


## Article

# A Non-Dominated Genetic Algorithm Based on Decoding Rule of Heat Treatment Equipment Volume and Job Delivery Date

Yan Liang <sup>1,2,\*</sup>  and Qingdong Zhang <sup>1</sup>

<sup>1</sup> School of Mechanical Engineering, University of Science and Technology Beijing, Beijing 100083, China; zhang\_qd@me.ustb.edu.cn

<sup>2</sup> Beijing Xinfeng Aerospace Equipment Co., Ltd., Beijing 100083, China

\* Correspondence: b20160238@xs.ustb.edu.cn

**Abstract:** This paper investigated the flexible job-shop scheduling problem with the heat treatment process. To solve this problem, we built an unified mathematical model of the heat treatment process and machining process. Up to now, this problem has not been investigated much. Based on the features of this problem, we are intended to minimize  $C_{max}$ , maximize the space utilization rate of heat treatment equipment, and minimize the total delay penalty to optimize the scheduling. By taking the dynamic process arrival under consideration, this paper proposed a set of decoding rules based on the heat treatment equipment volume and job delivery date to achieve a hybrid dynamic scheduling solution during one scheduling procedure. When the utilization rate of heat treatment equipment volume is maximized, and the job delivery date is taken under consideration, it is preferred to minimize the number of workpiece batches in the same job, and reduce the waiting time of the pending job. In combination with the improved adaptive non-dominated genetic algorithm, we worked out the solution. Furthermore, we verified the effectiveness of the proposed decoding rules and improved algorithm through algorithm comparison and calculation results. Finally, a software system for algorithm verification and algorithm comparison was developed to verify the validity of our proposed algorithm.

**Keywords:** heat treatment process; decoding based on the equipment volume and delivery date; non-dominant genetic algorithm; the utilization rate of heat treatment equipment volume



**Citation:** Liang, Y.; Zhang, Q. A Non-Dominated Genetic Algorithm Based on Decoding Rule of Heat Treatment Equipment Volume and Job Delivery Date. *Algorithms* **2021**, *14*, 310. <https://doi.org/10.3390/a14110310>

Academic Editor: Antonio Della Cioppa

Received: 24 September 2021  
Accepted: 18 October 2021  
Published: 25 October 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



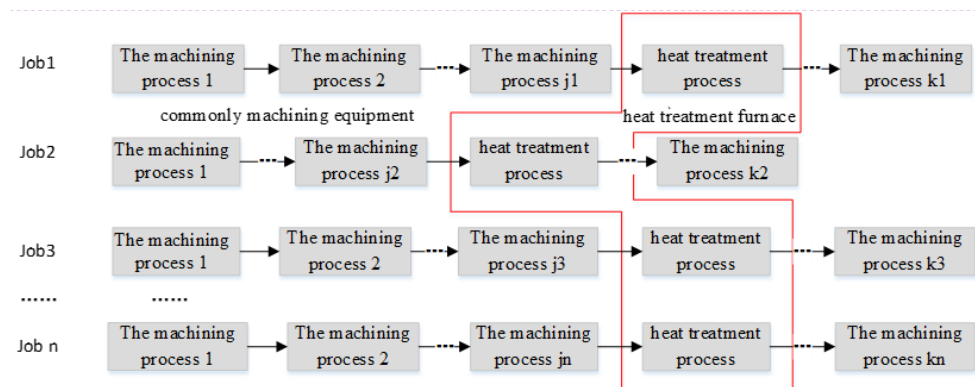
**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Flexible Job-shop Scheduling Problem (FJSP) is always deemed as the scheduling problem closest to actual production by a majority of scholars and enterprises, primarily used to solve the scheduling problem that only one workpiece can be processed by a single equipment such as machining in the workshop at a certain point of time. Nevertheless, in actual production process, mechanical parts not only need producing through machining and other processes, but batch heat treatment process is also involved, as shown in Figure 1. In view of the above fact, traditional job plan scheduling often only takes machining process into account, regards heat treatment process as outsourcing, only gives it adequate waiting time, and believes this heat treatment process can be replaced. Apparently, the method of replacing heat treatment process with adequate pure waiting time fails to present more accurate heat treatment plan, accurate heat treatment completion time of the work piece is unavailable through subsequent machining process, and the job plan ends up lacking instructive significance.

The difference between heat treatment process of mechanical workpiece and other machining process lies in that machining process is characterized by one machining equipment only processing one workpiece at a certain point of time, but heat treatment process is characterized by one heat treatment equipment processing a batch of workpieces simultaneously at a certain point of time, this batch of workpieces have the same heat treatment conditions, and the total volume of this batch of workpieces does not exceed maximum

capacity of this heat treatment equipment. The heat treatment process method considerably differs from batch process of the last testing or cleaning process of semiconductor parts, the batch process of semiconductor parts belongs to the last process, and any type of parts can be put into the batch processing equipment for processing simultaneously, whereas mechanical heat treatment process is an intermediate part, followed by other processes afterwards, and only workpieces with the same or similar heat treatment process can be combined and put into the same heat treatment equipment for processing.



**Figure 1.** Processing Route for Mechanical Parts with Heat Treatment Process.

In response to the above flexible job-shop scheduling problem with heat treatment process, this article aims to solve the needs of arranging workpieces on processable equipment in a reasonable order, and rationally combine the workpieces to process on heat treatment equipment in a reasonable order, so as to minimize the maximum makespan of workpiece and maximize the utilization rate of heat treatment equipment, while meeting the delivery date of workpiece.

## 2. Literature Review

Classic flexible job-shop scheduling problem (FJSP) has been very compliant with actual production workshop, and extensively and deeply studied, but in semiconductor industry, ceramic manufacturing and other fields, flexible job workshop cannot be accurately described because they have batch processing part. Under this, batch scheduling has gradually caught people's eyes, under extensive research of people.

Early scholars mostly solved the batch scheduling problem through heuristic rule. Ikura Y et al. (1986) considered two problems when solving single batch processing machine, proposed a scheduling rule based on First-Only-Empty with time complexity of  $O(n)$  for the job scheduling problem without deadline, acquired minimum makespan, and obtained optimal solution; for the scheduling problem with the same job scale and deadline, considered workpiece assignment date and deadline, proposed a scheduling rule based on Greedy-Adjusted-Bunching with time complexity of  $O(n^2)$ , gained high-quality scheduling scheme, and proved high efficiency of both rules [1]. Chandru V et al. (1993) adopted an optimal branch and bound algorithm to accurately solve single machine and parallel machine scheduling problem against the research background of the last test operation in semiconductor manufacturing, and proved the proposed heuristic algorithm through a large number of experiments, which can obtain better solution within a reasonable cpu time [2]. On the context of semiconductor-based test aging operation, Uzsoy, R (1994) considered different production scales when solving batch scheduling problem, indicating such problems were NP-hard problems. He proposed two heuristic rules, and combined with branch and bound algorithm to show the ability of obtaining a better solution in conjunction with experiments [3]. When studying the same constraint problem, Dupont, L (1998) took minimized average flow time as the optimization target, and proposed a new DYNA heuristic rule method that if the workpiece size was between 1 and 10, there were significant advantages, but the advantages were not obvious for large-scale problem

scheduling results [4]. Azizoglu, M (2000) considered different job processing times and different job scales when studying batch scheduling problems, and proposed more effective branch and bound algorithm, to prove through experiments that returns optimal solutions for problem instances with up to 25 jobs within 30 min of CPU time [5]. Chang and Wang et al. (2004) studied batch scheduling problems of different job scales, while considering different arrival time  $\alpha$  and different processing time  $\beta$  at the same time, proposed a heuristic algorithm, with minimum makespan as the optimization target, and concluded through experimental results that when the ratio of  $\alpha$  to  $\beta$  was greater than 1, it had certain instructive significance [6].

As metaheuristic algorithm such as genetic algorithm (GA) and simulated annealing algorithm (SA) were emerged and widely applied in research, many scholars used them in the field of batch scheduling, and verified a well-designed metaheuristic algorithm could lead to a solution better than heuristic algorithm within a reasonable time through a large number of experiments. Melouk S et al. (2004) used the simulated annealing algorithm to solve the scheduling problem of single batch processing machine. Compared with running results of CPLEX, it featured faster running time and better scheduling solution, especially manifested in the most significantly shortened makespan under 50–100 jobs [7]. Damodaran and Manjeshwar et al. (2006) solved batch scheduling problem of different job scales by using genetic algorithm, and compared with previous SA, GA can get better solution in a shorter time, with better robustness [8]. Damodaran et al. (2007) proceeded from the improvement of equipment utilization rate, adopted simulated annealing algorithm to solve batch scheduling problem of different job sizes, and proved that the simulated annealing algorithm had faster processing time, by comparing with CPLEX [9]. Manjeshwar and Damodaran et al. (2009) proposed a heuristic algorithm based on Johnson's algorithm and simulated annealing algorithm, while studying flow shop scheduling problem with two batch processors, verified the effectiveness of algorithm through random experiments, and applied such algorithm in electronic manufacturing companies, thus increasing maximum makespan by 20–25% on average through actual running verification, delivering demonstrative application to actual enterprises [10]. When studying a two-stage flexible flow shop problem where parallel batch processing equipment existed at each stage, Gerstl and Mosheiov (2014) [11] proposed a polynomial time dynamic programming algorithm, considered the optimization target as makespan minimization and flowtime minimization, but without sufficient instances for verification. Xu et al. (2012) considered job release time, processing time and job size scheduling issues in batch scheduling, and verified by introducing waste and idle definitions, using ant colony algorithm, combining with the proposed candidate list strategy and heuristic information construction method, comparing GA algorithm experiment that the ant colony algorithm in combination with candidate list had better robustness, and always better than GA algorithm, which was more obvious in large-scale job instances [12].

When studying the scheduling problem of single batch processing machine with different job sizes, Jia and Leung (2014) took wasted space minimization as the optimization target, max-min ant system algorithm, combined with candidate set strategy and wasted space-based heuristic pheromone update, introduced specific local search method to obtain better algorithm performance, and verified superiority of algorithm through a large number of experiments [13]. When studying batch scheduling problem of flexible flow shop, Zeng Z et al. (2018) took the maximum makespan, power consumption and material loss model as the optimization target, established multi-objective optimization model, adopted hybrid non-dominated genetic algorithm II (NSGA -II) to solve it, and took the actual paper mill scheduling problem as an example to verify the effectiveness of this method by comparing manual scheduling [14]. Muter B (2020) studied scheduling problem of single and parallel batch processing machines, proposed a reconstruction method based on Dantzig-Wolfe two-layer decomposition parallel batch processing machine, and gave precise algorithm, so that strict upper and lower limits of parallel problem were available by using the single processing machine, providing insights for the solutions of follow-up

researches [15]. Through regarding the waiting time of the workpiece as the scheduling target, considering the dynamic arrival time, and ensuring the delivery time, Shen Fengping et al. (2020) figured out this problem with the heuristic algorithm for the batch process in combination with the particle swarm algorithm. The effectiveness of the algorithm was verified by simulation [16]. However, they only considered the mechanical batching of workpieces to be processed in batch limited by the batch machine capacity, but failed to consider the combination of this workpiece batch with the subsequent workpiece batch with the similar process, which makes the space utilization rate of the batch machine small. To minimize the maximum makespan, Liu Rong et al. (2020) figured out this problem with the integrated block decoding rules and improved genetic algorithm, and verified the ability of the cluster selection strategy jumping out of the local optimal solution and the feasibility of the algorithm through simulation [17]. However, they failed to consider the space utilization rate of the batch machine, and the blind batching of workpieces with the urgent delivery time may lead to the delay.

In summary, batch scheduling problem has attracted the attention of scholars at home and abroad, and considerable results have been achieved, but the investigated problems are much simpler than those tackled by this paper. This paper tackled the FJSP of the heat treatment process and machining process, which is closer to that occurred in the actual workshop.

The research background of this paper is aerospace pump valve parts. Their production process involves frequent heat treatment, and the heat treatment workshop belongs to another independent production department. Because this workshop undertakes the heat treatment job of multiple workshops in the whole plant, and can organize internal tasks of the heat treatment process by itself, it is very difficult to guarantee the delivery time of a single product. Although there are many types and large quantities of aerospace pump valve parts, their material, size and thickness are similar to each other, and these types have the potential for the batch heat treatment. Changing the current scattered delivery time for the heat treatment leads to unnecessary waiting and blind batching. Therefore, it is essential to improve the orientation of the quantity of the part production workshop-manufactured parts delivered to the heat treatment workshop to satisfy the heat treatment furnace-enabling conditions. This method is an effective way to address this problem in this paper.

To solve this problem through optimizing several parameters, such as minimizing the total job delay penalty, minimizing the maximum workpiece makespan and minimizing the utilization rate of the heat treatment equipment volume, this paper proposed an improved non-dominated genetic algorithm, and two decoding methods for the machining process and batch process with the batch processing step as the node based on the process encoding method. At the early stage of population initialization, we took the batch processing as a node, and adopted the heuristic rule, that is, the smaller relaxation time the process adopts before the batch process, the longer the heat treatment time is, and the higher priority the workpiece combined with the previous batch has. While the process after the batch process adopted the heuristic rule to perform initialization, that is, the shorter the relaxation time is, the higher priority the workpiece has. Meanwhile, we adopted the adaptive crossover and mutation operator to avoid prematurity of the algorithm, theoretically promoting continuous evolution of the algorithm, and thus achieving the global search.

### 3. Problem Description & Mathematical Modeling

#### 3.1. Problem Description

The flexible job-shop scheduling problem with heat treatment process studied in this paper is an extension of traditional flexible job-shop scheduling problem. This problem involves  $n$  jobs to be processed, each job  $J_j$  ( $j = 1, 2, \dots, n$ ) has different delivery dates  $D_j$ , the quantity and size are  $N_j$  and  $S_j$  respectively, each workpiece has machining process and a heat treatment process, and heat treatment process can be an intermediate processing process. With  $m-1$  machining equipment and 1 heat treatment equipment of limited capacity,

each machining equipment can only process one workpiece simultaneously, and each heat treatment equipment can process multiple workpieces at the same time on the premise of unexceeding its capacity. Each job is known to be able to combine with other jobs of heat treatment simultaneously. The difficulty of this research problem lies in rational arrangement of job processing sequence and equipment selection under the premise of limited resource scheduling, combined batch processing in heat treatment part, as well as minimization of maximum makespan of workpiece and utilization rate of heat treatment equipment volume, while meeting job delivery date.

In the problem of this article, the following assumptions need consideration:

- (1) The process is not performed by batches in the machining part;
- (2) Total volume of each batch of job is less than maximum volume of heat treatment equipment;
- (3) All jobs can start processing at time 0;
- (4) All equipment can be processed at time 0;
- (5) The processing date system of all equipment is consistent;
- (6) The circulation time between adjacent processes is not considered;
- (7) The processing time of each equipment in machining process is unchanged;
- (8) After the previous process of heat treatment process is completed, uniformly move for heat treatment processing;
- (9) Once heat treatment starts, it cannot be interrupted, and other jobs cannot be added before the treatment is completed;
- (10) All workpieces in the same job must complete heat treatment, before uniformly move for the next process.

### 3.2. Mathematical Modeling

Parameters:

$J_j$ : Job  $j$ ,  $j = 1, 2, \dots, n$ , of which  $n$  indicates the quantity of Jobs;

$pc_j$ :  $J_j$  delay penalty coefficient;

$N_j$ : size of  $J_j$ ,  $j = 1, 2, \dots, n$ ;

$D_j$ : Delivery date of  $J_j$ ,  $j = 1, 2, \dots, n$ ;

$s_j$ : Single piece volume of  $J_j$ ,  $j = 1, 2, \dots, n$ ;

$O_{ji}$ :  $i$ th process of  $J_j$ ,  $i = 1, 2, 3, \dots, G_j$ ,  $G_j$  is the number of process for  $J_j$ ;

$M_p$ : Equipment  $p$ ,  $p = 1, 2, 3, \dots, m$ , of which  $m$  indicates the quantity of equipment;

$$B_p = \begin{cases} 1, & \text{if } M_p \text{ is Heat Treatment Machine;} \\ 0, & \text{Otherwise;} \end{cases};$$

$T_{ji}$ : Theoretical single piece processing time of  $O_{ji}$  (machining process);

$HT_j$ : Theoretical heat processing time of  $J_j$ ;

$$X_{jip} = \begin{cases} 1, & \text{if } O_{ji} \text{ can be processed on } M_p \\ 0, & \text{Otherwise} \end{cases};$$

$$K_{ji} = \begin{cases} 1, & \text{if } O_{ji} \text{ is next process of the} \\ & \text{heat treatment process} \\ 0, & \text{Otherwise} \end{cases};$$

$$R_{jk} = \begin{cases} 1, & \text{if } J_j \text{ and } J_k \text{ can be combined for} \\ & \text{processing at the same time} \\ 0, & \text{Otherwise} \end{cases};$$

$C$ : Theoretical volume of heat treatment equipment;

Decision variables:

$S_{ji}$ :  $O_{ji}$  processing start time (machining process);

$E_{ji}$ :  $O_{ji}$  processing end time (machining process);

$PT_{ji}$ : Actual processing time of  $O_{ji}$  (machining process);

$E_j$ :  $J_j$  completion time;

$$ID_j = \begin{cases} 1, & \text{if } E_j > D_j, \\ 0, & \text{Otherwise} \end{cases};$$

$TD_j$ :  $J_j$  delay penalty;  $B_q$ : batches  $q, q = 1, 2, \dots, n, q$  is the number of heat treatment equipment processing batches;

$$Y_{jq} = \begin{cases} 1, & \text{if } J_j \text{ exist of } B_q \\ 0, & \text{Otherwise} \end{cases};$$

$N_{jq}$ : the count of  $Y_{jq}$ ;

$$V_{jip} = \begin{cases} 1, & \text{if } O_{ji} \text{ is processed by } M_p, \\ 0, & \text{Otherwise} \end{cases};$$

$C_q$ : The actual occupied volume of the  $q$ th heat treatment;

$S_q$ : The actual starttime of the  $q$ th heat treatment;

$T_q$ : The actual process time of the  $q$ th heat treatment;

$E_q$ : The actual endtime of the  $q$ th heat treatment;

$$U_{jip-j'ip} = \begin{cases} 1, & \text{if } O_{ji} \text{ is the next process of} \\ & O_{j'ip} \text{ processed on } M_p \\ 0, & \text{Otherwise} \end{cases};$$

The flexible job-shop scheduling problem with heat treatment process studied herein has three scheduling targets, namely minimizing maximum makespan of workpiece (Equations (1) and (4)), maximizing average utilization rate of heat treatment equipment volume (Equation (2)) and minimizing delay penalty of workpiece (Equation (3)).

$$\min : C_{max} = \max(E_j), \forall j \tag{1}$$

$$\max : \eta_h = \frac{1}{q} \cdot \sum_{q=1}^q \frac{C_q}{C}, \forall q \tag{2}$$

$$\min : \sum_{j=1}^n (E_j - D_j) \cdot ID_j \cdot pc_j \tag{3}$$

$$E_j = \max(E_{ji}), \forall j, i \tag{4}$$

$$S_{ji} \geq 0, \forall j, i \tag{5}$$

$$PT_{ji} = T_{ji} \cdot N_j, \forall j, i \tag{6}$$

$$E_{ji} = S_{ji} + PT_{ji}, \forall j, i \tag{7}$$

$$\sum_{p=1}^m X_{jip} > 0, \forall j, i, p \tag{8}$$

$$(S_{ji} - E_{uv}) \cdot (S_{uv} - E_{ji}) \cdot V_{jip} \cdot V_{uop} \leq 0, \forall j, i, u, v \tag{9}$$

$$S_{ji} + (1 - U_{jip-j'ip}) \cdot L \geq S_{j'ip} + PT_{j'ip} \tag{10}$$

$$\sum_{p=1}^m B_p = 1, \forall p \tag{11}$$

$$E_q = S_q + T_q, \forall q \tag{12}$$

$$R_{jk} \cdot \frac{HT_j}{HT_k} = R_{jk} \tag{13}$$

$$\sum_{q=1}^B N_{jq} = N_j, \forall j \tag{14}$$

$$E_{ji} \leq S_{j(i+1)}, \forall j, i \tag{15}$$

$$\min(Y_{jq} \cdot S_q) \geq \max(E_{ji} \cdot (1 - K_{ji})), \forall j, i, q \tag{16}$$

$$\min(S_{ji} \cdot K_{ji}) \geq \max(Y_{jq} \cdot E_q) \quad (17)$$

Equations (5)–(7) indicate the relationship between the end time, actual processing time and start time of a process. Equation (8) indicates that each processing process should have at least one piece of processable equipment. Equations (9) and (10) indicate that the same equipment can only execute one process at most at any time. Equation (11) indicates that only one piece of heat treatment equipment can be available. Equation (12) indicates the relationship among the start time, end time and heat treatment processing time of each batch on the heat treatment equipment. Equation (13) indicates that only those workpieces with the same heat treatment processing time can enter the same batch. Equation (14) indicates that if the same job is divided into multiple batches for processing, the total quantity of workpieces in all the batches belonging to this job is equal to the total quantity of this job. Equation (15) indicates that each process can start only after the previous process is completed. Equation (16) indicates that for the workpieces in the same batch for the heat treatment, only after the previous process of this batch is completed, these workpieces in this batch can be processed with the heat treatment. Equation (17) indicates that after all workpieces of the same job have been processed with the heat treatment, the next process of this job can start.

#### 4. Improved Non-Dominated Sorting Genetic Algorithm

It can be seen from the above model that it is a non-linear mathematical model with multiple optimization objectives. Generally, this problem can be solved with traditional methods, such as the linear weighting method and efficacy coefficient method, but the corresponding precise weight cannot be provided. Additionally, a large number of experiments are required to further determine its scope, and its subjectivity is predominated. With the increase of the job scale, these methods cannot solve the subsequent resulting problems. This paper adopted the Non-dominated Sorting Genetic Algorithm (NSGA-II), which is featured with the powerful parallel search capability, and can get multiple Pareto optimal solutions in one run. So, it is extremely suitable for solving the multi-objective FJSP with the heat treatment process herein. Through introducing the new dominated sorting method, and adopting the congestion degree and elite retention strategy, the NSGA-II algorithm has overcome obvious shortcomings of the original NSGA algorithm, such as the high time complexity and short of the elite retention strategy. As a result, it has been applied to numerous industries with excellent results.

Compared with the traditional NSGA-II algorithm, the improved NSGA designed herein has the following features: (1) Designed a kind of process-based one-layer encoding with the heat treatment process as a node to execute initialization according to different urgency degrees before and after the heat treatment process during the population initialization; (2) Performed decoding according to the production scheduling rule of the equipment utilization rate to maximize the equipment utilization rate in the specific process order; (3) Abandoned the fixed crossover and mutation probability of the traditional NSGA-II in the crossover and mutation operation, and adopted the adaptive crossover and mutation probability, which can integrate the global search capability of the algorithm with its convergence. The flowchart based on the improved NSGA is shown in Figure 2.

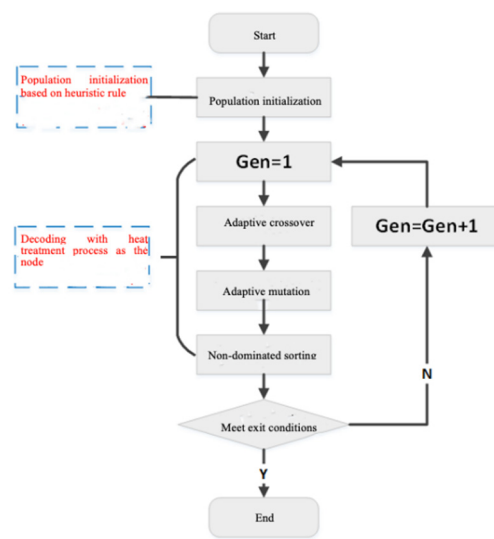


Figure 2. Improved NSGA.

4.1. Encoding Mechanism of the Process with the Heat Treatment Process as a Node

This paper adopts a process-based encoding method with the heat treatment process as a node, which can number all processes of each job, and especially mark the heat treatment process. In this way, the machining processes before and after the heat treatment process can be quickly distinguished, and further compiled according to the number of its occurrences in the chromosome. Figure 3 shows a chromosome containing 7 genes, of which the gene marked with ‘H’ is the heat treatment process. It can be seen from this figure that the heat treatment process of Job 1 is located at the 5th position, the heat treatment process of Job 2 is located at the 7th position, and the heat treatment process of Job 3 is located at the 8th position. It can be seen from this chromosome that the organization sequence is as follows: Machining process 1 of Job 1 → Machining process 1 of Job 3 → Machining process 2 of Job 3 → Machining process 1 of Job 2 → heat treatment process of job 1 → Heat treatment process of Job 3 → Heat treatment process of Job 2 → Machining process 2 of Job 2 → Machining process 2 of Job 1.



Figure 3. Process Encoding with the Heat Treatment as a Node.

4.2. Decoding Rules Based on the Equipment Utilization Rate

With regards to the said encoding rule, we adopted the decoding rule based on the equipment utilization rate, and still took the heat treatment process as a node. The machining process before and after the heat treatment process follows the decoding rule based on the bottleneck equipment, while the heat treatment process follows the decoding rule based on the utilization rate of the heat treatment equipment. In this section, let’s take the scheduling of five jobs (including the heat treatment process) on four pieces of machining equipment and one piece of heat treatment equipment as an example, and their settings are shown in Tables 1 and 2.



**Table 1.** Processing scheduling of 5 jobs on 4 equipment and 1 heat treatment equipment.

Workpiece	Batch	Process	Single Piece Processing Time/h	Whether Heat Treatment	Processable Equipment
Job1	12	O <sub>11</sub>	3	N	M1/M3
		O <sub>12</sub>	2	N	M2/M3/M4
		O <sub>13</sub>	5	Y	B1
		O <sub>14</sub>	3	N	M1/M4
Job2	16	O <sub>21</sub>	4	N	M1/M2
		O <sub>22</sub>	2	N	M2/M4
		O <sub>23</sub>	5	Y	B1
		O <sub>24</sub>	3	N	M1/M3/M4
Job3	15	O <sub>31</sub>	2	N	M2/M3
		O <sub>32</sub>	4	N	M1/M4
		O <sub>33</sub>	6	Y	B1
		O <sub>34</sub>	3	N	M1/M2
Job4	14	O <sub>41</sub>	2	N	M1/M4
		O <sub>42</sub>	3	N	M1/M2
		O <sub>43</sub>	3	N	M1/M3/M4
		O <sub>44</sub>	5	Y	B1
		O <sub>45</sub>	2	N	M1/M3
Job5	17	O <sub>51</sub>	3	N	M1/M2
		O <sub>52</sub>	2	N	M2/M3/M4
		O <sub>53</sub>	6	Y	B1
		O <sub>54</sub>	2	N	M1/M4

**Table 2.** Basic data of the decoding rule example.

Job	Workpiece Quantity in Job	Single Workpiece Volume	Heat Treatment Time	Delivery Date	Jobs to Form Batches
Job1	12	2	5	150	Job2; Job4
Job2	16	3	5	200	Job1; Job4
Job3	15	3	6	230	Job5
Job4	14	2.5	5	185	Job1; Job2
Job5	17	2	6	160	Job3

1. Decoding rules for machining processes before and after the heat treatment process

The machining process follows the decoding rules based on the bottleneck equipment to guarantee full utilization of equipment resources, reduce uneven distribution of bottleneck equipment, improve the coordination of the scheduling solution in resources, and maximize the equipment utilization rate. Take the chromosome [2, 1, 1, 3, 5, 4, 4, 4, 2, 3, H1, H3, H2, H4, H5, 4, 5, 2, 1, 3] as an example to perform decoding based on the equipment utilization rate of the machining process with the procedure as follows:

Step 1.1: Find the process corresponding to the next gene in the chromosome without equipment, and judge whether it is a machining process. If it is not, implement Step 1.4. If it is, judge whether it is a process before the heat treatment process, and if it is, implement Step 1.2, otherwise implement Step 1.5;

Step 1.2: Traverse the set of optional machining equipment for this process, and calculate the bottleneck degree  $BN_p$  of each optional equipment  $M_p$  according to Equation (18). The larger the  $BN_p$  value is, the more the processes are on this equipment. So this equipment is the bottleneck equipment.

$$BN_p = LE_p + \sum_{j=1}^n \sum_{i=1}^{G_j} (T_{ji} \cdot N_j) \cdot X_{jip} \tag{18}$$

Among them,  $LE_p$  indicates the end time of the last process on the equipment  $M_p$  at the current time node.

- Step 1.3: Choose the equipment with the smallest equipment bottleneck degree from the set of optional equipment for this process, and implement Step 1.4;
- Step 1.4: Judge whether there is any unscheduled process in the current chromosome. If there is, implement Step 1.1; otherwise, stop decoding;
- Step 1.5: If the process is the machining process after the corresponding heat treatment process, judge whether the heat treatment process of the job corresponding to this process is arranged. If so, implement Step 1.2, otherwise the implement decoding rule 2.1 of the heat treatment process;
2. Decoding rule of the heat treatment process

During the production scheduling process of the heat treatment, to maximize the utilization rate of the heat treatment equipment volume, (1) it is required to batch the workpieces of the same job, and combine them with the workpieces of other jobs. However, the larger the number of batches in the same job, the smaller the number of workpieces in each batch. If the same job is distributed in a scattered manner, there is a time interval for heat treatment operation between different small batches. The batch that has completed heat treatment processing prior to the last batch has to wait in the waiting area for the last batch of this job to complete heat treatment processing before they are uniformly delivered to the next machining equipment for processing, which is not in favor of minimizing the maximum makespan of the job; (2) Meanwhile, because the job is dynamically delivered to the heat treatment process, it is unknown when the job that can be batched with the pending job will arrive. If it is only considered to maximize the heat treatment equipment volume, the job waiting time must be increased, and thus the job delivery date cannot be satisfied as required.

Therefore, we designed a decoding rule based on the heat treatment equipment volume and job delivery date herein. When maximizing the utilization rate of the heat treatment equipment volume in combination of considering the job delivery date, we tried to reduce the batching quantity of workpieces in the same job and waiting time of pending jobs. This decoding rule introduces a combined batch processing acceptance factor  $\delta$ , which balances the effects of both the delay rate  $td$  of the job delivery date and the increase rate  $\Delta_\eta$  of the utilization rate  $\eta$  of the heat treatment equipment volume in an all-round manner. The risk factor of the combined batch processing at the current time is  $\delta_t = \frac{td_t}{\Delta_{\eta_t}}$ . The smaller the delay rate of the current job delivery date, the larger the utilization rate of the heat treatment equipment volume, and the smaller the risk probability of combined pending batch processing. Among them, the increase rate of the utilization rate of the heat treatment equipment volume is  $\Delta_\eta = \eta - \eta'$ , where  $\eta'$  is the utilization rate of the previous equipment volume.

The decoding rule based on the heat treatment equipment volume and job delivery date designed herein can be executed as follows:

- Step 2.1: Initialize the remaining volume  $C' = C$  of the current heat treatment equipment;
- Step 2.2: Judge whether this process is the heat treatment process, if it is, obtain the job  $J_j$  corresponding to the current process, its unit volume  $S_j$ , job quantity  $N$ , and quantity  $N'$  of workpieces without the combined batch, and then implement Step 2.3. Otherwise implement Step 1.2;
- Step 2.3:  $J_j$  can be combined with a batch  $C^h$  in the uncompleted batch set, and implement Step 2.4; otherwise reset  $C^{h+1} = C - N' \times S_j$  to generate a new batch  $C^{h+1}$ , and then implement Step 2.8;
- Step 2.4: If the remaining volume of the heat treatment equipment of the current batch is  $C^{h'} \geq N' \times S_j$ , update  $C^{h'} = C^{h'} - N' \times S_j$ , and the current job batch grouping is completed. The start and end time of the job  $J_j$ 's heat treatment process can be obtained according to the heat treatment node and time of this batch, and implement Step 2.5; otherwise, implement Step 2.7;
- Step 2.5: If  $C^{h'} = 0$ , complete this batch, update the start and end time of all jobs in this batch, update  $C^{h+1} = C$ , and implement Step 1.2; otherwise, implement Step 2.7;

- Step 2.6: If there is no job in the gene behind the current gene that can be batched with  $J_j$  in code, update  $C^{h+1'} = C$  to complete this batch, and implement Step 1.2; otherwise, calculate the acceptance probability of the first heat treatment process  $J_k$  behind  $J_j$  in the code by job  $J_j$ . Among them, when the non-heat treatment process between two genes is simulated as the pure theoretical processing work hours, the current moment  $t$  is the earliest start time of  $J_k$ , and calculate  $\delta_t = \frac{td_t}{\eta_t}$ . If  $\delta_t \leq 0.865$ , choose Wait, and implement Step 1.2; otherwise, update  $C^{h+1'} = C$  to complete this batch, and implement Step 1.2;
- Step 2.7: If there is no job in the gene behind the current gene that can be batched with  $J_j$  in code, and  $N' \times S_j > C$ ,  $N' = N' - C'/S_j$ , complete this batch, reset and proceed to the next batch, and update  $N' = N' - C'/S_j$ . You can obtain the start time of the new batch according to the completion time of the previous batch, and obtain the start and end time of the heat treatment process of job  $J_j$ , and implement Step 1.2; otherwise, implement Step 2.8;
- Step 2.8: Calculate the acceptance probability of the first heat treatment process  $J_k$  behind  $J_j$  in code by job  $J_j$ . Among them, when the non-heat treatment process between two genes is simulated as the pure theoretical processing work hours, the current moment  $t$  is the earliest start time of  $J_k$ , and calculate  $\delta_t = \frac{\eta_t}{td_t}$ . If  $\delta_t \geq 0.865$ , choose Waite, and  $N' = N' - C^h/S_j$  to complete this batch. Reset and proceed to the next batch, update  $C^{h+1'} = C - N' \times S_j$ . You can obtain the start time of the new batch according to the completion time of the previous batch, and obtain the start and end time of the heat treatment process of job  $J_j$ , and implement Step 1.2; otherwise, complete this batch, reset and proceed to the next batch, and update  $C^{h+1'} = C - N' \times S_j$ . You can obtain the start time of the new batch according to the completion time of the previous batch, and obtain the start and end time of the heat treatment process of job  $J_j$ , and implement Step 1.2;

This section takes the heat treatment process of five jobs as an example to decode the above case according to the above steps. Tables 1 and 2 show the basic data (of which the volume of the heat treatment equipment is 40), and Figure 4 shows the arrangement results of the heat treatment process.

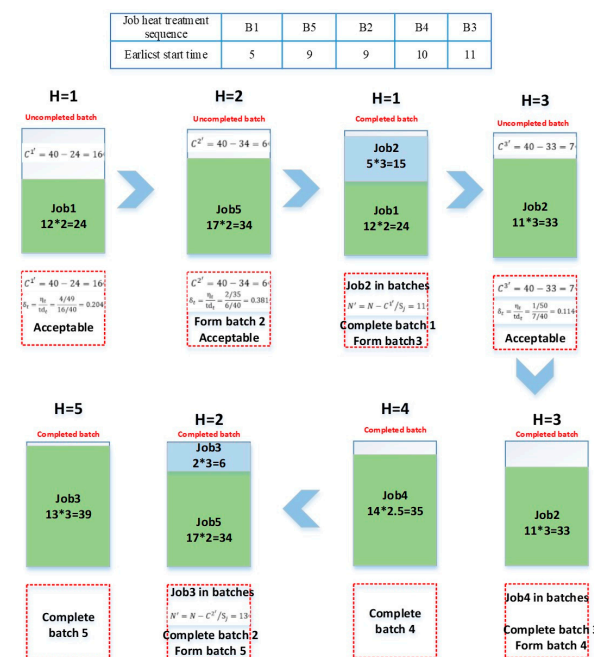


Figure 4. Heat Treatment Decoding Process Instance.

### 4.3. Population Initialization

To acquire accurate solutions, the population initialization method herein is not generated randomly, but the initial solution is generated by following the heuristic rule based on the urgency degree defined herein. Compared with the initial feasible solution generated from an entirely random number, this method can lead to an ideal scheduling plan at the early stage of evolution. The urgency degree proposed in this article is defined as follows. With the heat treatment process as the node, distinguish the machining processes before and after the heat treatment process from each other. The urgency degree of the machining process before the heat treatment process is affected by the possibility of batch grouping of the heat treatment and the relaxation time of the job where this process belongs, and the urgency degree of the machining process after heat treatment process is only affected by relaxation time of the job where the process belongs. The earlier the job process that can be batched is selected, and the earlier the process with the small relaxation time of the job where the process belongs is selected. Initialization is conducted accordingly.

### 4.4. Adaptive Crossover Operator

This paper adopts the adaptive crossover operator based on the process encoding, which can prevent the algorithm from obtaining a local optimal solution. In this way, it can be seen that its quality directly determines the global search ability of the algorithm. The standard NSGA-II algorithm adopts the fixed crossover probability, where  $P_c$  is the crossover probability ( $P_c = 0.4 \sim 0.8$ ). The larger the crossover probability is, the more rapidly the new individuals are generated, and the more easily the excellent individuals can be destroyed. The adaptive crossover operator can judge the population status according to the evolution algebra. When it is relatively dispersed, the crossover is performed with a large probability, and the probability calculation is shown in Equation (19).

$$P_c = \max P_c - (\max P_c - \min P_c) \times \frac{i}{Gen} \tag{19}$$

This paper adopts the POX crossover. First of all, the genes in the chromosome are randomly divided into two parts (i.e., the fixed-order item and fixed-position item), and then the genes of the fixed-position item in the parent individual P1 and P2 are passed to the offspring S1 and S2 without any change. Afterwards, the genes of the fixed-order item of the parent individual P2 and P1 are inserted into the gaps of S1 and S2 in order. Figure 5 shows the crossover operation process.

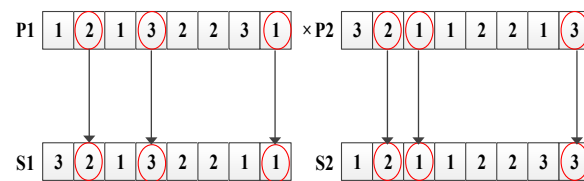


Figure 5. Crossover Operator Based on the Process Coding.

### 4.5. Adaptive Mutation Operator

The standard NSGA-II algorithm adopts the fixed mutation probability, where  $P_m$  is the mutation probability ( $P_m = 0.001 \sim 0.01$ ). If it is too small, it may cause the algorithm falling into premature convergence of the local optimum. The adaptive mutation operator can judge the population status according to the evolution algebra, and perform mutation operation with a large probability when the population individual tends to converge at the later stage of evolution. The probability is calculated as shown in Equation (20).

$$P_m = \min P_m + (\max P_m - \min P_m) \times \frac{i}{Gen} \tag{20}$$

This paper adopts the exchange of genes in the chromosome to achieve individual mutation with the increased population diversification. The x in the following figure is the mutation operator based on the gene exchange. Figure 6 shows the mutation operation process.

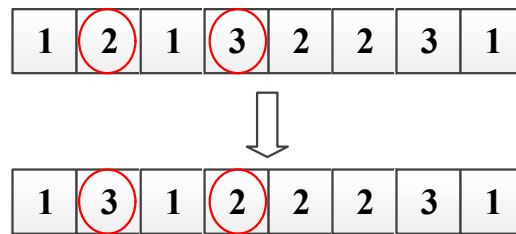


Figure 6. Mutation Operator Based on the Process Coding.

### 5. Instance Verification & Algorithm Comparison

To perform verification: (1) Accept the impact of the acceptance factor value on the maximum makespan and heat treatment equipment utilization rate herein; (2) Through comparing our algorithm with other algorithms based on other decoding rules, prove that the decoding strategy proposed herein has improved the algorithm convergence in terms of handling the FJSP with the heat treatment process. 12 instances MK01-12 are extended from Brandimarte (1993) and the detailed information is shown in Figure 7, where n indicates the job quantity of each instance, oj indicates the quantity scope of each job, m indicates the quantity of machining equipment in this workshop, mh indicates the quantity of batch processing equipment in this workshop, copro indicates the scope of the process quantity contained in each job, meq indicates the scope of the quantity of optional processing equipment for each machining process, proc indicates the scope of the processing time for each machining process, del indicates the scope of the job delivery date, SingleV indicates the volume of the single piece in each job, proh indicates the processing time of the heat treatment process of each job, and HMaxVol indicates the maximum volume of the heat treatment equipment in this workshop.

Instance	Depc	n	oj	m	mh	copro	meq	proc	del	SingleV	proh	HMaxVol
MK1	1–2	10	10–15	6	1	5–7	3	1–7	900–1000	1–10	4–6	50
MK2	1–2	10	10–15	6	1	5–7	6	1–7	1000–1100	1–10	4–6	50
MK3	1–2	15	10–15	8	1	10–10	5	1–20	3900–4000	1–10	4–6	50
MK4	1–2	15	10–15	8	1	3–10	3	1–10	1100–1200	1–10	4–6	50
MK5	1–2	15	10–15	4	1	5–10	2	5–10	3300–3400	1–10	4–6	50
MK6	1–2	10	10–15	15	1	15–15	5	1–10	1300–1400	1–10	4–6	50
MK7	1–2	20	10–15	5	1	5–5	5	1–20	3300–3400	1–10	4–6	50
MK8	1–2	20	10–15	10	1	5–10	2	5–20	4300–4400	1–10	4–6	50
MK9	1–2	20	10–15	10	1	10–15	5	5–20	6500–6600	1–10	4–6	50
MK10	1–2	20	10–15	15	1	10–15	5	5–20	5000–5100	1–10	4–6	50
MK11	1–2	30	10–15	5	1	5–8	2	10–30	6500–6600	1–10	4–6	50
MK12	1–2	30	10–15	10	1	5–10	2	10–30	5000–5100	1–10	4–6	50

Figure 7. Instance Data.

All the above instance verification algorithms run on a personal computer with the processor of i5-2400 CPU, dominant frequency of 3.1 GHz and memory of 4 GB. The improved NSGA proposed herein was used to solve the problem. The number of iteration times of the algorithm was set to 200, the initial population size is set to 100, the maximum crossover probability was set to 0.45, the minimum crossover probability was set to 0.25, the maximum mutation probability was set to 0.1, and the minimum mutation probability was set to 0.01.

To obtain the optimal value of the acceptance factor, in this section, we calculated different values of the acceptance factor, and then obtained the solution result of the FJSP with the heat treatment process with the improved genetic algorithm by following the decoding rule proposed herein. For Case MK10 proposed above, we solved this problem with the acceptance factor, provided that the value range is [0, 1] and the value increment is 0.001. Solved this problem with the acceptance factor at each value for 20 times respectively, and investigated the maximum makespan, heat treatment equipment utilization rate and running time of the best result, which are shown in Figure 8 below.

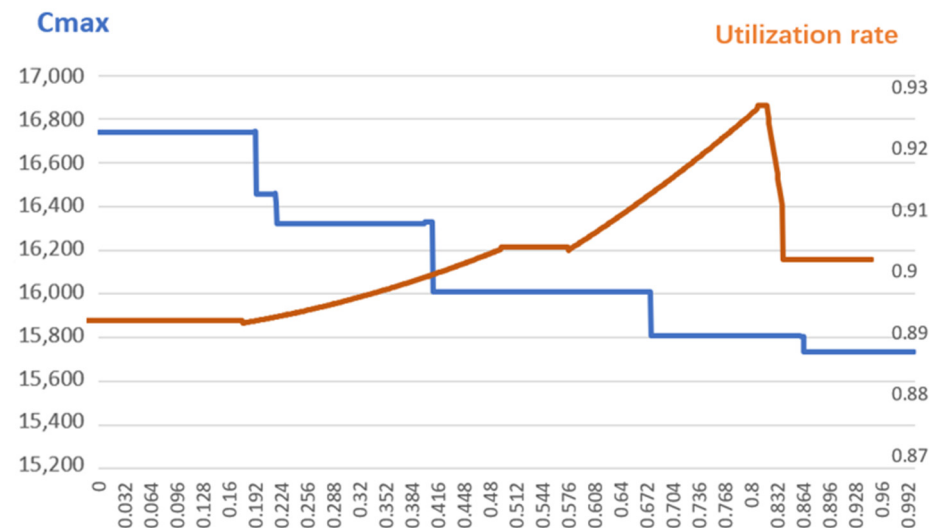


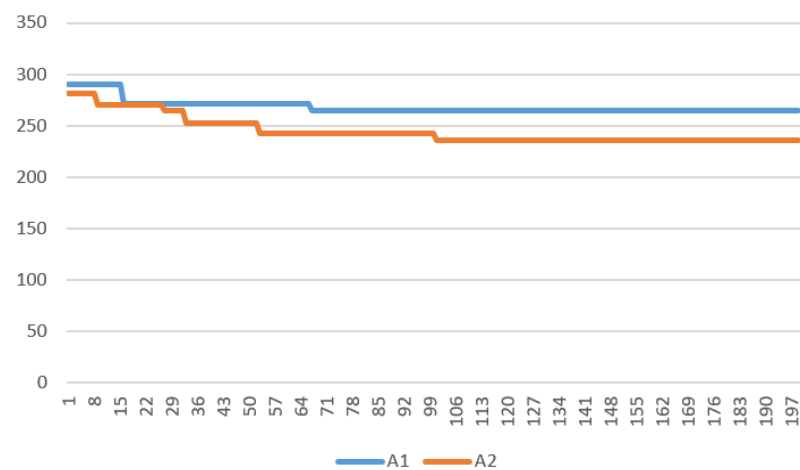
Figure 8. Optimized Curve of the Acceptance Factor for Case MK10.

Because this paper studied the multi-objective optimization problem with the adoption of the non-dominated genetic algorithm, multiple non-dominated solutions may be obtained during each algorithm calculation. Therefore, only the solutions with the shortest running time in the Pareto optimal solution set are compared in this section. As shown in Figure 7, the case solution was obtained within the effective time with the decoding rule and adaptive cross-mutation operator proposed herein, and more than 90% of the utilization rate of the heat treatment equipment volume was obtained. It can be seen from this figure that when the acceptance factor is between [0.860, 0.871], the minimum  $C_{max}$  and maximum equipment utilization rate are obtained. Therefore, when the average value  $\delta = \frac{0.860+0.871}{2} = 0.865$  is taken, the obtained maximum makespan is the smallest, and the utilization rate of the heat treatment equipment volume is the largest.

To further verify the contribution of the decoding strategy proposed herein to the improvement of the algorithm convergence, we performed algorithm comparison with 12 cases proposed in this paper, the number of iteration times of the algorithm was set to 200, and the population quantity was set to 30. Table 3 shows the running results of above 12 cases with different decoding methods for different heat treatment processes in combination with the non-dominated genetic algorithm. Among them, A1 is the improved non-dominated genetic algorithm herein based on the decoding strategy proposed herein; A2 is the inverse scheduling strategy proposed in [18] in combination with the non-dominated genetic algorithm. Table 3 shows that the solutions of 12 cases were calculated with two algorithms for 20 times respectively to evaluate the scheduling objective and the average running time of the algorithm for the best results. Figure 9 shows the convergence diagram of two algorithms for MK12 (where  $C_{max}$  is the average maximum total makespan of workpieces, TP is the average total delay penalty,  $\eta_h$  is the average utilization rate of the heat treatment equipment, and CT is the average running time of the algorithm).

**Table 3.** Running results of the said 12 cases with the two algorithms.

Case	A1				A2			
	$C_{max}$	TP	$\eta_h/\%$	CT/s	$C_{max}$	TP	$\eta_h/\%$	CT/s
MK1	57	3.10	93.17	4.2	73	1.23	85.2	6.4
MK2	53	0	93.89	10.2	64	1.02	75.3	8.7
MK3	319	0	93.13	78.9	327	13.45	79.8	61.9
MK4	117	0	94.01	21.8	147	38.76	81.9	14.8
MK5	249	9.81	92.27	18.7	256	46.87	74.6	12.9
MK6	164	0	91.82	62.1	193	21.92	72.8	57.8
MK7	228	0	94.32	42.7	236	46.91	70.7	29.4
MK8	485	0	93.79	37.2	534	5.89	74.9	22.8
MK9	541	11.68	94.21	191.4	567	38.64	73.1	185.1
MK10	532	0	89.88	179.6	576	29.36	75.2	185.0
MK11	1039	0	92.31	47.0	1169	101.62	75.0	41.4
MK12	1051	17.39	90.85	52.4	1089	123.47	75.7	45.6

**Figure 9.** Convergence Diagram of Two Algorithms for Case MK12.

It can be seen from the said result data and convergence graph that for the different case size:

- (1) From the perspective of the optimal  $C_{max}$  obtained with this algorithm: A1 can always obtain the optimal  $C_{max}$  smaller than that of A2, and this advantage becomes more significant as the scale of the case increases. It has proved that the decoding rules based on the heat treatment equipment volume and job delivery date proposed in this paper can significantly improve the global optimization capability of an algorithm.
- (2) From the perspective of the heat treatment equipment utilization rate obtained with the algorithm: for all cases with different sizes, A1 can get a higher heat treatment equipment utilization rate (all cases with A1 are greater than 88%, and most cases are above 90%), and A1 is significantly better than A2.
- (3) From the perspective of the algorithm running time: the running time of the two algorithms is equivalent to each other, which has proved that the improved genetic algorithm and the decoding rules based on the heat treatment equipment volume and job delivery date proposed in this paper can complete the iterative optimization within the limited running time.

- (4) From the perspective of the algorithm convergence speed: two algorithms have excellent convergence at the initial stage, but A1 has the faster convergence speed and better algorithm performance than A2 during the entire iteration process.

Figure 10 is the Pareto frontier of Case MK3. To display the output of the scheduling result with the heat treatment process guided by the decoding rules, we generated the equipment Gantt chart of a non-dominated solution of the Pareto frontier.

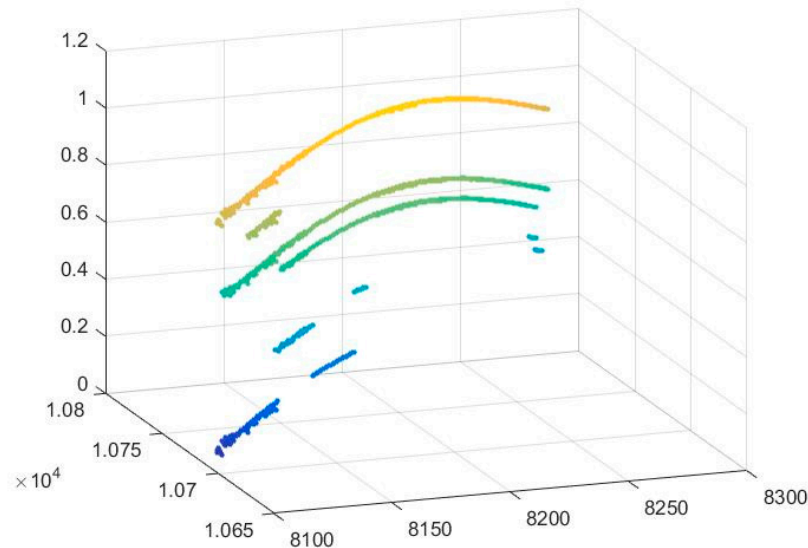


Figure 10. Pareto frontier.

Figure 11 shows the equipment Gantt chart of the MK3 result obtained with the algorithm proposed herein, where Equipment 6 is the heat treatment equipment. It can be seen from the figure and Table 4 that the heat treatment process has been taken under consideration timely, other machining processes are tightly scheduled on the equipment, and the equipment utilization rate is high [19].

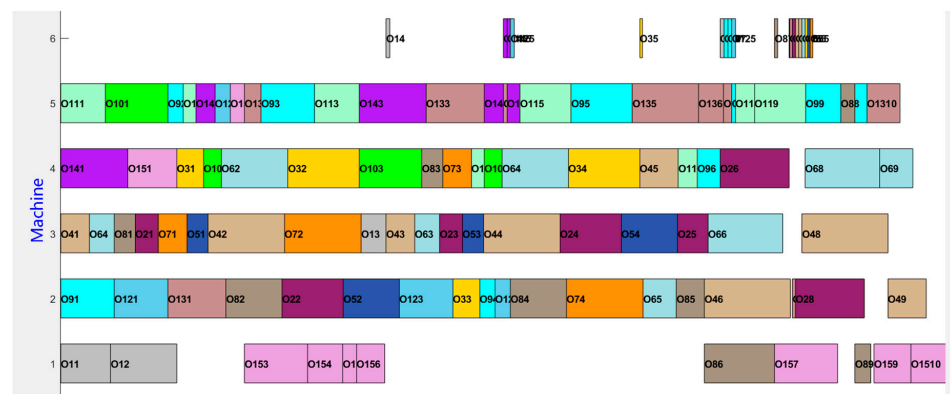


Figure 11. Equipment Gantt Chart for the Corresponding Production Scheduling Results of Case MK3.

Table 4. A non-dominated optimization target value and running time of MK3.

Case	$C_{max}$	TP	$\eta_h$	CT
MK3	327	0	93.27	71.6

Among them, the batches on the heat treatment equipment corresponding to MK3 are shown in Figure 12 below. It can be seen from the data in this figure that the heat



treatment equipment has been almost always in the running status after it started running, and the equipment utilization rate of each heat treatment batch is high, which has directly proved the effectiveness of the coding-encoding method containing the heat treatment process proposed herein to solve the actual problems. In addition, it can be seen from the following figure that the most of orders are processed in batch for the heat treatment, but the heat treatment processing time of various batches is concentrated, which is more beneficial to minimizing  $C_{max}$  and ensuring the delivery time of workpieces. Meanwhile, it has also proved the effectiveness of the decoding rules for the machining process based on the equipment bottleneck and the heat treatment process based on the heat treatment equipment volume and job delivery data.

Batch No.	Quantity of Batches on Heat Treatment	Maximum Volume of Batch Processing	Batch Quantity	Start Time	End Time	Included Order and Piecework Quantity	Batch Use Volume
1	23	50	1	510	516	J01-12	48
2	23	50	2	1034	1040	J01-1/J09-4	44
3	23	50	1	1040	1046	J09-5	50
4	23	50	2	1046	1052	J09-3/J10-10	50
5	23	50	2	1052	1058	J10-4/J11-10	48
6	23	50	1	705	711	J12-8	48
7	23	50	2	1052	1058	J12-4/J14-2	44
8	23	50	1	694	700	J14-5	50
9	23	50	1	700	706	J14-5	50
10	23	50	1	694	700	J14-3	30
11	23	50	1	1142	1147	J02-10	50
12	23	50	2	1147	1152	J02-2/J04-4	46
13	23	50	1	1152	1157	J04-5	45
14	23	50	2	1157	1162	J04-5/J06-1	48
15	23	50	2	1162	1167	J06-12/J08-2	50
16	23	50	1	1119	1124	J08-7	49
17	23	50	2	1218	1223	J08-2/J15-11	36
18	23	50	1	908	912	J03-7	49
19	23	50	2	1167	1171	J03-7/J05-1/	50
20	23	50	2	1171	1175	J05-10/J07-8/	50
21	23	50	2	1175	1179	J07-7/J13-1/	43
22	23	50	1	1143	1147	J13-6/	48
23	23	50	1	1143	1147	J13-6/	48

**Figure 12.** Heat Treatment Batches on the Heat Treatment Equipment corresponding to MK3.

Figure 13 shows the equipment Gantt chart of the MK7 result obtained with the algorithm proposed herein, where Equipment 6 is heat treatment equipment, and Table 5 shows the optimization target value and running time of this non-dominated solution. As can be seen from the figure below, the machining equipment has almost been running during the work, the heat treatment equipment has rarely waited during the processing, the utilization rate of the heat treatment equipment is 94.13%, and heat treatment batches are connected with each other at an almost zero gap. Such phenomena have proved the effectiveness of the decoding rules for the heat treatment process based on the heat treatment equipment volume and delivery date proposed herein.

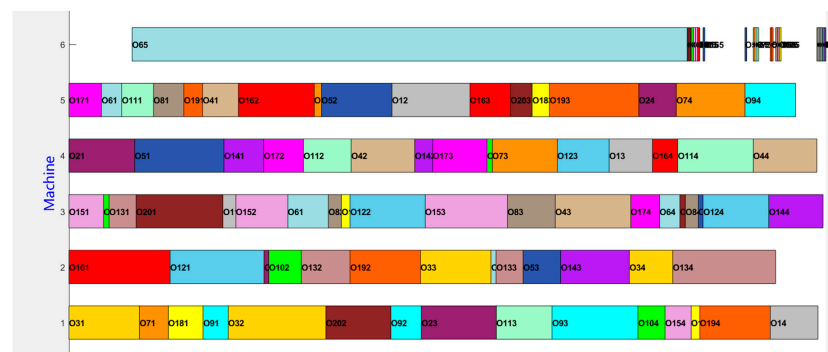


Figure 13. Equipment Gantt Chart for the Production Scheduling Results of Case MK7.

Table 5. A non-dominated optimization target value and running time of MK7.

Case	$C_{max}$	TP	$\eta_h$	CT
MK7	221	1.01	94.13	43.8

Figure 14 shows the equipment Gantt chart of the MK12 result obtained with the algorithm proposed herein, where equipment 8 is heat treatment equipment as shown in the figure. Meanwhile, it also has proved the effectiveness of the decoding rules for the heat treatment process-machining process proposed herein in large cases. Table 6 shows the optimization target value and running time of this non-dominated solution.

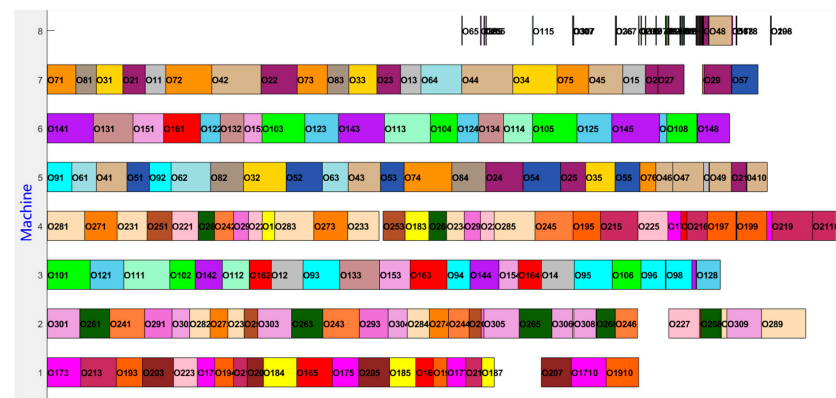


Figure 14. Equipment Gantt Chart for the Production Scheduling Results of Case MK12.

Table 6. A non-dominated optimization target value and running time of MK12.

Case	$C_{max}$	TP	$\eta_h$	CT
MK12	1040	0	92.77	57.9

### 6. Conclusions

This paper investigated the FJSP with the heat treatment process from a novel perspective. To describe and solve this problem, we built an unified mathematical model of the heat treatment process and machining process. To achieve scheduling optimization objectives, such as minimizing  $C_{max}$ , maximizing the space utilization rate of the heat treatment equipment, and minimizing the total delay penalty, this paper proposed a set of decoding rules based on the heat treatment equipment volume and job delivery date. This set of rules regards the heat treatment process as a node, and divides the decoding process into two coupling parts, that is, the heat treatment process scheduling and the machining process scheduling.

With the aim to minimize  $C_{max}$ , we proposed a set of decoding rules for the machining process based on the equipment bottleneck to schedule the machining process by considering the dynamic process arrival time. For the heat treatment process, when the utilization rate of the heat treatment equipment volume is maximized, and the job delivery date is taken under consideration, we proposed a set of decoding rules for the heat treatment process based on the heat treatment equipment volume and the job delivery date to schedule the heat treatment process. With this decoding rules, it is preferred to minimize the number of workpiece batches in the same job, reduce the waiting time of the job to be processed, and ensure the delivery date of the job. Finally, the subsequent machining process is scheduled by following the decoding rules for the machining process based on the equipment bottleneck to achieve the decoding of one chromosome in the end.

This decoding process can be oriented to the problem itself to avoid the generation of invalid and poor solutions. In this way, a proper solution can be obtained in combination with the improved adaptive non-dominant genetic algorithm.

To verify the effectiveness of the decoding rules and algorithms proposed herein, this paper improved 12 cases in the Brandimarte (1993) literature. Furthermore, we obtained several solutions with the decoding rules and improved non-dominated genetic algorithms proposed herein, and the reverse scheduling strategy proposed in the reference in combination with the genetic algorithm, which have further proved the effectiveness and feasibility of the set of decoding rules based on the heat treatment equipment volume and job delivery date, and the improved non-dominant genetic algorithm proposed herein. It has guiding significance for the production of aerospace pumps and valves and the scheduling of the workshops with the same production mode.

**Author Contributions:** Conceptualization, Y.L.; Data curation, Y.L. and Q.Z.; Formal analysis, Y.L. and Q.Z.; Funding acquisition, Q.Z.; Investigation, Y.L.; Methodology, Y.L. and Q.Z.; Writing—original draft, Y.L.; Writing—review & editing, Y.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by University of Science and Technology.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Ikura, Y.; Gimple, M. Efficient scheduling algorithms for a single batch processing machine. *Oper. Res. Lett.* **1986**, *5*, 61–65. [[CrossRef](#)]
2. Chandru, V.; Lee, C.Y.; Uzsoy, R. Minimizing total completion time on batch processing machines. *Int. J. Prod. Res.* **1993**, *31*, 2097–2121. [[CrossRef](#)]
3. Uzsoy, R. Scheduling a single batch processing machine with non-identical job sizes. *Int. J. Prod. Res.* **1994**, *32*, 1615–1635. [[CrossRef](#)]
4. Dupont, L.; Jolai, F. Minimizing makespan on a single batch processing machine with non-identical job sizes. *Eur. J. Autom. Syst.* **1998**, *32*, 431–440.
5. Azizoglu, M.; Webster, S. Scheduling a Batch Processing Machine with Non-Identical Job Sizes. 2000. Available online: <https://www.tandfonline.com/doi/abs/10.1080/00207540050028034> (accessed on 24 September 2021).
6. Chang, P.-C.; Wang, H.-M. A heuristic for a batch processing machine scheduled to minimise total completion time with non-identical job sizes. *Int. J. Adv. Manuf. Technol.* **2004**, *24*, 615–620. [[CrossRef](#)]
7. Melouk, S.; Damodaran, P.; Chang, P.-Y. Minimizing makespan for single machine batch processing with non-identical job sizes using simulated annealing. *Int. J. Prod. Econ.* **2004**, *87*, 141–147. [[CrossRef](#)]
8. Damodaran, P.; Manjeshwar, P.K.; Srihari, K. Minimizing makespan on a batch-processing machine with non-identical job sizes using genetic algorithms. *Int. J. Prod. Econ.* **2006**, *103*, 882–891. [[CrossRef](#)]
9. Damodaran, P.; Srihari, K.; Lam, S.S. Scheduling a capacitated batch-processing machine to minimize makespan. *Robot. Comput.-Integr. Manuf.* **2007**, *23*, 208–216. [[CrossRef](#)]

10. Manjeshwar, P.; Damodaran, P.; Srihari, K. Minimizing makespan in a flow shop with two batch-processing machines using simulated annealing. *Robot. Comput.-Integr. Manuf.* **2009**, *25*, 667–679. [[CrossRef](#)]
11. Gerstl, E.; Mosheiov, G. A two-stage flexible flow shop problem with unit-execution-time jobs and batching. *Int. J. Prod. Econ.* **2014**, *158*, 171–178. [[CrossRef](#)]
12. Xu, R.; Chen, H.; Li, X. Makespan minimization on single batch-processing machine via ant colony optimization. *Comput. Oper. Res.* **2012**, *39*, 582–593. [[CrossRef](#)]
13. Jia, Z.-H.; Leung, J.Y.T. An improved meta-heuristic for makespan minimization of a single batch machine with non-identical job sizes. *Comput. Oper. Res.* **2014**, *46*, 49–58. [[CrossRef](#)]
14. Zeng, Z.; Hong, M.; Man, Y.; Li, J.; Zhang, Y.; Liu, H. Multi-object optimization of flexible flow shop scheduling with batch process—Consideration total electricity consumption and material wastage. *J. Clean. Prod.* **2018**, *183*, 925–939. [[CrossRef](#)]
15. Muter, B. Exact algorithms to minimize makespan on single and parallel batch processing machines. *Eur. J. Oper. Res.* **2020**, *285*, 470–483. [[CrossRef](#)]
16. Fengping, S.; Jingjing, L.; Yulong, Y. Heat Treatment Batch Scheduling Model and PSO Solution for the Minimum Waiting Time. Available online: [https://tow.cnki.net/kcms/detail/detail.aspx?filename=JSJZ202004042&dbcode=CRJT\\_CJFD&dbname=CJFDLAST2020&v=](https://tow.cnki.net/kcms/detail/detail.aspx?filename=JSJZ202004042&dbcode=CRJT_CJFD&dbname=CJFDLAST2020&v=) (accessed on 24 September 2021).
17. Rong, L.; Lin, Z.; Chao, W. Research on Flexible Job-Shop Scheduling Problem with the Parallel Batch Processor. Available online: <https://xueshu.baidu.com/usercenter/paper/show?paperid=1e4g0xp0pr5k0pt0ps4f0pd0w5584347> (accessed on 24 September 2021).
18. Jiangxu, Q. Integrated Scheduling Method of Machine-Heat Treatment in Complex Manufacturing System. Ph.D. Thesis, Beijing Institute of Technology, Beijing, China, December 2011.
19. Li, H. Research on Batch Scheduling Methods of the Flexible Flow Shop Based on DES-GEP. Ph.D. Thesis, Dalian University of Technology, Dalian, China, June 2021.