

## Research Article

# Dynamic Adjustment Strategy of Rail Guide Vehicle

Wenbin Liu <sup>1,2</sup> and Dongbing Liu <sup>3</sup>

<sup>1</sup>Wuhan Technology and Business University, Wuhan 430065, China

<sup>2</sup>Wuchang Institute of Technology, Wuhan 430065, China

<sup>3</sup>College of Mathematics and Computer, Panzhihua University, Panzhihua 617000, Sichuan, China

Correspondence should be addressed to Wenbin Liu; [liuwenbin@wtbu.edu.cn](mailto:liuwenbin@wtbu.edu.cn) and Dongbing Liu; [liudongbingcqu@163.com](mailto:liudongbingcqu@163.com)

Received 27 August 2021; Revised 27 September 2021; Accepted 30 September 2021; Published 30 October 2021

Academic Editor: Sang-Bing Tsai

Copyright © 2021 Wenbin Liu and Dongbing Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

For the dynamic adjustment strategy of intelligent rail guide vehicle, the dynamic planning model has been built with the longest working time in the paper. In solving this problem, the longest target of the rail guide vehicles' running time is converted to find the optimal moving path of the rail guide vehicles. Drawing on the shortest path idea, the model uses the Floyd algorithm and the simulated annealing algorithm. The longest running time of the rail guide vehicles in the model is the innovation point in the paper. When the rail guide vehicles path is optimal, when the rail guide vehicles is the busiest, the system is the most efficient.

## 1. Introduction

In Figure 1, the intermediate equipment is a rail guide vehicle with a cleaning groove and a manipulator. Rail guide vehicle is abbreviated as RGV. A cleaning groove washes only a material at a time. There are two claws at the front of the manipulator; through rotation, it can grab one material each to complete the loading and unloading operation. Computer number controllers are arranged on both sides [1–3]. Computer number controller is abbreviated as CNC. A material conveyor belt is installed in front of each CNC. The loading conveyor belt is on the right and is responsible for delivering unprocessed material to the CNC. The unloading conveyor belt is on the left and is responsible for sending the processed and cleaned material out of the system [4, 5]. Other auxiliary equipment is designed to ensure the normal operation of the system [6–8].

RGV is an abbreviation for rail guide vehicles. With the rapid development of the industrial age, the mechanization of industry becomes more and more frequent. According to the demand of different industrial processing, RGV can be designed to run the track rail for the longest time [9–11]. In the process of industrial operation, the reduction of personnel improves the safety, is not affected by the special circumstances such as time and space, effectively improves

the efficiency of industrial processing, promotes industrial production, and maximizes the profits obtained by industrial manufacturers [12–14]. These are inseparable from intelligent processing systems. At the same time, intelligent machinery also needs to be improved artificially [4, 5]. RGV has an intelligent control capability to receive and send command signals [15–18].

Figure 2 shows the schematic figure about intelligent machining system of rail guide vehicle. There are eight computer number controllers (CNCs), a rail guide vehicle (RGV), and two conveyor belts for materials. The rail guide vehicle is an intelligent machine. It does not need to be run by people [19–21]. It can run freely on a fixed track. According to the instruction, it can automatically control the moving direction and distance. There are one mechanical arm, two mechanical claws, and a material cleaning tank in a rail guide vehicle. It can finish some tasks like loading, unloading, and cleaning material [22–25].

## 2. Material Processing in a Process

First of all, only one-process material processing situation is considered, and a model that can make RGV work more efficiently is established. To make RGV work most efficiently, we assume that RGV is moving all the time for eight hours of



FIGURE 1: Workshop layout diagram of RGV—CNC.

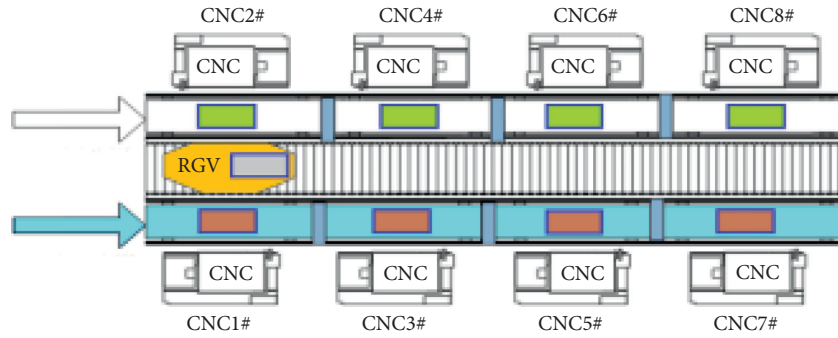


FIGURE 2: Intelligent machining system of rail guide vehicle.

a shift, and when multiple CNCs send a demand signal to the RGV, the RGV chooses to work on the nearest CNC to it. From this, we transform the time problem into the displacement problem, which can simplify the RGV dynamic scheduling model and the corresponding solution algorithm to the problem of finding the longest running time of the rail guide vehicle. The cyclic dynamic programming model is established firstly. Then, the MATLAB software is used to program and solve the problem. Finally, the MATLAB software is used to find out the path that makes the RGV work most efficiently.

### 2.1. Cyclic Dynamic Programming Model

- (1) Four phases: the first stage comprises the first CNC and the second CNC. The second stage comprises the third CNC and fourth CNC. The third stage comprises the fifth CNC and sixth CNC. The fourth stage comprises the seventh CNC and eighth CNC.

- (2) Selecting state  $S_k$ :

$$S_k = (\text{CNC}\#k, \text{CNC}\#k + 1). \quad (1)$$

- (3) Determining strategies  $\mu_k(s_k)$ : RGV units of movement are represented for 1, 2, 3 or -1, -2, -3.
- (4) Decision value  $v_k(S_k)$ : working hours of RGV = movement time of RGV + loading, cleaning, and discharging time of CNC#k and CNC#k + 1.

- (5) Writing the state transition equation:

$$S_{k+1} = T_k[S_k, \mu_k(s_k)]. \quad (2)$$

- (6) Establishing index function  $f_{1k}(S_k)$ : RGV working times from the initial moment to the end of this phase.

- (7) Establishing objective function:

$$\text{Max} f_{1k}(S_k) = \sum_k v_k(S_k). \quad (3)$$

After eight hours, it eventually stops at a certain stage.

**2.2. Floyd Algorithm.** Through the Floyd algorithm, the shortest path time of each vertex is calculated in the graph  $G = (V, E)$ . If  $w(i, j)$  is represented for the realizable distance between  $i$  and  $j$ ,  $w(i, j)$  is represented for the shortest distance between  $i$  and  $j$ . For, for any  $i, j, k$ , there is  $w(i, j) \leq w(i, k) + w(k, j)$ . The algorithm is represented by matrix and calculated systematically. By iteration, the situation by which the above theorem is not satisfied is eliminated. Each  $n \times n$  matrix  $E$  and matrix  $R$  in order for a graph can be calculated with a given edge length  $d_{ij}$  at  $n$  ports.  $d_{ij}$  in the matrix  $W$  is represented for the distance from the  $i$  vertex to the  $j$  vertex.  $d_{ij}$  in the matrix  $P$  is represented for the distance from the  $i$  vertex to the  $j$  vertex (vertex represented by recorded values  $b_{ij}$ ), i.e., a transfer route.

The steps are as follows:

*Step 1.  $F_0$ :*

$$\begin{aligned} W_{ij}^{(0)} &= [W_{ij}]^{(0)}, \\ W_{ij}^{(0)} &= \begin{cases} d_{ij}, & \text{there is a edge between } v_i \text{ and } v_j, \\ \infty, & \text{there isn't a edge between } v_i \text{ and } v_j \\ 0, & i = j, \end{cases} \\ R^0 &= [r_{ij}^{(0)}], \\ r_{ij}^{(0)} &= \begin{cases} j, & W_{ij}^{(0)} < \infty, \\ 0, & W_{ij}^{(0)} = \infty \text{ or } i = j. \end{cases} \end{aligned} \quad (4)$$

*Step 2.  $F_1$ :* if we know  $W^{(k-1)}$  and  $R^{(n)}$ , we may solve as follows.

$$\begin{aligned} W_{ij}^{(k)} &= \min [W_{ij}^{(k-1)}, W_{ik}^{(k-1)} + W_{kj}^{(k-1)}], \\ r_{ij}^k &= \begin{cases} r_{ij}^{(k-1)}, & \text{if } W_{ij}^{(k)} = W_{ij}^{(k-1)}, \\ r_{ik}^{(k-1)} + r_{kj}^{(k-1)}, & \text{if } W_{ij}^{(k)} = W_{ik}^{(k-1)} + W_{kj}^{(k-1)}. \end{cases} \end{aligned} \quad (5)$$

*Step 3.  $F_2$ :* if  $k < n$ , repeat  $F_1$ . If  $k = n$ , end.

If  $W_{ij}^{(k-1)} \rightarrow W_{ij}^{(k)}$  can shorten the length of the road,  $W_{ij}^{(k)}$  is changed and noted in the  $R$  matrix. Finally, we can obtain  $W^{(n)}$  and  $R^{(n)}$ . The shortest path length and transfer route are obtained.

### 2.3. Solution of Floyd Algorithm

- (1) A matrix sequence  $B_1, \dots, B_k, \dots, B_8$  is recursively generated. Line  $i$ , column  $j$  elements of the matrix  $A_{k(i,j)}$  represent the shortest path length in which the number of vertex passing through the path is not greater than  $k$  from vertex  $v_i$  to vertex  $v_j$ . The iteration formula used in the calculation is as follows.

$$\begin{aligned} B_k(i, j) &= \min(B_{k-1}(i, j), B_{k-1}(i, k) + B_{k-1}(k, j)), \\ &(k \text{ is number of iterations, } i, j, k = 1, 2, \dots, 8). \end{aligned} \quad (6)$$

- (2) When  $k = n$ ,  $B_n$  is the ordinal number of passing vertices through the shortest path between vertices.

*Step 1.* The two matrices are initialized, and matrix  $W$  is obtained as shown in Table 1 and matrix  $P$  is obtained as shown in Table 2.

*Step 2.* By the iterative formula (1), the number of iterations is 8 by using the MATLAB equation.

*Step 3.* It can be concluded that when only one-process material processing operation is considered,  $B_8$  (number of iterations is 8) is the serial number of the vertices passing through the shortest path between the vertices.

*2.4. Testing the Practicability of the Model and the Effectiveness of the Algorithm.* The intelligent processing system is used to obtain three sets of data of operation parameters as shown in Table 3.

On the basis of the established cyclic dynamic programming model, three groups of data in Table 3 are put into the program by Floyd algorithm and MATLAB software, respectively. RGV scheduling strategy and work efficiency tables for one process to 3 groups can be obtained as shown in Table 4.

*2.5. Consideration of Occurring Failures.* During processing, CNC may fail. According to statistics, the probability of failure is about 1%. Unfinished materials are scrapped if handled manually. The time for each troubleshooting is between 10 and 20 minutes. Operation sequence is immediately run after troubleshooting.

#### 2.5.1. Circular Dynamic Programming Model

- (1) Four phases: the first stage comprises the first CNC and the second CNC. The second stage comprises the third CNC and fourth CNC. The third stage comprises the fifth CNC and sixth CNC. The fourth stage comprises the seventh CNC and eighth CNC.
- (2) Selecting state  $S_k$ :

$$S_k = (\text{CNC}\#k, \text{CNC}\#k + 1). \quad (7)$$

- (3) Determining strategies  $\mu_k(s_k)$ : RGV units of movement are represented for 1, 2, 3 or  $-1, -2, -3$ .
- (4) Decision value  $v_k(S_k)$ : working hours of RGV = movement time of RGV + loading, cleaning, and discharging time of CNC# $k$  and CNC# $k + 1$ .
- (5) Writing the state transition equation:

$$S_{k+1} = T_k[S_k, \mu_k(s_k)]. \quad (8)$$

- (6) Establishing index function  $f_{1k}(S_k)$ : RGV working hours from the initial moment to the end of this phase.
- (7) Establishing objective function:

$$\text{Max } f_{1k}(S_k) = \sum_k v_k(S_k). \quad (9)$$

After eight hours of constraint, it eventually stops at a certain stage.

TABLE 1: Matrix  $W$ : the distance between two CNCs in RGV motion.

	CNC1	CNC2	CNC3	CNC4	CNC5	CNC6	CNC7	CNC8
CNC1	$\nu_{11}$	$\nu_{12}$	$\nu_{13}$	$\nu_{14}$	$\nu_{15}$	$\nu_{16}$	$\nu_{17}$	$\nu_{18}$
CNC2	$\nu_{21}$	$\nu_{22}$	$\nu_{23}$	$\nu_{24}$	$\nu_{25}$	$\nu_{26}$	$\nu_{27}$	$\nu_{28}$
CNC3	$\nu_{31}$	$\nu_{32}$	$\nu_{33}$	$\nu_{34}$	$\nu_{35}$	$\nu_{36}$	$\nu_{37}$	$\nu_{38}$
CNC4	$\nu_{41}$	$\nu_{42}$	$\nu_{43}$	$\nu_{44}$	$\nu_{45}$	$\nu_{46}$	$\nu_{47}$	$\nu_{48}$
CNC5	$\nu_{51}$	$\nu_{52}$	$\nu_{53}$	$\nu_{54}$	$\nu_{55}$	$\nu_{56}$	$\nu_{57}$	$\nu_{58}$
CNC6	$\nu_{61}$	$\nu_{62}$	$\nu_{63}$	$\nu_{64}$	$\nu_{65}$	$\nu_{66}$	$\nu_{67}$	$\nu_{68}$
CNC7	$\nu_{71}$	$\nu_{72}$	$\nu_{73}$	$\nu_{74}$	$\nu_{75}$	$\nu_{76}$	$\nu_{77}$	$\nu_{78}$
CNC8	$\nu_{81}$	$\nu_{82}$	$\nu_{83}$	$\nu_{84}$	$\nu_{85}$	$\nu_{86}$	$\nu_{87}$	$\nu_{88}$

TABLE 2: Matrix  $P$ : passing vertices between two CNCs.

	CNC1	CNC2	CNC3	CNC4	CNC5	CNC6	CNC7	CNC8
CNC1	0	1	2	3	4	5	6	7
CNC2	0	1	2	3	4	5	6	7
CNC3	0	1	2	3	4	5	6	7
CNC4	0	1	2	3	4	5	6	7
CNC5	0	1	2	3	4	5	6	7
CNC6	0	1	2	3	4	5	6	7
CNC7	0	1	2	3	4	5	6	7
CNC8	0	1	2	3	4	5	6	7

TABLE 3: Three sets of data of operation parameters of the intelligent processing system (unit of time is seconds).

System operation parameters	Group 1	Group 2	Group 3
RGV time required to move 1 unit	20	23	18
RGV time required to move 2 units	33	41	32
RGV time required to move 3 units	46	59	46
CNC material requirement time to finish a process	560	580	545
RGV time required for one loading and discharging to CNC1#, CNC3#, CNC5#, CNC7#	28	30	27
RGV time required for one loading and discharging to CNC2#, CNC4#, CNC6#, CNC8#	31	35	32
RGV time required to complete a material cleaning operation	25	30	25

TABLE 4: RGV scheduling strategy and work efficiency for one process for 3 groups.

Group	RGV scheduling strategy	Number of processed materials	Work efficiency
Group 1	4-6-5-8-7-1-2-3	405	50.62
Group 2	5-4-3-1-2-8-7-6	390	48.75
Group 3	1-2-3-4-5-6-7-8	417	52.12

2.5.2. *Solution of the Model.* From the establishment of the model, it can be found that the model is the same when the fault occurs and when the fault does not occur, but the algorithm should be changed when the MATLAB software is used to calculate.

The specific approach is as follows.

A random number producer is set up, which is assigned to each machining table according to equal probability. When the RGV rotates to the faulty machining table, only the RGV motion time is considered, not the time of feeding, discharging, and cleaning RGV.

Through the Floyd algorithm and the MATLAB software, the three groups of data in Table 3 are substituted into the program. We can get three sets of data on material processing failures. In one process, the first set of data on material processing failures is shown in Table 5.

### 3. Material Processing in Two Processes

Each material's first and second processes are processed by two different CNCs in turn. According to the problem analysis of the material processing operation of only one working procedure, the RGV dynamic scheduling model and the corresponding algorithm problem are obtained, which include the material processing operation of two working procedures. The dynamic scheduling model with the highest working efficiency is established. If the process of material requires two processes, it is necessary to have different CNCs to install different tools to complete the processing; the tool cannot be replaced during processing. The first and second processes need to be processed in turn on different CNCs, and the completion time is different. Each CNC can only complete one of them. Through analysis,

TABLE 5: The first set of data on material processing failures.

Material serial number	CNC number	Start time	Closing time
53	7	3720	0
61	7	0	4874
266	6	19117	0
274	6	0	20271
362	6	26041	0
370	6	0	27195
380	8	27324	0
388	8	0	28478

it is decided to use odd-number CNC to process the first process and even-number CNC to process the second process. The two-dimensional dynamic programming model is established by the method of partition processing under the condition that only one working procedure is considered.

### 3.1. Setting Up a Two-Dimensional Dynamic Programming Model considering the Material Processing in Two Processes

- (1) First of all, it is assumed that CNC# $m$  ( $m = 1, 2, 3, 4, 5, 6, 7, 8$ ) is installed as follows. CNC#  $m$  ( $m = 1, 3, 5, 7$ ) was installed as the first procedure. CNC#  $m$  ( $m = 1, 3, 5, 7$ ) was installed as the second procedure.
- (2)  $k$  phase variables are set,  $k = 1, 2, 3, 4, 5, 6, 7, 8$ .
- (3) The state variable  $S_k^1$  is set to CNC# $k$  state. The state variable  $S_k^2$  is set to CNC# $k + 1$  state.
- (4) Policy variables are  $\mu_k^1(s_k^1)$  and  $\mu_k^2(s_k^2)$ .  $\mu_k^1(s_k^1)$  means that the RGV unit of movement is 1, 2, 3.  $\mu_k^2(s_k^2)$  means that the RGV unit of movement is -1, -2, -3.
- (5) Decision variables are  $\mu_k^1(s_k^1)$  and  $\mu_k^2(s_k^2)$ . RGV working hours = RGV movement time + CNC# $k$  and CNC# $k + 1$  feeding, cleaning, and discharging time.
- (6) State transfer equation:

$$\begin{aligned} S_{k+1}^1 &= T_k [S_k^1, \mu_k^1(S_k^1)], \\ S_{k+1}^2 &= T_k [S_k^2, \mu_k^2(S_k^2)]. \end{aligned} \quad (10)$$

(7) Setting up the index function  $f_{1k}(S_k^1, S_k^2)$ :  $f_{1k}(S_k^1, S_k^2)$  represents RGV working hours from inception to the end of the phase.

(8) Establishing objective function:

$$\text{Max } f_{1k}(S_k^1, S_k^2) = \sum_k [v_k^1(S_k^1) + v_k^2(S_k^2)]. \quad (11)$$

### 3.2. Establishing a Simulated Annealing Algorithm by considering the Processing of Two Materials

(1) Assuming the energy of the material is  $E(i)$  under state  $i$ , then the material entering from state  $i$  to state  $j$  follows the following rule.

If  $E(j) \leq E(i)$ , we accept that the state is converted. If  $E(j) > E(i)$ , the probability of the acceptance of the state is  $e^{(E(i)-E(j))/KT}$ .

(2) Probabilities in state  $i$  satisfy the Boltzmann distribution.

$$P_T(X = i) = \frac{e^{-(E(i)/KT)}}{\sum_{j \in S} e^{-(E(j)/KT)}}. \quad (12)$$

Obviously,

$$\lim_{T \rightarrow \infty} \frac{e^{-(E(i)/KT)}}{\sum_{j \in S} e^{-(E(j)/KT)}} = \frac{1}{|S|}. \quad (13)$$

This suggests that all states have the same probability at high temperatures. When the temperature drops, there is

$$\lim_{T \rightarrow 0} \frac{e^{-(E(i)-E_{\min})/KT}}{\sum_{j \in S} e^{-(E(j)-E_{\min})/KT}} = \lim_{T \rightarrow 0} \frac{e^{-(E(i)-E_{\min})/KT}}{\sum_{j \in S_{\min}} e^{-(E(j)-E_{\min})/KT}} = \begin{cases} \frac{1}{|S_{\min}|}, & i \in S_{\min}, \\ 0, & \text{other,} \end{cases} \quad (14)$$

$$E_{\min} = \min_{j \in S} E(j),$$

$$S_{\min} = \{i | E(i) = E_{\min}\}.$$

The above formula shows that when the temperature drops to very low, the material enters the minimum energy state with a high probability. The problem to be solved is an optimization problem seeking a minimal path, so the simulated annealing optimization method is obtained by applying the path optimization problem in physics to the simulated annealing optimization problem in this problem.

### 3.3. Solving Simulated Annealing Algorithm by considering the Material Processing Situation of Two Processes

- (1) First, an initial journey  $T_0$  and an initial solution  $x(0)$  to the optimization problem are given.  $x(0)$  generate the next solution  $x'$  ( $x' \in N[x(0)]$ ). When it gets satisfied on the following probability,  $x'$  is a new solution  $x(1)$ :

$$P(x(0) \longrightarrow x') = \begin{cases} 1, & f(x') < f(x(0)), \\ e^{-(f(x')-f(x(0)))/T_0}, & \text{other.} \end{cases} \quad (15)$$

- (2) For a certain journey  $T_i$  and a solution  $x(k)$  to the optimization problem,  $x'$  can be generated. When it satisfy on the following probability,  $x'$  is a new solution  $x(k+1)$ :

$$P(x(k) \longrightarrow x') = \begin{cases} 1, & f(x') < f(x(k)) \\ e^{-(f(x')-f(x(k)))/T_0}, & \text{other.} \end{cases} \quad (16)$$

Under the journey  $T_i$ , after many transfers, we found the journey  $T_{i+1} < T_i$ . Under the conditions  $T_{i+1}$ , the above process is repeated. We constantly find the new solution and the alternate process of the last journey. Finally, the optimal solution is obtained with the shortest distance within 8 hours of continuous operation in each shift.

**3.4. Solution of Material Processing Model in Two Processes.** According to the simulated annealing algorithm model, by replacing the three groups of data in Table 3 with the MATLAB software, we can obtain the working time of the material processing as shown in Table 6.

Using the annealing algorithm, RGV scheduling strategy can be obtained as shown in Table 7 and work efficiency can be obtained as shown in Table 8 for 3 groups for two processes.

It is seen from Table 4 and 8 that within one working day, the number of two processing processes is reduced by about half, but the work efficiency tends to be stable, and the material processing situation of the three groups of data is maintained at about 24 times per hour.

## 4. Considering Only the Failure of One-Process Material Processing

### 4.1. Establishing a Fault Model during Material Processing

- (1) Four stages: the first stage comprises the first CNC and the second CNC. The second stage comprises the third CNC and fourth CNC. The third stage comprises the fifth CNC and sixth CNC. The fourth stage comprises the seventh CNC and eighth CNC.

- (2) Selecting state  $S_k$ :

$$S_k = (\text{CNC}\#k, \text{CNC}\#k + 1). \quad (17)$$

- (3) Determining strategies  $\mu_k(s_k)$ : RGV units of movement are represented for 1, 2, 3 or -1, -2, -3.

- (4) Decision value  $\nu_k(S_k)$ : working hours of RGV = movement time of RGV + loading, cleaning, and discharging time of CNC#k and CNC#k + 1

- (5) Writing the state transition equation:

$$S_{k+1} = T_k[S_k, \mu_k(s_k)]. \quad (18)$$

- (6) Establishing index function  $f_{1k}(S_k)$ : RGV working hours from the initial moment to the end of this phase.

- (7) Establishing objective function:

$$\text{Max } f_{1k}(S_k) = \sum_k \nu_k(S_k). \quad (19)$$

After eight hours of constraint, it eventually stops at a certain stage.

**4.2. Material Processing Fault Model of One Process for Solution.** From the establishment of the model, the model is the same as that without failure, but the algorithm should be changed when using MATLAB software.

Specific practice is to set up a random number producer, assigned to each processing table by equal probability. When the RGV turns to the faulty processing table, consider only the RGV movement time, not the RGV loading, unloading, and cleaning material time. For example, 1% probability is generated by the number of intervals setting [1,100]. Assuming a number like 50, you can specify which CNC fails.

According to the established cyclic dynamic planning model, the Floyd algorithm uses MATLAB software, and the obtained data are given in Table 9.

Table 9 shows that there are four failed CNCs during processing one material process.

TABLE 6: The working time of the material processing.

Material serial number	CNC number in process 1	Starting time	Closing time	CNC number in process 2	Starting time	Closing time
197	1	27685	28250	2	28285	28850
198	3	27768	28333	4	28398	28963
199	5	27881	28446	6	28511	29076
200	7	27994	28559	8	28624	29189
201	1	28250	28815	2	28850	
202	3	28333	28898	4	28963	
203	5	28446	29011	6	29076	
204	7	28559	29124	8	29189	
205	1	28815				
206	3	28898				
207	5	29011				
208	7	29124				

TABLE 7: RGV scheduling for 3 groups for two processes.

Group	RGV scheduling strategy in process 1	RGV scheduling strategy in process 2
Group 1	1—3—5—7	2—4—6—8
Group 2	1—3—5—7	2—4—6—8
Group 3	5—7—1—3	2—4—6—8

TABLE 8: Work efficiency for 3 groups for two processes.

Group	Number of processed materials	Work efficiency
Group 1	197	24.625
Group 2	198	24.75
Group 3	198	24.75

TABLE 9: Material processing failure in one process.

Material serial number during a failure	CNC number during a failure	Starting time during a failure	End of the time during a failure
53	7	3720	0
61	7	0	4874
266	6	19117	0
274	6	0	20271
362	6	26041	0
370	6	0	27195
380	8	27324	0
388	8	0	28478

## 5. Material Processing Fault Model for Two Processes

### 5.1. Establishing a Material Processing Fault Model for Two Processes

- (1) First of all, it is assumed that CNC# $m$  ( $m = 1, 2, 3, 4, 5, 6, 7, 8$ ) is installed as follows. CNC# $m$  ( $m = 1, 3, 5, 7$ ) was installed as the first procedure. CNC# $m$  ( $m = 1, 3, 5, 7$ ) was installed as the second procedure.
- (2) We set  $k$  phase variables,  $k = 1, 2, 3, 4, 5, 6, 7, 8$ .
- (3) The state variable  $S_k^1$  is set to CNC# $k$  state. The state variable  $S_k^2$  is set to CNC# $k + 1$  state.

- (4) Policy variables are  $\mu_k^1(s_k^1)$  and  $\mu_k^2(s_k^2)$ .  $\mu_k^1(s_k^1)$  means that the RGV unit of movement is 1,2,3.  $\mu_k^2(s_k^2)$  means that the RGV unit of movement is -1, -2, -3.
- (5) Decision variables are  $\mu_k^1(s_k^1)$  and  $\mu_k^2(s_k^2)$ .

RGV working hours = RGV movement time + CNC# $k$  and CNC# $k + 1$  feeding, cleaning, and discharging time.

- (7) State transfer equation:

$$\begin{aligned}
 S_{k+1}^1 &= T_k [S_k^1, \mu_k^1(S_k^1)], \\
 S_{k+1}^2 &= T_k [S_k^2, \mu_k^2(S_k^2)].
 \end{aligned}
 \tag{20}$$

TABLE 10: Material processing failure in the two processes.

Material serial number during a failure	CNC number during a failure	Starting time during a failure	End of the time during a failure
90	3	10180	0
94	3	0	0
98	3	0	11557
211	5	24060	0
215	5	0	0
219	5	0	25437
86	4	10239	0
90	4	0	0
94	4	0	11616
207	6	24119	0
211	6	0	0
215	6	0	25496

(8) Setting up the index function  $f_{1k}(S_k^1, S_k^2)$ :  $f_{1k}(S_k^1, S_k^2)$  represents RGV working hours from inception to the end of the phase.

(9) Establishing objective function:

$$\text{Max } f_{1k}(S_k^1, S_k^2) = \sum_k [v_k^1(S_k^1) + v_k^2(S_k^2)]. \quad (21)$$

**5.2. Solution of the Material Processing Fault Model for the Two Processes.** According to the established cyclic dynamic planning model and calculated using MATLAB software by the Floyd algorithm, the results are given in Table 10.

According to the table, when processing two material processes, there are four failed CNCs, each with three CNC numbers.

## 6. Summary

The cyclic dynamic planning model with the longest running time and the dynamic planning model are established in the paper. Using the most short circuit idea, the longest running time is transformed to find the optimal moving path of the track car.

## Data Availability

No data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] Y. Wu, Y. Liu, D. Zhang, and Z. Han, "Study on the dynamic scheduling of RGV based on genetic algorithms," *Lift Transport Machinery*, vol. 6, pp. 20–23, 2012.
- [2] Y. Liu, *Study on the Dynamic Scheduling of RGV Based on Genetic Algorithms*, Hefei University of Technology, Hefei, China, 2012.
- [3] H. Chen and Q. Sun, "Study on the linear reciprocating 2-RGV system scheduling based on the TS algorithm," *Industrial Engineering and Management*, vol. 20, no. 05, pp. 80–88, 2015.
- [4] H. Chen, "Model and algorithm for 2-RGV scheduling problems based on the partition method," *Industrial Engineering and Management*, vol. 19, no. 6, pp. 70–77, 2014.
- [5] Z. Zha, J. Li, and R. Sheng, "Application of intelligent translation rail guide vehicle," *Robotics Technology and Application*, vol. 5, pp. 42–43, 2017.
- [6] H. Gu, P. Zou, and W. Xu, "Self-learning algorithm for optimizing the scheduling problem of the circular shuttle vehicle," *The Theory and Practice of System Engineering*, vol. 33, no. 12, pp. 3223–3230, 2013.
- [7] A. Shi, L. Li, S. Xue, and B. Shao, "Research on the dynamic scheduling problem of intelligent RGV-CNC," *Computer and Digital Engineering*, vol. 49, no. 04, pp. 760–765, 2021.
- [8] S. Wang, "The RGV dynamic scheduling model based on the Memetic algorithm," *Journal of Shanxi Datong University (Natural Science Edition)*, vol. 37, no. 01, pp. 37–40, 2021.
- [9] B. Jing, "Dynamic scheduling strategy for smart RGV," *Electronic Testing*, vol. 21, pp. 64–65, 2020.
- [10] W. Wei, Z. Yang, H. Xie, and Y. Hu, "Intelligent RGV dynamic scheduling of the multi-objective planning model," *Computer Applications and Software*, vol. 37, no. 04, pp. 178–185, 2020.
- [11] L. Zhou, Y. Zeng, G. Xie, and J. Li, "Intelligent RGV dynamic scheduling strategy based on multi-objective optimization," *Automation Applications*, vol. 6, pp. 55–58, 2020.
- [12] Y. Li, L. Cai, R. Jia, Y. Zhang, and X. Zhu, "Intelligent RGV dynamic scheduling strategy based on the DP-FCFS algorithm," *Journal of Chengdu Institute of Technology*, vol. 23, no. 02, pp. 48–53, 2020.
- [13] Y. Niu, "Study on the intelligent RGV scheduling strategy for two processes," *Modern Trade Industry*, vol. 41, no. 17, pp. 194–196, 2020.
- [14] D. Wei, E. Hao, and Y. Wu, "Two-process processing policies for smart RGV," *Electronic Devices and Information Technology*, vol. 4, no. 03, pp. 78–79, 2020.
- [15] S. Li, "Dynamic scheduling strategy for intelligent RGV based on a greedy algorithm," *Scientific and Technological Innovation and Application*, vol. 6, pp. 16–17, 2020.
- [16] J. Li, "Intelligent RGV dynamic scheduling policy for the FMS two-process system considering random failures," *Industrial Control Computers*, vol. 33, no. 02, pp. 117–120, 2020.
- [17] Y. Du, M. Feng, J. Li, and R. Fang, "Dynamic scheduling strategy for smart RGV," *Shantou University Journal (Natural Science Edition)*, vol. 35, no. 01, pp. 54–65, 2020.
- [18] X. Yao, Y. Liu, and J. Liu, "Dynamic scheduling strategy based on a single-process intelligent processing system," *Ordnance Automation*, vol. 39, no. 01, pp. 35–39, 2020.



- [19] X. Qiao, "Dynamic scheduling strategy for smart RGV," *Journal of Nanning Normal University (Natural Science Edition)*, vol. 36, no. 04, pp. 40–44, 2019.
- [20] Q. Zeng and F. Qi, "RGV dynamic scheduling policy based on computer simulation failure," *Engineering Construction and Design*, vol. 19, pp. 165-166, 2019.
- [21] M. Yuan, I. Yang, R. Sun, and P. Lv, "Dynamic scheduling strategy for RGV in a circular intelligent processing system," *Value Engineering*, vol. 38, no. 27, pp. 43–45, 2019.
- [22] M. Zhang, M. Chen, X. Wang, and L. Zhuang, "Research on intelligent RGV dynamic scheduling strategy for multi-objective optimization and simulation," *China New Communications*, vol. 21, no. 18, p. 113, 2019.
- [23] T. Yao and N. Xu, "Dynamic scheduling strategy for smart RGV," *Computer Knowledge and Technology*, vol. 15, no. 22, pp. 270–273, 2019.
- [24] Y. Zhou, Y. Li, M. Zuo, and Y. Li, "Research on the dynamic scheduling strategies of intelligent RGV," *Journal of Qiqihar University (Natural Science Edition)*, vol. 35, no. 05, pp. 72–77, 2019.
- [25] Z. Zhu and X. Meng, "Modeling and optimization of RGV dynamic scheduling based on particle group algorithm and simulation technology," *Mechanical and Electronic Information*, vol. 18, pp. 98–101, 2019.