energy-saving optimization method for rescheduling based on multi-objective gravitational search algorithm. In addition, gao [4] proposed a two-stage artificial bee colony algorithm to solve

the rescheduling problem of flexible job shop; Subsequently, the above algorithm [5] is further improved in the literature to make it suitable for the rescheduling problem of fuzzy flexible production workshops. A small number of research on dynamic scheduling, such as Zhang Qin, etc., are based on flexible job shop, supported by simulation technology and intelligent optimization algorithm, integrate energy flow and dynamic events into discrete event simulation software, and build an energy-saving dynamic scheduling simulation model to assist the dynamic scheduling decision of the workshop.

It can be seen from the above literature that the current scheduling strategies for emergency order insertion are mostly focused on the overall rescheduling of orders, and the stability and global optimization of scheduling are difficult to guarantee. Although the research on dynamic job shop scheduling has increased, the overall number is still small. Although the static scheduling is relatively stable and easy to implement, it may lead to the failure to obtain the optimal results.

2. Problem Description and Mathematical Model

2.1. Problem Description

According to the demand of order scheduling, it is necessary to describe the scheduling problem of structural parts production workshop in detail. The problem description and constraints are as follows: first, generate the initial scheme for multi order entry in the workshop, set it as s , take the order as the management unit, the workpiece properties, processing technology and processing requirements of the same order are the same, and allocate n workpieces to M equipment for processing under one order. During the execution of the initial scheme s , considering the impact of the abnormal disturbance of order insertion, the physical equipment data are collected in real time during the task execution of the initial scheme, and the results are continuously compared with the estimated calculation results of the virtual data. When there is a large difference, it is determined to be an emergency single order insertion

disturbance, and then the rescheduling is triggered. The rescheduling scheme SR is regenerated based on scheme s , and the delay time r of emergency order insertion is introduced. Table 1 describes the same order parameters in the workshop scheduling process. Relevant decision variables are defined as follows:

Table 1. Symbols and definitions table

Symbols	definitions
I_a	Order Number
n	Number of workpieces included in the order
m	Number of equipment allocated to this order
N_i	The i th workpiece
M_j	The j th device
O_j	Number of processing workpieces allocated to the j th equipment
L	Number of operations of the order piece
L_k	The k th operation of the workpiece
$[0, T]$	Recording period during scheduling
t	Any time belonging to $[0, t]$
D	Order processing cycle
T_{\max}	The time point at which the last workpiece of the order is processed
t_{k2}	Processing duration of the k -th operation of the workpiece
S_{ik}	Time point at which the k -th operation of workpiece I in the initial scheme starts processing
S'_{ik}	Time point at which the k -th operation of workpiece I in the initial scheme is completed
R_{ik}	The time point at which the k -th operation of the inserted workpiece I start processing
R'_{ik}	The time point at which the k -th operation of the inserted workpiece I am completed
S_{Rt}	The time point at which the last workpiece of the rescheduling scheme SR completes processing
X_{ij}	The position of the equipment in the workshop layout matrix $a*b$, where i is the number of rows and j is the number of columns

The problem is subject to the following assumptions and constraints:

- ① All equipment and workpiece materials are available at the initial time;
- ② Each workpiece is allocated to at least one equipment for processing, and the unassigned equipment is set to 0;
- ③ Each equipment can only process one workpiece at a time (when the fixture tray is used to clamp multiple parts for processing at the same time, the tray is regarded as the minimum unit of calculation);
- ④ Each workpiece can only be processed by one set of equipment at the same time, and the current equipment is set to 2 when processing;
- ⑤ Any processing procedure of the workpiece can only be carried out after the previous procedure is completed;
- ⑥ When processing multiple orders in a workshop, the workpiece of the same order can only be processed on the equipment in the area divided by the order;
- ⑦ When processing multiple orders in the workshop, the division areas of each order do not coincide;
- ⑧ After the equipment starts the process processing, the processing cannot be interrupted;
- ⑨ The processing time of each process is known, and the processing time of the same process is fixed;

⑩ The loading and unloading time of the workpiece is known and fixed, and the workpiece cannot be processed during loading and unloading, and the current equipment is set to 1 when loading and unloading;

2.2. Mathematical model

$$\min(\max_{1 \leq i \leq n} (S_{Rt}) + \sum_{i=1}^n \sum_{k=1}^L |R_{ik} - R'_{ik}|) \quad (1)$$

$$X_{ij} \in [0, 1, 2], \quad \forall i \in [1, a], \quad \forall j \in [1, b] \quad (2)$$

$$T + C \leq D, \quad C \in R \quad (3)$$

$$S_{ik} > S'_{ik-1}, \quad S_{ik} + t_{k2} = S'_{ik} \\ \forall k \in [1, L], \quad \forall i \in [1, n] \quad (4)$$

$$x_j + y_j \leq 1, \quad \forall j \in [1, m] \quad (5)$$

$$(S_{ik}, S'_{ik-1}) \cap (S'_{ik}, S_{ik}) = \emptyset \quad (6)$$

$$\sum_{j=1}^m x_{ij} = 1, \quad \forall i \in [1, n] \quad (7)$$

The objective function formula (1) is makespan minimization, and the maximum completion time is expressed as the total time of the initial scheme s ; Constraint formula (2) defines that the equipment set to 0, 1 and 2 in the workshop distribution matrix are represented as idle, loading and unloading, and processing respectively; Constraint formula (3) defines that the execution time of the whole scheduling scheme plus the constant C (the consumption time of warehousing, packaging, packing and transportation after the order processing) must be less than or equal to the order delivery date D ; Constraint formula (4) indicates that the processing of the workpiece must complete the previous process before it can enter the processing of the next process, and the equipment processing cannot be interrupted after the processing starts; Constraint formula (5) indicates that the same equipment cannot process multiple orders at the same time; Constraint formula (6) means that each workpiece can only be processed on one equipment at the same time; Constraint equation (7) defines that when processing multiple orders in the workshop, the selected processing machine and the divided processing area of each order cannot overlap, and one equipment can only be selected by one order.

3. Genetic Algorithm

3.1. Encoding and Decoding

The chromosome of FJSP is composed of operation sequence part and machine allocation part. This coding method can simplify the decoding process and reduce the cost. Figure 1 shows a coding example of an FJSP instance, which represents the solution to the optimization problem. The length of chromosome is twice the number of operations. There are 8 operations in this example, so the length of chromosome is 16. When scanning the operation sequence, the workpiece layer is in the order from left to right. The first "L" refers to the first operation of the job processed on the machine, corresponding to the "3" of the machine allocation part; The second "1" represents the second operation of workpiece 1 to be processed on the machine, corresponding

to "2" in the machine allocation section. According to the above description, the operation sequence of machine M1 is O41→O21→O32, machine M2 is O31→O42→O12, machine M3 is O11→O22. The operation sequence and machine allocation of the solution can be interpreted as: {(O11, M3), (O41, M1), (O31, M2), (O21, M1), (O42, M2), (O32, M1), (O12, M2), (O22, M3)}.

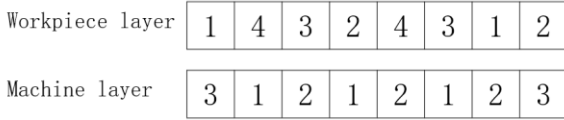


Figure 1. Chromosome coding map

3.2. Fitness function

Fitness function is a method for evaluating individual quality, screening scheme and determining excellent individuals in the group. Usually, the fitness function is achieved by calculating the fitness of an individual, which is based on the reciprocal of the objective function value. In genetic algorithm, fitness function plays an important role, which can quantify the degree of individual adaptation and serve as the basis for selection and evolution. Through the evaluation of fitness function, genetic algorithm can optimize the search process, retain and develop better individuals in the population. The fitness function is:

$$f(x) = \frac{1}{\min(\max_{1 \leq i \leq n} (S_{Rt}) + \sum_{i=1}^n \sum_{k=1}^L |R_{ik} - R'_{ik}|)} \quad (8)$$

3.3. Selection Operation

In order to make the genetic algorithm develop in a better and more advantageous direction at the early stage of iteration, the selection operation is used to select excellent individuals from the population. Roulette selection is a commonly used method for selection based on fitness value. This method determines the selection probability according to the fitness value of the individual. The higher the fitness value, the higher the probability that the individual will be selected. This can ensure that excellent individuals get more opportunities to be selected in the group, which is recorded as P_a . The calculation formula of P_a is:

$$P_a = \frac{f_a}{F} \quad (9)$$

Where f_a is the fitness value of everyone in the population; F is the total fitness of the population.

3.4. Cross Operation

In this paper, uniform crossover is used to generate new individuals. The specific implementation is to randomly select the loci of two individuals and exchange these loci with equal probability, to generate two new individuals. The purpose of this crossover method is to increase the diversity of genetic algorithm and improve the effect of global search. To solve this problem, non-uniform crossover probability is introduced P_c :

$$P_c = P_{c_0} + \gamma \left(1 - \frac{gen}{MAXGEN} \right) \quad (10)$$

Where P_{c_0} is the initial crossover rate; gen refers to the current number of iterations; $MAXGEN$ represents the total number of iterations; γ is a constant of (0, 1).

In genetic algorithm, there are many ways of crossover operation, including uniform crossover, multipoint crossover, local uniform crossover, sequential crossover and periodic crossover. These methods provide a variety of strategies for gene exchange and combination in the evolution process of genetic algorithm. In this paper, the uniform crossover method is adopted, which is to randomly select the loci of two individuals, and exchange these loci with equal probability, to generate two new individuals. Uniform crossover is a common crossover operation mode of genetic algorithm, which can help the algorithm better search the solution space and enhance the global search ability. An example of uniform crossing is shown in Figure 2.

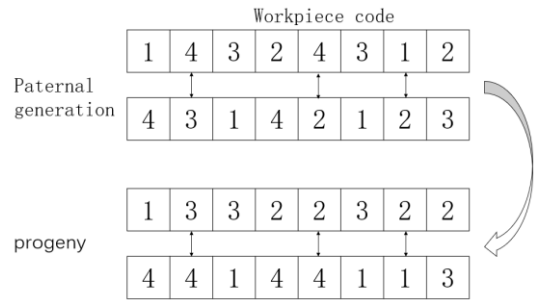


Figure 2. Example diagram of chromosome process code crossing

The selected loci for crossover operation are shaded in Fig. 2 and Fig. 3, and the probability of each locus being selected is random and not certain.

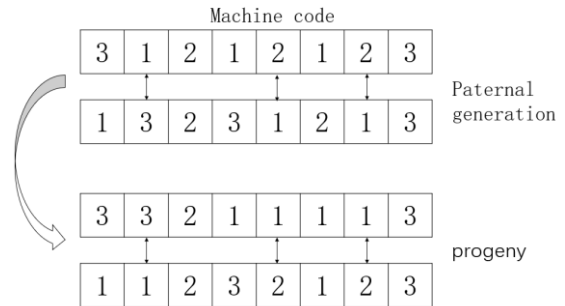


Figure 3. Chromosome machine coding crossover

3.5. variation operation

In the field of traditional genetic algorithm, the common practice is to operate with fixed mutation probability. However, this method may make the algorithm fall into local optimal solution, so it can not find a better solution. In order to solve this problem, the non-uniform mutation rate can be introduced to make the genetic algorithm explore more comprehensively in the search space. Generally, the rate of variation in the initial stage is high, to quickly explore the solution space. As the number of iterations increases, the mutation rate gradually decreases, to search the optimal solution more accurately. The specific nonuniform variation rate can be calculated according to equation (11).

$$P_m = P_{m_0} + \rho \left(1 - \frac{gen}{MAXGEN} \right) \quad (11)$$

Where P_{m_0} is the initial variation rate. The meaning of this function is that as the number of iterations increases, the rate of variation will slowly decrease, and the range of ρ is (0,1), which is used to control the rate of decline of the rate of

variation. When the value of ρ is close to 1, the rate of decline of the rate of variation will be relatively slow; When the value of ρ is close to 0, the rate of variation will decline faster. Using non-uniform mutation makes the mutation rate gradually reduce with the increase of iteration times, to achieve better optimization effect. This paper proposes an improved mutation method of artifact coding, which aims to improve the fitness of offspring and evolve in a better direction, while maintaining population diversity. This method uses mutation operator based on neighborhood search to carry out mutation of workpiece coding. In the machine coding, the single point mutation method is used to replace the selection operation of the machine. The specific operation diagram is shown in Figure 4 and figure 5.

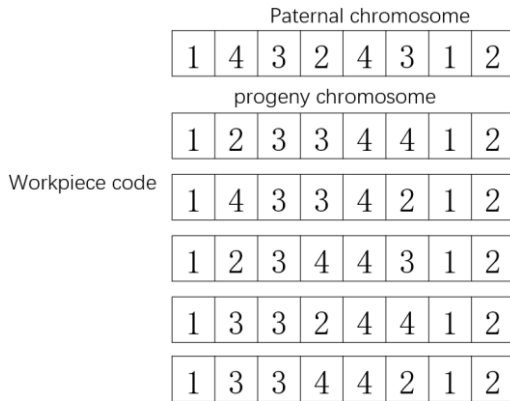


Figure 4. Variation operation based on domain operation in process layer

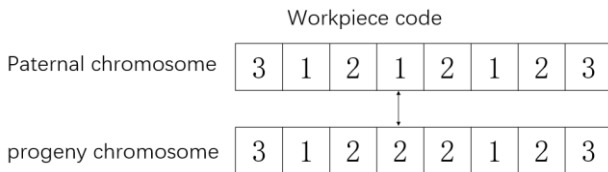


Figure 5. Machine coding single point mutation operation

The specific steps of variable neighborhood mutation operation are as follows:

- (1) Select a certain number of different positions to represent genes of different artifacts. These locations can be randomly selected or selected using other strategies.
- (2) These positions are fully arranged to generate all solutions in the current neighborhood. Different combinations of neighborhood solutions can be obtained by fully arranging these positions.
- (3) Calculate the fitness of each neighborhood solution, and use the fitness function to evaluate according to the specific problem.
- (4) The neighborhood solution with the best fitness is

selected as the current solution. According to the fitness evaluation results, the neighborhood solution with the highest fitness is selected as the current solution.

Through the above steps, the variable neighborhood operation can be realized, and the search space and diversity can be increased, to improve the convergence and search ability of the algorithm.

3.6. Genetic Algorithm Process

The algorithm flow chart is shown in Figure 6, and the specific steps are as follows:

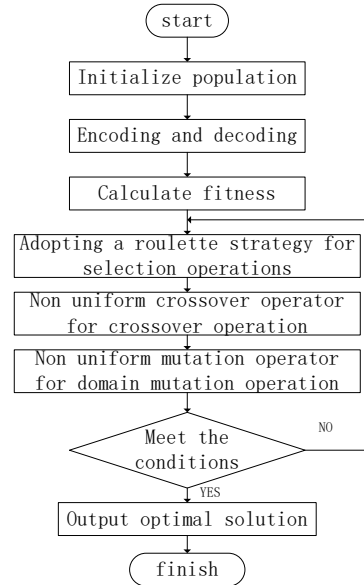


Figure 6. Algorithm flow chart

4. Simulation Results and Analysis

In this experiment, MATLAB data calculation software is used as the data analysis experimental platform to simulate the reactive scheduling scheme. According to the scheduling requirements of a structural parts manufacturing enterprise to undertake orders, the enterprise prescheduled 720 sets of CNC machining machines, and generated the initial scheme based on makespan minimization optimization algorithm. In the complete process from enterprise order acceptance to delivery, the order attribute includes the preparation time of the enterprise for tools, fixtures, fixtures and programming in the early stage, as well as the packaging and shipping time after the completion of the later order. If there are multiple batches of delivery demand orders, the batch delivery in the middle does not affect the order processing, so the statistics are still based on the delivery time of the last batch. The attribute of the sample order in this experiment is shown in Table 2.

Table 2. Order property sheet

Name/Model	Preparation in advance			Initial time (days)	later stage			Later time (days)
	Tool	Programming	Fixture		Clean (times)	Sealing box	Logistics	
A middle frame	√	√	√	15	2	√	Land transportation	4
B middle frame	√	√	√	15	2	√	Land transportation	4
Back cover of phone		√	√	10	1	√	Land transportation	3
Rear lid supplement		√	√	√ 10	1	√	Land transportation	3

D_a is the processing cycle, and d_a is the actual maximum completion time. The specific process attribute table of each

order including processing cycle D , loading and unloading time LD and processing time CNC is shown in Table 3.

Table 3. Process attribute table

Order	Name/Model	Number of workpieces/n	Number of operations/L	Process/Production cycle (s)	Processing cycle D (days)
1	A middle frame	500,000	2	LD1: 25; CNC1: 450 LD2: 25; CNC2: 510	30
2	B middle frame	600,000	2	LD1: 20; CNC1: 300 LD2: 20; CNC2: 300	25
3	Back cover of phone	600,000	2	LD1: 40; CNC1: 380 LD2: 40; CNC2: 600	35
4	Rear lid supplement	700,000	1	LD1: 20; CNC1: 110	14

After analyzing the order and workpiece attributes, the processing completion time T_{ia} a single workpiece and the total processing time T_{Ia} a single order can be calculated by the order quantity I_a , the total number of workpieces n , the number of machine tools m , the number of operations L , the processing time t_{ak2} for each operation. When there are multiple processes in a workpiece processing process, the number of processes is l_a . At this time, the processing completion time of a workpiece is the sum of the loading and unloading time of each process and the processing time of each process, expressed as:

$$T_{ia} = \sum_{k=1}^{L_a} (t_{ak1} + t_{ak2}) \quad (12)$$

The total processing time of this order is:

$$T_{Ia} = \frac{n_a}{m_a} \left[\sum_{k=1}^{L_a} (t_{ak1} + t_{ak2}) \right] \quad (13)$$

The value of T_{Ia} is calculated from equation (13) as the fitness determination value of makespan minimization. According to the processing cycle of each order as the constraint condition of T_{Ia} of the order, when the value of m will cause T_{Ia} to exceed the processing cycle, the value of m is invalid, and the sum of m_a between orders does not exceed the number of prescheduled machine tools of the enterprise. The constraint substituted into equation (14) is expressed as:

$$m_1 + m_2 + m_3 + m_4 \leq 720 \quad (14)$$

In the enterprise order management, to deal with the order delay caused by abnormal interference and accidents, we can adjust the logistics mode, such as changing from land transportation to air transportation, to alleviate the pressure of delay. In the algorithm processing, the planned completion time of the original order is advanced, so as to ensure that the estimated maximum completion days d_a of the initial scheme is controlled within the elastic range of 5 days to more than 2 days in advance (compensated by changing the logistics method), and the minimum value is found, so as to improve the flexibility of the enterprise and the ability to deal with emergencies. Thus, the value range calculation expression of order m is the conversion of equation (15):

$$m_a \in \left\{ \frac{n_a}{D_a - 5} \left[\sum_{k=1}^{L_a} (t_{ak1} + t_{ak2}) \right], \frac{n_a}{D_a + 2} \left[\sum_{k=1}^{L_a} (t_{ak1} + t_{ak2}) \right] \right\} \quad (15)$$

D_a is the processing cycle of order a ; Since m represents the number of processing equipment selected for the order, the calculation result should be rounded to an integer. Through calculation, the value range of m for each order of the sample is: $m1 \in [182, 234]$; $m2 \in [164, 223]$; $m3 \in [198,$

$246]$; $m4 \in [65, 118]$; Therefore, the experimental constraints defined are shown in Table 4.

Table 4. Define constraints

Serial	Number of devices/ m_a	Constraint range	Estimated maximum completion days/ d_a	Constraint range
1	m_1	[182, 234]	d_1	[25, 32]
2	m_2	[164, 223]	d_2	[20, 27]
3	m_3	[198, 246]	d_3	[30, 37]
4	m_4	[65, 118]	d_4	[9, 16]

The generated simulation data of some initial schemes are shown in Table 5.

Table 5. Simulation results of some initial schemes

NO.	$m1_Nu$ m	$m2_Nu$ m	$m3_Nu$ m	$m4_Nu$ m	ma_Su m
1	215	209	233	61	718
2	200	189	229	102	720
3	193	211	199	116	719
4	185	217	243	75	720
5	184	175	204	156	719
6	213	217	231	59	720
7	226	188	228	77	719
8	192	187	201	140	720
9	224	191	225	77	717
10	230	188	201	101	720
11	229	197	198	96	720
12	215	166	232	105	718
13	202	183	205	128	718
14	214	208	210	88	720
15	211	184	236	89	720
16	222	174	212	111	719
17	232	198	215	75	720
18	210	174	226	108	718
19	228	199	202	91	720
20	223	166	213	118	720
21	196	178	240	104	718
22	191	166	222	141	720
23	198	178	203	141	720
24	182	175	209	154	720
25	183	174	210	153	720

The simulation results of T_{Ia} for the maximum completion time of each order part are shown in Table 6.

Table 6. Simulation results of partial completion time

NO.	T ₁₁ _Day	T ₁₂ _Day	T ₁₃ _Day	T ₁₄ _Day	T _{1A} _Day
1	27.19	21.27	31.59	17.27	97.31
2	29.22	23.52	32.14	10.33	95.21
3	30.28	21.06	36.99	9.08	97.42
4	31.59	20.48	30.29	14.04	96.41
5	31.77	25.40	36.08	6.75	100.00
6	27.44	20.48	31.87	17.85	97.64
7	25.86	23.64	32.29	13.68	95.47
8	30.44	23.77	36.62	7.47	98.30
9	26.09	23.27	32.72	13.68	95.76
10	25.41	23.64	36.62	10.43	96.10
11	25.52	22.56	37.18	10.97	96.23
12	27.19	26.77	31.73	10.03	95.72
13	28.94	24.29	35.91	8.23	97.36
14	27.31	21.37	35.05	11.97	95.70
15	27.70	24.15	31.19	11.83	94.88
16	26.33	25.54	34.72	9.49	96.08
17	25.19	22.45	34.24	14.04	95.92
18	27.83	25.54	32.57	9.75	95.70
19	25.64	22.33	36.44	11.57	95.98
20	26.21	26.77	34.56	8.93	96.47
21	29.82	24.97	30.67	10.13	95.59
22	30.60	26.77	33.16	7.47	98.00
23	29.52	24.97	36.08	7.47	98.04
24	32.11	25.40	35.22	6.84	99.57
25	31.94	25.54	35.05	6.88	99.42

Based on the constraint conditions and the initial preset values of the experiment, the optimal results of the solution are shown in Table 7.

Table 7. Statistical table of optimal results

Order	Number of equipment	Proportion of workpieces	Number of remaining workpieces	Maximum completion time/T _{1a} (s)	Maximum completion on days (days)
1	211	2369	+141	2393365	27.7
2	184	3260	+160	2086957	24.15
3	236	2542	+88	2694915	31.19
4	89	7865	+15	1022472	11.83

It can be seen from table 10 that the makespan minimization function value is:

$$f_1 = T_{11} + T_{12} + T_{13} + T_{14} = 27.7 + 24.15 + 31.19 + 11.83 = 94.87 \quad (16)$$

According to the scheduling requirements of a batch of orders of the enterprise, the simulation experiment takes the number of prescheduled NC machining machines as the target, and the optimal initial scheme for selecting machining machines is: M1=211; M2=184 sets; M3=236 sets; M4=89 sets, the total maximum completion time of this batch of orders is 94.87 days. If the maximum T_{1a} value is determined in the parallel processing mode, the completion time of the scheduling scheme of this batch of orders is 31.19

days. Compared with the traditional manual decision-making scheduling scheme, the calculation results of equation (11) show that the production efficiency of each order is improved: E1=7.67%, E2=3.4%, E3=10.89%, E4=15.5%; The elastic redundancy r of each order is r1=4.3 days, r2=2.85 days, r3=5.81 days, r4=4.17 days respectively.

5. Conclusion

In view of the order scheduling problem faced by the intelligent terminal structure workshop, an effective order scheduling scheme is successfully proposed by constructing the mathematical model of the intelligent terminal structure workshop order scheduling and applying the genetic algorithm to solve it. The results show that this scheme has significant advantages in improving production efficiency and shortening production cycle. Compared with the traditional manual decision-making scheduling, it achieves efficiency improvements of 7.67%, 3.4%, 10.89% and 15.5% on different orders, respectively. It not only provides an efficient solution for the order scheduling problem of the intelligent terminal structure workshop, but also provides new ideas and methods for the research of similar scheduling problems.

References

- [1] Yan Haoyun, Li Hongyu. Rush order inserting problem based on load-oriented manufacturing control technique [J]. Computer Integrated Manufacturing Systems, 2009, 15(09):1809-1815. DOI:10.13196/j.cims.2009.09.147.yanhy.025
- [2] Zakaria Z, Petrovic S. Genetic algorithms for match-up rescheduling of the flexible manufacturing systems [J]. Computers & Industrial Engineering, 2012, 62 (2): 670-686
- [3] Gao Kaizhou, Suganthan P N, Tasgetiren M F, et al. Effective ensembles of heuristics for scheduling flexible job shop problem with new job insertion [J]. Computers & Industrial Engineering, 2015, 90 (12): 107-117
- [4] Gao Kaizhou, Suganthan P N, Chua T J, et al. A two-stage artificial bee colony algorithm scheduling flexible job-shop scheduling problem with new job insertion [J]. Expert Systems with Applications, 2015, 42 (21):7652-7663.
- [5] ZHANG M, TAO F, NEE A Y C. Digital twin enhanced dynamic job-shop scheduling [J]. Journal of Manufacturing System, 2021, 58, Part B:145-156.
- [6] CHEN F Y, et al. "Textile Flexible Job-Shop Scheduling Based on a Modified Ant Colony Optimization Algorithm." Applied Sciences 14.10 (2024): 4082.
- [7] WU M L, DONG S Y, and TIAN Y L. "An improved particle swarm algorithm with the elite retain strategy for solving flexible jobshop scheduling problem." Journal of Physics: Conference Series. Vol. 2173. No. 1. IOP Publishing, 2022.
- [8] ZHONG C T, GANG L, ZENG M. "Belugawhaleoptimization: Anovelnature-inspiredmetaheuristicalgorithm." Knowledge-BasedSystems251 (2022):109215.
- [9] JIANG B, et al. "A Review on Intelligent Scheduling and Optimization for Flexible Job Shop." International Journal of Control, Automation and Systems 21.10 (2023): 3127-3150.