

# Dynamic Scheduling Strategy of Intelligent RGV Based on Multi-layer Predictive Optimization

**Yunhui Zeng**

College of Intelligent Science and Engineering, Jinan University, No. 206, Qianshan Road, Xiangzhou District, Zhuhai City, Guangdong Province, China

**Yilin Chen**

College of Intelligent Science and Engineering, Jinan University, No. 206, Qianshan Road, Xiangzhou District, Zhuhai City, Guangdong Province, China

**Hongfei Guo**\*

College of Internet of Things and Logistics Engineering, Jinan University, No. 206, Qianshan Road, Xiangzhou District, Zhuhai City, Guangdong Province, China

**Li Huang**

College of Intelligent Science and Engineering, Jinan University, No. 206, Qianshan Road, Xiangzhou District, Zhuhai City, Guangdong Province, China

**Wenjuan Hu**

College of Intelligent Science and Engineering, Jinan University, No. 206, Qianshan Road, Xiangzhou District, Zhuhai City, Guangdong Province, China

## Abstract

This paper takes the dynamic scheduling of intelligent RGV as the research object and explores the problem of materiel machining of intelligent RGV for one and two procedures. In the process of establishing a materiel machining operation model for one procedure, firstly, the banker algorithm is used to provide a scheduling strategy for the RGV, dynamically predict the evolution process of the resource allocation process, and determine the order in which the CNC performs the task. Then, the non-preemptive least laxity first concept is introduced to improve the utilization rate of CNC and minimize the time for the computer machine tools CNC to wait for response. In order to simplify the calculation and improve the feasibility, based on the main idea of the banker algorithm, the evolution of the situation is only carried out in three levels, which makes the algorithm calculate moderately and has certain reference value for the prediction of the evolution process. Moreover, in the process of establishing a materiel machining operation model for two procedures, the bat algorithm is used to establish the model from the macroscopic perspective, and finally the dynamic scheduling strategy of RGV is obtained. In this paper, the dynamic scheduling strategy of intelligent RGV established for the materiel machining for one and two procedures provides a theoretical basis for the development of RGV dynamic scheduling strategy in the actual production process.

**Keywords:** Multi-layer predictive optimization; RGV dynamic scheduling; Banker algorithm; The least laxity; Bat algorithm.

CC BY: [Creative Commons Attribution License 4.0](https://creativecommons.org/licenses/by/4.0/)

## 1. Introduction

Since the industrial revolution in the 19th century, with the rapid development of control engineering, mechatronics, information technology and other fields, the logistics system has begun to develop into automation, unmanned and intelligent. The Rail Guided Vehicle (RGV) integrates various high-tech technologies. It can be used as an auxiliary equipment for a three-dimensional warehouse or as a stand-alone system and it is widely used in modern logistics systems [1]. In the process of industrial production involving RGV system, because of the time and resource are shared between different products or the production of multiple batches of the same product, conflicts are easy to occur. Therefore, organizational production must be arranged carefully and time as well as equipment should be allocated according to market demand or production tasks, achieving RGV system production process scheduling optimization to improve production efficiency and economic efficiency [2]. At present, the research on RGV mainly focuses on the design, control, dispatching rules and path selection of RGV. The research literature on the overall scheduling of RGV is rare. Liu et al. proposed the path division mode and path integration mode of double RGV transportation systems and their logistics planning, using AutoMod simulation software to model these two operation modes [3]. Nie Feng et al. proposed a nearest neighbor algorithm based on the principle of sequential queuing for a RGV multi-functional shuttle which was designed for an automated three-dimensional library and combined with the strategy setting method to optimize operating scheduling in order to improve the efficiency of operating storage [4]. Based on a disaster relief materiel automated warehouse, Chen Hua established a mixed integer linear programming model for double RGV scheduling problem based on region partition, and proposed a hybrid genetic algorithm to minimize the total outgoing time of all materiel [5]. Luo Jian and Wu Changqing proposed an improved quantum particle swarm optimization (QPSO) algorithm to improve the efficiency of RGV

system in automatic vehicle access system. In this algorithm, Gauss mutation operator was introduced to overcome the shortcoming that it is easy to fall into local optimum [6]. J Lee and R Maneesavet put forward five RGV dispatching rules under the background of manufacturing system, and evaluated the five rules and three common dispatching rules through simulation experiments [7]. The literatures above study scheduling problems separately from traditional rule scheduling or modern intelligent algorithms. The advantages of these methods are that they can get a feasible solution. The disadvantage of these methods is that they can't solve the problems accurately and optimize the dynamic scheduling process of RGV, so they can't effectively improve the production efficiency.

On the basis of the above research, this paper takes two operating scenarios of intelligent RGV machining system as the research object, introduces non-preemptive least laxity first concept, establishes RGV scheduling models by using banker algorithm and bat algorithm, and finally proposes an intelligent RGV dynamic scheduling strategy based on multi-layer predictive optimization. This paper innovatively applies the bat algorithm, which is often used to solve continuous domain problems, to discrete problems. This new intelligent evolutionary algorithm is used to solve the optimal solution of RGV target location. Based on the idea of banker algorithm, three levels of search algorithm are used in motion decision-making, which can prevent the solution that is local optimum but not global optimum and solve the dynamic scheduling problem of RGV well.

## 2. The Mathematical Model

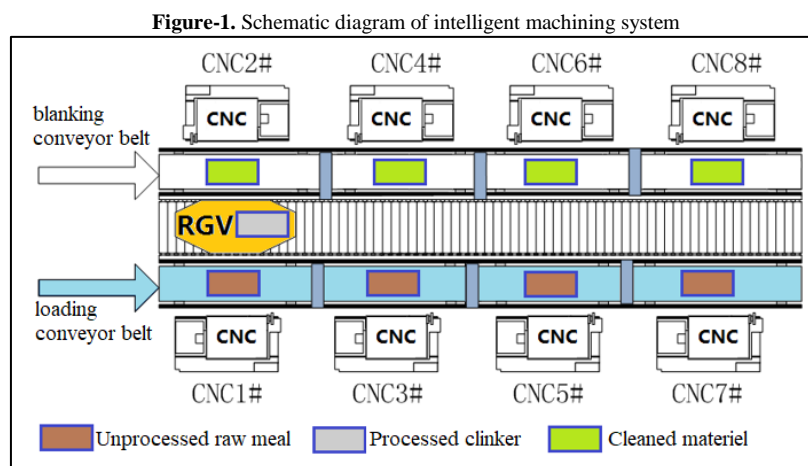
### 2.1. Data Acquisition and Assumptions

The original data of this research object is from the topic of 2018 National College Students Mathematical Modeling Contest (China Society for Industrial and Applied Mathematics, 2018). To facilitate analysis and understanding, we make the following assumptions about the model:

- (1) Assumed that all machine equipment and accessories including computer machine tools CNC are fault-free;
- (2) Assumed that the processor chip of the RGV of the orbital automatic guided vehicle is good, and the optimal strategy can be quickly calculated;
- (3) Assumed that there is no stoppage in each shift;
- (4) Assumed that the processed materiel is sufficient;
- (5) Assumed that the power energy is sufficient during the processing.

### 2.2. Operating Scenarios of Intelligent Machining System

The simple intelligent machining system studied in this paper consists of eight CNC machine tools, one orbital automatic guided vehicle RGV, one RGV linear track, one loading conveyor belt and one blanking conveyor belt. A simplified diagram is shown in Figure-1. In this intelligent machining system, RGV, the smart car which is unmanned and free to run on fixed track, is responsible for production scheduling to complete the work. It can automatically control the moving direction and distance according to the instructions as well as complete the tasks such as loading, blanking and cleaning materiel through its own mechanical arm, two mechanical grippers and cleaning groove [8].



The operating scenarios of the intelligent machining system can be roughly divided into the following two types:

- (1) Materiel machining for one procedure: each CNC is equipped with the same tool, and the materiel can be processed on any CNC;
- (2) Materiel machining for two procedures: the first and second procedure of each materiel can only be processed by two different CNC in sequence;

This paper will establish a model for the two operating scenarios of intelligent machining system to obtain the best dynamic scheduling strategy of RGV.

### 3. Establishment of RGV Scheduling Model

Through the intelligent machining system, the factory further rationally allocates and optimizes internal and external resources, shortens manufacturing cycle and reduces production cost. The production scheduling is precisely the means for effectively configuring and optimizing resources, which can reasonably arrange the processing order of the operations and optimize the performance indicators of the relevant systems under the constraints of various systems.

The problem studied in this paper involves the establishment of a linear flow scheduling model. The ultimate goal of reasonable scheduling is to minimize the total task time of eight CNCs in each shift, so that the CNC achieves the highest utilization. In the production machining system, the banker algorithm [9] is used to provide a scheduling strategy for the RGV, determining the execution order of the CNC. The algorithm dynamically predicts the evolution of the situation in the process of resource allocation, and can effectively avoid the deadlock problem [10].

#### 3.1. Banker Algorithm

The banker algorithm is a typical algorithm designed by computer scientist Azge Dijkstra from Dutch to solve the deadlock problem of resource allocation in the operating system. In the decision-making process, the banker algorithm needs to know in advance the maximum demand for resources of each active process in the whole life process. This static information and the dynamic information of the resource allocation during the actual operation process together determine whether all active processes can be performed in a certain order.

Suppose there are  $n$  processes  $P_1, P_2, \dots, P_n$ , sharing  $m$  class resources  $R_1, R_2, \dots, R_m$ .

$$\text{Available}=(a_1, a_2, \dots, a_m)$$

(1) Available resource vector:

The number of elements in the vector is  $m$ . Each element represents the number of units available for a certain type of resource. The initial value is the total number of units of the resource configured in the system, and its value changes dynamically with the allocation and recycling of the resource.

(2) Maximum demand matrix: Max

$\text{Max}(i, j)=k$ , this matrix indicates that the process  $P_i$  requires a total of  $k$  units of  $R_j$  resources.

$\text{Max}(i, j)=k$ , which indicates that the process  $P_i$  requires a total of  $k$  units of  $R_j$  resources.

(3) Allocation matrix: Allocation

$\text{Allocation}(i, j)=k$ , which indicates that process  $P_i$  already occupies a total of  $k$  units of  $R_j$  resources.

(4) Demand Matrix: Need

$\text{Need}(i, j)=k$ , which means that the process  $P_i$  needs to complete  $k$  units of  $R_j$  resources, that is, the remaining resource requirements.

The following relationships exist between the above three matrices:

$$\text{Need}(i, j)=\text{Max}(i, j)-\text{Allocation}(i, j) \quad (1)$$

Suppose  $\text{Request}_i$  is the request vector of the currently running process  $P_i$ . There is  $\text{Request}_i[j]=k$ , which indicates that process  $P_i$  is currently requesting  $k$  units of  $R$  resources. When process  $P_i$  issues a resource request, the system checks as follows:

- 1) If  $\text{Request}_i \geq \text{Need}[i]$ , the request is an illegal request and is processed accordingly because it requires more resources than it has announced, otherwise, the next step is taken.
- 2) If  $\text{Request}_i \geq \text{Available}$ , it means that the resources available to the system cannot meet the process requirements and block the process; otherwise, the next step is taken.
- 3) The system tentatively allocates the required resources to the process  $P_i$ , that is, does not perform actual resource allocation, but merely modifies the relevant data structure.

$$\text{Available} = \text{Available} - \text{Request}_i \quad (2)$$

$$\text{Allocation}[i] = \text{Allocation}[i] + \text{Request}_i \quad (3)$$

$$\text{Need}[i] = \text{Need}[i] - \text{Request}_i \quad (4)$$

- 4) Perform a security detection algorithm to check whether the system is in a safe state after the resource allocation. If it is safe, it will officially allocate resources to process  $P_i$ . Otherwise, the trial allocation will be invalidated, the original resource allocation state will be restored, and process  $P_i$  will continue to wait [11].

### 3.2. Establishment of Material Machining Operation Model for One Procedure

For the material machining operation of one process, it is not necessary to consider the order of execution of the process, which is the most basic and simple operation situation. When establishing the RGV dynamic scheduling model for this scenario, the ultimate goal to be achieved is to minimize the total task shelving time of 8 CNCs in each shift. If the basic idea of the banker algorithm is adopted, in the process of establishing the model, the CNC is equivalent to the process in the operating system, and the task is equivalent to the resources in the operating system. Different from the banker algorithm in the operating system, in the algorithm, the maximum demand for resources by the process is the CNC needs for resources from the RGV, that is, the set of instructions issued by the RGV, which is a dynamic variable related to the state of the system.

In order to improve the utilization rate of CNC and shorten the time for CNC machine tool to wait for response, the non-preemptive minimum slack priority concept is introduced in the linear flow scheduling system [12]. That is, the priority of the task is determined according to the degree of slack of each CNC. The higher the urgency of the task, the higher the priority given to the task, so that it is prioritized.

In the actual production process, in order to achieve higher output, CNC machine tool CNC should immediately issue a new task after completing a task, that is, the time when the CNC completes the task = the time when the CNC sends out the new task. Finally, the idle time of the CNC is reduced, and the utilization of production resources is maximized. On this basis, the criterion for the non-preemptive minimum slack priority concept is the urgency of demand for each CNC task, which is quantified as:

The task laxity of a CNC = the time when the CNC completed the task - the time when the RGV reached the location of the CNC, which is equivalent to: The task laxity of a CNC = the time when the CNC sends a new task demand - the time when the RGV arrives at the CNC location.

The description is made using a mathematical language as follows:

$$S_n = T_{new\_n} - T_{arrived} \quad (5)$$

Where  $S_n$  is the slack of the demand for the CNC task of  $n$ ,  $T_{new\_n}$  is the time required for the CNC to issue a new task, and  $T_{arrived}$  is the travel time taken by the RGV to reach the location of the designated CNC.

During the execution of a process by RGV, after RGV completes a loading and unloading operation for a CNC, it will rotate the arm to move the clinker from one robot to the top of the cleaning tank for cleaning operation and analyze the machining execution process.

The time when a CNC completes an old task, that is, the time required to issue a new task, is an accumulation process, including the task execution time received after starting the work, the total time shelved the task waits for the response, and the loading and unloading time, the mathematical language is expressed as follows:

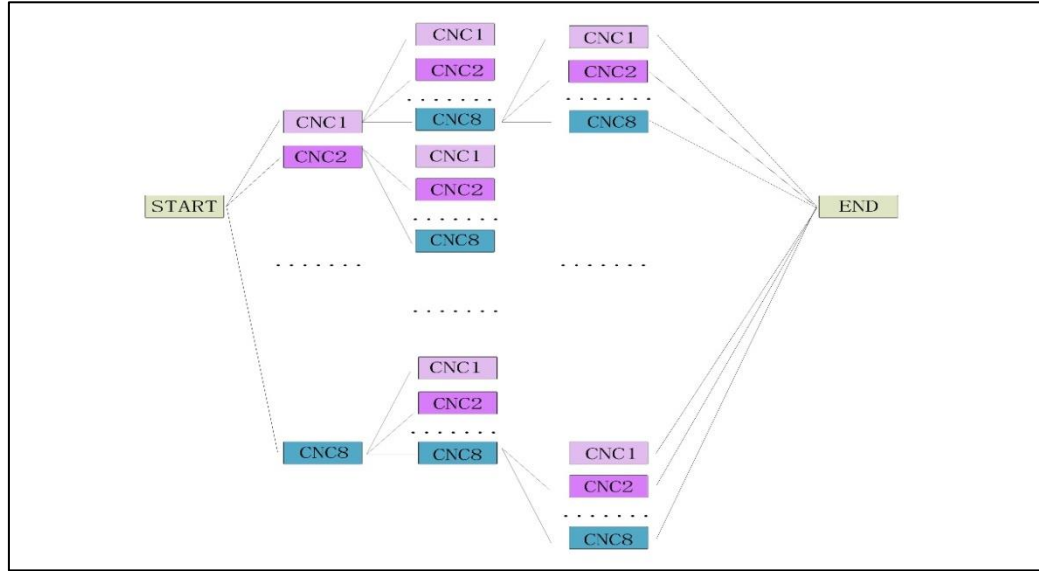
$$T_{new\_n} = \sum_{i=1}^{n-1} T_{task\_n} + \sum_{i=1}^{n-1} T_{wait\_n} + \sum_{i=1}^{n-1} (T_{top} + T_{down}) \quad (6)$$

Where  $T_{task\_n}$  is the time when the demand signal is sent by the task  $n$ ,  $T_{wait\_n}$  is the time when the task  $n$  waits for the response and is put on hold,  $T_{top}$  is the loading time of the task, and  $T_{down}$  is the blanking time of the task. The time when the RGV reaches the specified CNC location is also an accumulation process, including the waiting time after the RGV arrives at the specified position in advance, the time the RGV cleans the clinker, the time for loading and unloading the CNC machine tool, and the moving time of the RGV, expressed in mathematical language as follows:

$$T_{arrived} = \sum_{i=1}^{n-1} T_{move\_n} + \sum_{i=1}^{n-1} T_{wait\_RGV\_n} + \sum_{i=1}^{n-1} (T_{top} + T_{down} + T_{wash}) \quad (7)$$

Where  $T_{move\_n}$  is the travel time taken by the RGV to reach the location of the specified CNC of the task  $n$ ,  $T_{wait\_RGV\_n}$  is the time that the RGV waits to arrive at the specified position in advance when the task  $n$  is executed,  $T_{top}$  is the loading time of the task,  $T_{down}$  is the blanking time of the task, and  $T_{wash}$  is the time during which the RGV cleans the clinker.

Based on the judgment of task laxity, the banker algorithm is used to dynamically predict the evolution of the process in the process of resource allocation to reduce the robustness of decision making [13]. In the process of the process, the evolution of the situation will continue to push, into the state of infinite loop, the amount of calculations will increase, Resulting in the algorithm is not desirable. Therefore, in order to simplify the calculation and improve the feasibility, based on the main idea of the banker algorithm, the evolution of the situation is only carried out in three levels. This choice makes the calculation of the algorithm moderate, and has a certain reference value for the development of the situation. See Figure-2 for details.

**Figure-2.** Three-level Derivation Prediction Based on Banker Algorithm

The first level: RGV internal chip calculates the time when the RGV car arrives at each CNC position and the time when each CNC completes the current task. The task laxity of eight CNCs are calculated by traversing according to formula (5).

The second level: traversing the eight choices that appear in the first level, and calculating the time when the RGV car arrives at each CNC location and the time when each CNC completes the current task through the RGV internal chip again. The task laxity of eight CNCs are calculated by traversing according to formula (5).

The third level: traversing the choices that appear in the second level, and calculating the time when the RGV car arrives at each CNC location and the time when each CNC completes the current task through the RGV internal chip again. The task laxity of eight CNCs are calculated by traversing according to formula (5).

After three advancement calculations,  $8 \times 8 \times 8$  task laxity is obtained, and the 512-task laxity obtained are compared to take the minimum laxity. The CNC selected at the first level is obtained by inversely deriving the source of the slack and this CNC is the object of resource allocation for RGV.

### 3.3. Establishment of Materiel Machining Operation Model for One Procedure

The difference between the materiel machining operation of the two procedures and the materiel machining operation of one procedure is that the processing procedure of each materiel includes two processes, and is processed by two different CNCs in sequence. The machining operations of the two procedures are larger than the materiel machining of one procedure, and the difficulty of task scheduling increases.

What has changed in hardware is that eight CNCs have been divided into two types, one for processing one and the other for processing two. The ratio of the number of CNC machine tools arranged for the two procedures depends on the proportion of time between completion of procedure 1 and completion of procedure 2, the layout should follow the principle of even distribution.

If the materiel machining operation of one procedure is continued, the computational complexity will increase sharply, and the scheduling algorithm of the materiel machining operation in one procedure is at the lower level, that is the resources are allocated to the task, and the problem is solved in microscopic vision. However, in the materiel machining of the two procedures, it is impossible to simulate the assignment process of the task resources from the microscopic visual, so it needs to be transformed into the model establishment on the macroscopic vision.

In order to analyze from macroscopic vision, the bat algorithm [14] was introduced to model the materiel machining operations of the two procedures. Firstly, the orbital automatic guided vehicle RGV is analogized to a bat; Secondly, the size of the objective function value is used to judge whether the current RGV is located, that is, whether shelving time of the total task can be the shortest; Finally, the process of RGV moving the detection target is analogized to the process of searching for the optimal solution of the objective function, thereby establishing a scheduling model.

#### 3.1.1. Simulating the Movement of Bats

In the process of simulation, the orbital automatic guided vehicle RGV is compared to a bat, then the position  $X$  and velocity  $V$  of the bat in the one-dimensional search space are updated.

The update method of the speed  $V_t$  of the bat at time  $t$  is:

$$V_t = V_{t-1} + (X_{t-1} - X^*)f \quad (8)$$

In the formula,

$$f = f_{\min} + (f_{\max} - f_{\min})\beta \quad (9)$$



Among them,  $\beta \in [0,1]$  is a random vector subject to uniform distribution,  $f$  is the frequency of the bat,  $X^*$  is the optimal solution of the current global position.

During the production operation, each CNC machine tool has two working states, namely idle state and task waiting response state, which can be represented by 0-1 binary code:

$$X = \begin{cases} 0 & \text{idle state} \\ 1 & \text{task waiting response state} \end{cases}$$

Combined with the speed  $V_t$  of the bat at time  $t$ , its position can be updated to:

$$X_t = X_{t-1} + V_t \quad (10)$$

Because the product  $(X_t - X^*)f_i$  is a speed increment, a factor can be fixed to adjust the speed change according to actual needs. In the actual problem solving, the value of  $f_i$  can be determined according to the size of the search range. There are eight CNC machine tools here, so it is initialized to  $f_{\min} = 0, f_{\max} = 8$ . Initially, each bat is randomly assigned a frequency by a uniform distribution between  $[f_{\min}, f_{\max}]$ .

Similar to other bionic algorithms, local search behavior can also be set in the search optimization process of simulated bats. For local search, the update formula for each bat position is as follows:

$$X_{\text{new}} = X_{\text{old}} + \varepsilon A_t \quad (11)$$

In the formula,  $\varepsilon \in [-1,1]$  is a random number,  $A_t$  is the loudness of the bat over a period of time.

(2) Loudness and pulse emission rate

The loudness  $A_t$  of the pulse and the rate of transmission  $r_t$  are updated as the iterative process progresses. Generally, in the process of approaching the prey, the loudness will gradually decrease, the rate of pulse emission will gradually increase. The update formula for the loudness  $A_t$  and the emission rate  $r_t$  of the pulse emission is as follows:

$$A_{t+1} = \alpha A_t \quad (12)$$

$$r_{t+1} = r_0 [1 - \exp(-\gamma t)] \quad (13)$$

among them,  $A_t$  is the impulse loudness of the bat at time  $t$ ,  $r_t$  is the pulse emission rate of the bat at time  $t$ , the constant  $\alpha$  is the pulse loudness attenuation coefficient, the constant  $\gamma$  is the pulse frequency increase coefficient [15].

In fact,  $\alpha$  is similar to the cooling factor in the cooling process in the simulated annealing algorithm [16]. It's not hard to find that for any  $0 < \alpha < 1, \gamma > 0$ , there is:

$$\lim_{t \rightarrow \infty} A_t = 0 \quad (14)$$

$$\lim_{t \rightarrow \infty} r_t = r_0 \quad (15)$$

In practice,  $\alpha = \gamma = 0.9$  is usually used.

Among them,  $A_0$  is the initial loudness of the bat,  $r_0$  is the initial pulse rate of the bat.

As long as the new solution is improved, the bat's loudness and pulse rate will be updated, which means that the bat will always fly toward the optimal solution, that is, RGV will allocate resources to the most suitable CNC.

## 4. Conclusion

This paper takes the dynamic scheduling of intelligent RGV as the research object and explores the two operating scenarios of materiel machining of intelligent RGV for one and two procedures. In the process of establishing a materiel machining operation model for one procedure, firstly, the banker algorithm is used to provide a scheduling strategy for the RGV, dynamically predict the evolution process of the resource allocation process, and determine the order in which the CNC performs the task. Then, the non-preemptive least laxity first concept is introduced to improve the utilization rate of CNC and minimize the time for the computer machine tools CNC to wait for response. Based on the banker algorithm, the evolution of the situation is only carried out in three levels to reduce the robustness of the decision, simplify the calculation and improve the feasibility. In addition, in the process of establishing a materiel machining operation model for two procedures, the bat algorithm is used to establish the model from the macroscopic perspective, and finally the dynamic scheduling strategy of RGV is obtained. This paper establishes algorithm models for RGV real-time monitoring and scheduling based on two operating scenarios of intelligent RGV machining system, providing a theoretical basis for the development of RGV dynamic scheduling strategy in the actual production process.

However, due to space limitations, there are some shortcomings in this paper, which are embodied in:

- (1) Three-layer traversing search is carried out with the idea of banker algorithm, which has a large amount of calculation and limits the response sensitivity of RGV.
- (2) The failure of the machine is not considered in this paper, but the machine will inevitably fail, so this model may not be applicable when the machine is frequently faulty.
- (3) Regarding the scheduling problem of the two procedures, it does not consider how to avoid the occurrence of deadlock, which may cause the machining flow enter a deadlock state, making the entire system paralyzed.
- (4) Due to the lack of relevant data, the model was not fully verified.

## References

- [1] Jiang, J., 2016. "Hybrid algorithm for rounding rail guided vehicle optimization scheduling." *Computer Engineering and Applications*, vol. 52, pp. 242–247.
- [2] Zhang, 2012. "Research on production scheduling problems in process industry based on particle swarm optimization algorithm." *Computer Engineering and Applications*, vol. 48, pp. 225–228.
- [3] Liu, C., 2010. "Operation policy research of double rail-guided vehicle based on simulation." In *International Conference on E-Product E-Service and E-Entertainment. IEEE*, pp. 1–4.
- [4] Nie, F. J., 2007. "Research on the optimal scheduling of RGV system." *Logistics Technology*, vol. 27, pp. 251–253.
- [5] Chen, H. J., 2014. "Model and algorithm for two rail-guided vehicles scheduling problem based on a partitioning approach." *Industrial Engineering and Management*, vol. 19, pp. 70–77.
- [6] Luo, J. J., 2011. "Modeling and optimization of RGV system based on improved QPSO." *Computer Integrated Manufacturing Systems*, vol. 17, pp. 321–328.
- [7] Lee, J. J., 1999. "Dispatching rail-guided vehicles and scheduling jobs in a flexible manufacturing system." *International Journal of Production Research*, vol. 37, pp. 111–123.
- [8] Zhou, Z. J., 2018. "Dynamic scheduling policy model of intelligent RGV." *China Southern Agricultural Machinery*, vol. 49, pp. 55–56+66.
- [9] Yong, J., 2017. "Searching feasible resources to reduce false-positive situations for resolving deadlocks with the Banker's algorithm in railway simulation." *Journal of Rail Transport Planning & Management*, vol. 7, pp. 50–61.
- [10] Xu Gang, J., 2004. "Deadlock problem in manufacturing systems " *Systems Engineering-Theory & Practice*, vol. 24, pp. 106–111.
- [11] Hou, G. J., 2006. "In-depth analysis of banker algorithms." *Journal of Weifang University*, vol. 6, pp. 46–48.
- [12] Zhang, 2011. "Least laxity first information scheduling algorithm of industrial ethernet ring network." *Journal of South China University of Technology(Natural Science Edition)*, vol. 39, pp. 76–80.
- [13] Shiv, K. J., 1968. "Robustness in sequential investment decisions." *Management Science*, vol. 15, pp. B-18–B-29.
- [14] Ma Xiangli, J., 2015. "Application of bat algorithm in vehicle routing problem of logistics distribution." *Mathematics in Practice and Theory*, vol. 45, pp. 80–86.
- [15] Ding, Z. J., 2017. "Cloud computing environment user task scheduling efficiency optimization simulation." *Computer Simulation*, vol. 34, pp. 298–301.
- [16] Zhang, 1997. "Fast simulated annealing algorithm and its application." *Oil Geophysical Prospecting*, vol. 32, pp. 654–660.