Research and Analysis of Intelligent RGV Based on Dynamic Scheduling Optimization Model

Zheng Wang¹, Zeyu Zhou², Jiawei Liu³

(¹School of Electromechanic Engineering, Hohai University, Changzhou Jiangsu China; ²School of Business Administration, Hohai University, Changzhou Jiangsu China; ³School of The Internet of Things Engineering, Hohai University, Changzhou Jiangsu China;)

About the author: Zheng Wang, Email: 1428558627@qq.com, undergraduate; Zeyu Zhou, undergraduate; Jiawei Liu, undergraduate;

Abstract  Aiming at the problem of fault-free scheduling in single process, a single process fault-free dynamic scheduling algorithm is proposed by giving several scheduling principles and several lemma. Aiming at the problem of fault-free scheduling of two-process, this paper puts forward the definition and determination principle of "Balance Principle", "Rare CNC" and "m-Center Principle", and gives the algorithm of two-process fault-free dynamic scheduling based on these principles and "single-process fault-free dynamic scheduling algorithm". Aiming at the problem of single process with fault scheduling and the problem of two-process fault scheduling, the number of faulty workpieces is determined according to 1% of the number of workpieces obtained by the corresponding fault-free dynamic scheduling algorithm, and randomly assigned to CNC and the time period of failure, using the finished time of CNC and the working time of RGV, The corresponding fault dynamic scheduling algorithm is given by using the corresponding trouble-free dynamic dispatching algorithm. Finally, this paper validates the operation example, and gives the idea and flow of the optimized all-intelligent RGV dynamic scheduling model algorithm.

1. Research background

Production planning and scheduling is an important means for enterprises to improve production efficiency, reduce production costs and improve market competitiveness. This requires scheduling decision tools to help business decision makers make the right and reasonable decisions in less time. Among them, shop scheduling is a kind of production resource allocation problem which satisfies the constraints of task configuration and process, duration and so on.

The research on the workshop scheduling problem has always been a hot topic in the theoretical circle, but because many workshop scheduling problems belong to NP problem, even some small-scale scheduling problems are difficult to obtain the optimal solution. With the development of computer technology, some intelligent algorithms, such as genetic algorithm and ant colony algorithm, have gradually become the hotspot to solve the problem of shop scheduling, and intelligent algorithm will also be the mainstream of the development of scheduling algorithm in the future.

2. Primarily

A specific intelligent machining system, consisting of 8 computer CNC machine tools (CNC), 1 orbital Automatic guide vehicles (RGV), 1 RGV linear tracks, 1 feeding conveyor belts, 1 feeding conveyor
belts and other ancillary equipment. RGV is a driverless, smart car that can run freely on a fixed track. It can automatically control the direction and distance of movement according to the instructions, and bring its own mechanical arm, two manipulator claws and material cleaning tank. It can complete the up and down materials and cleaning material and other work tasks.

The operating flow of RGV is as follows:

1. After the intelligent machining system is powered on and started, the RGV is in its initial position and all CNC is idle;
2. When the CNC is idle and the processing operation is completed, it will send the demand signal to RGV;
3. Once received the demand signal of a CNC, RGV will determine the order of the CNC’s loading and unloading work, and sequentially serve for it;
4. After completing a loading and unloading operation, the cleaning operation will follow up by RGV;
5. After completing a job task, RGV waits in place if no other job instructions are received.
6. The system repeats (3) to (5) over and over again until the system stops working and RGV returns to its original position.

This article will analyze the following three specific situations:

1. Material processing operations of single process, each CNC is installed with the same tool and materials can be processed on any CNC;
2. Material processing operation of two processes, the first and second processes of each material are processed by two different CNC in turn;
3. CNC may fail during processing (1% is the given probability of failure), each troubleshooting (manual processing, unfinished material scrapping) needs 10-20 minutes; Once troubleshooting finished, CNC joins the job sequence immediately. It is required to consider the material processing operation of single process and two processes respectively.

3. Model establishment and solution

3.1 Semi-intelligent single process fault-free dynamic scheduling Model

3.1.1 Semi-intelligent RGV Dynamic scheduling scheme operating principle

Lemma 1: The position of the CNC-a is \( S_a = \left\lceil \frac{a}{2} \right\rceil \), \( S_a \) is the position of CNC ordered with number a (\( S_a \) valued 1,2,3,4)), which is the upper integer of the pair.

Definition 1: If the RGV is in the middle of CNC-a and the CNC on the other side of it, the position of the RGV is defined as \( S_g = S_a \). (\( S_g \) is the position of RGV (valued as 1,2,3,4)).

Lemma 2: When requirement instructions are received from CNC-a, CNC-b, if \( (t_{d_{a,b}} + SS_a) > (t_{d_{a,b}} + SS_b) \) (\( t_n \) is the time requires RGV to move \( n \) length unit), the RGV moves firstly to CNC-b for the job; if \( (t_{d_{a,b}} + SS_a) \leq (t_{d_{a,b}} + SS_b) \), RGV firstly moves to the CNC-a for the work.

Proof: If RGV moves to CNC-b and then to CNC-a, the total time is \( T_a = t_{d_{a,b}} + SS_a + C + t_{d_{a,b}} \) (\( C \) is the time requires to complete the cleaning operation of an item), and if RGV moves to CNC-b and then to CNC-b, the total time is \( T_b = t_{d_{a,b}} + SS_b + C + t_{d_{a,b}} \). If \( T_a > T_b \), that is \( (t_{d_{a,b}} + SS_a) > (t_{d_{a,b}} + SS_b) \), the RGV firstly moves to the CNC-b for the work, otherwise, RGV firstly moves to the CNC-a.

Definition 2: Distance between RGV and CNC-a is defined as \( d_{R,a} = |S_R - S_a| \); Distance between
CNC-a and CNC-b is defined as \( d_{a,b} = |S_a - S_b| \).

A specific RGV operating principle is following:

**Principle 1:** When only 1 CNC requirements instructions are received, RGV moves to the CNC and carries out up and down materials and cleaning operations.

**Principle 2:** When 2 CNC requirements instructions (CNC-a, CNC-b) are received, let the distance between RGV and these two CNC machines be \( d_{R,a} \) and \( d_{R,b} \).

1. If \( d_{R,a} = d_{R,b} \), RGV moves to a CNC with an odd number first, and work on it; otherwise, randomly serve any of the CNCs.
2. If \( d_{R,a} > d_{R,b} \), the job is scheduled according to the following conditions:
   - **Case 1:** If b is even, a is odd, and \( T_a > T_b \), then RGV first moves to CNC-b and carries on the loading and unloading, cleaning and other operations; If \( T_a < T_b \), RGV first moves to CNC-a and carried out operations; If \( T_a = T_b \), RGV moves first to an odd number of CNC and then work.
   - **Case 2:** If b is an odd number or both a, b are even, RGV moves to CNC-b first.

**Principle 3:** When 3 CNC issuance requirements are received (CNC-a, b, c), let the distance between RGV and these 3 CNC machines be \( d_{R,a} \), \( d_{R,b} \), and \( d_{R,c} \).

1. When the farthest CNC number c is even, remove the CNC, and then the problem transforms into Principle 2.
2. When the farthest CNC number c is odd:
   - **Case 1:** If a or b or both a and b are odd, then eliminate the CNC-c, and then the problem transforms into Principle 2.
   - **Case 2:** If both a and b are even, when \( d_{R,a} < d_{R,b} \), discard the farther CNC-b, according to principle 2 to determine the order of RGV processing CNC-a and CNC-c; When \( d_{R,a} = d_{R,b} \), discard the CNC that is not on the same side with CNC-c, and then according to Principle 2 decided the order for RGV to deal with CNC-c and the CNC of the same side (one of CNC-a, CNC-b).

**Principle 4:** When 4 or 5 CNC issuance requirements are received, RGV moves first to the closest CNC.

**Principle 5:** When 6 CNC issuance requirements are received:
1. If there is a CNC demand order in the location of RGV, such CNC is served first.
2. If there is no demand order from the CNC location of the RGV, select any CNC that is the nearest number with the RGV distance to serve.

**Principle 6:** When 7 CNC issuance requirements are received, RGV serves the local CNC firstly.

**Principle 7:** When 8 CNC issuance requirements are received, RGV serves the local CNC with an odd position number firstly.

### 3.1.2 Semi-intelligent RGV algorithm

According to the principles given above, the dynamic scheduling solution algorithm of semi-intelligent RGV is presented:

**Semi-intelligent single-process trouble-free algorithm**

**Step 1:** After the intelligent machining system is energized, RGV in the initial position in the middle of CNC1 and CNC2, the initial working time of the RGV, that is \( \text{time} = 0 \), RGV continues the first round of job scheduling in the order of

\[ \text{CNC1} \rightarrow \text{CNC2} \rightarrow \text{CNC3} \rightarrow \text{CNC4} \rightarrow \text{CNC5} \rightarrow \text{CNC6} \rightarrow \text{CNC7} \rightarrow \text{CNC8}. \]

**Step 2:** Calculate the working time of the current RGV, \( \text{time} \); RGV immediately identify the number of CNC that issued the requirements:
1. If no other job instructions are received, RGV waits in place until the next job instruction.
If only 1 CNC issues instruction is received, the dispatch is carried out in accordance with principle 1, then perform Step3;
② If 2 CNC issues instructions are received, the dispatch is carried out in accordance with principle 2, then perform Step3;
③ If 3 CNC issues instructions are received, the dispatch is carried out in accordance with principle 3, then perform Step3;
④ If 4 or 5 CNC issues instructions are received, the dispatch is carried out in accordance with principle 4, then perform Step3;
⑤ If 6, 7 or 8 CNC issues instructions are received, the dispatch is carried out in accordance with principle 5, 6 or 7, then perform Step3;

**Step3:** When \( \text{time} \geq 3600 \times A \) seconds (\( A \) is the maximum time for a continuous job per shift (in hours)), the algorithm stops; otherwise, Step2 is executed.

[Note 1]: The formula for calculating the working time of RGV is \( (\text{time} = \text{time} + t_{\text{da}} + SS_\text{n} + C) \).

[Note 2]: In view of how to distinguish the number of CNCs issued by the RGV, this paper compares the working time of the current RGV by calculating the expected completion time of the CNC (that is, the sum of the start processing time and the length of the operation). If the expected completion time is less than the working time, such CNC is considered to have sent a request and program needs to count the number of these CNC.

### 3.2 Semi-intelligent two-process trouble-free dynamic scheduling model

#### 3.2.1 dual-process arrangement of different CNC

The first and second processes of each material are completed by two different CNC, and this article agreed that the second process must be started only if the first process has been completed. Then this article first needs to install 8 CNC with different tools to process different process.

**Definition 3:** Record \( x \) (non-integer) as the number of CNC producing the first process, \( 8 - x \) (non-integer) is the number of CNC producing the second process.

**Definition 4:** Record \( X \) as the nearest integer from the calculated \( x \), that is \( X = \lceil x + 0.5 \rceil \). Let \( \Delta x = X - x \).

**Balance principle:** In this paper, the smaller \( \Delta x \) of the two schemes is obtained:
① Let \( x_1 \) be the number of CNC in the production of the first process, \( 8 - x_1 \) be that of the second process. If the CNCs in production of the first process are placed in the odd digits, the CNCs in production of the second process are placed in the even digits, then

\[
\frac{x_1}{8-x_1} = \frac{p_1 + SS}{P_2 + SO}
\]  

\( p_1 \) is the time requires CNC to complete the first process. \( p_2 \) is the time requires CNC to complete the second process. \( SS \) is the time takes for RGV to load and unload once for CNC1, 3, 5 or 7. \( SO \) is the time takes for RGV to load and unload once for CNC2, 4, 6 or 8.

② Let \( x_2 \) be the number of CNC in the production of the first process, \( 8 - x_2 \) be that of the second process. If the CNCs in production of the first process are placed in the even digits, the CNCs in production of the second process are placed in the odd digits, then

\[
\frac{x_2}{8-x_2} = \frac{p_1 + SO}{P_2 + SS}
\]

By comparing \( \Delta x_1 = |X_1 - x_1| \) with \( \Delta x_2 = |X_2 - x_2| \), select the smaller one as \( \Delta x \), that is, select the scheme with the smaller difference between \( x \) and \( X \). According to this, set the number of CNC.
producing the first process is $X$, the number of CNC producing the second process is $8 - X$.

3.2.2 CNC arrangement of two-process Jobs

Definition 5: Let $M$ is the number of CNC in the production of the first process, $N$ is the number of CNC production of the second process. If $M < N$, note the CNC in production of the first process as "The first type of rare CNC"; If $M > N$, note the CNC in production of the second process as "The second type of rare CNC". Both "The first type of rare CNC" and "The second type of rare CNC" are called "rare CNC".

Definition 6: Give a graph $Q$ to find its $m$ vertices so that the sum of the distances of these $m$ vertices to the rest of the vertices is the smallest, then these $m$ vertices are called the $m$-Center vertices of the graph.

$m$-Center Principle Let number of rare CNC be $m$, (assumes that rare CNC is placed in odd digits), constructs a graph $G$, the 8 vertices of the graph correspond to 8 CNC, the distance of any two points CNC-$a$, CNC-$b$ is defined as $d_{a,b}$. The $m$ vertexes are taken from vertexes numbered odd so that these $m$ vertexes are the m-center of graph $G$. Such $m$ rare CNCs are placed according to the results from the m-center.

In the case of trouble-free dual-process, the arrangement of rare CNC is given according to the m-center principle.

① If 4 rare CNCs exist, arrange these 4 CNC on the same side;
② If 3 rare CNCs exist, according to the m-Center principle, to get the arrangement of 3 rare CNCs is optimal;
③ If 2 rare CNCs exist, according to the m-Center principle, get these 2 rare CNC placed in the same side of the middle two CNC;
④ If 1 rare CNC exists, this rare CNC is placed within other CNCs according to the m-Center principle.

**Semi-intelligent two-process trouble-free algorithm**

Step1: According to the balance principle, determine the number of rare CNC ($m$) and which side of rare CNC is placed (odd side or even side), and determine which process these $m$ rare CNC processing (the first, second way), and then according to the m-Center principle, determine the order of rare CNC;

Step2: After the intelligent machining system is energized, the initial position of the RGV is in the middle of CNC1 and CNC2, the initial working time of the RGV is $time = 0$; RGV first use "semi-intelligent single-process trouble-free algorithm" to serve the CNC in first process;

Step3: Calculate the current time, if $time \geq 3600*A$, perform Step4; otherwise RGV discriminates the requirements from the CNC:

① If the CNCs in different processes request, use "Semi-intelligent single-process trouble-free algorithm" to serve one of the CNC in the first process; then use "Semi-intelligent single process
trouble-free algorithm" to serve one of the CNC in the second process, then perform Step3;
② If no job instructions are received, or if the CNC that are demanding are all in the same process, perform Step3;
Step4: Output RGV scheduling arrangement, and the algorithm stops.

3.3 Semi-intelligent single process with fault dynamic scheduling model
RGV records the time of each CNC starting to work. When the time reaches the expected completion time for an item, if the CNC of that item does not send any demanding signal, RGV assumes that the CNC has failed, and change its expected job completion time to infinity.

Semi-intelligent single process with fault algorithm:
Step1: After the system is powered on, the initial position of RGV is set within CNC1 and CNC2 with the initial working time defined as time = 0. RGV starts the first round of job scheduling in the order of CNC1→CNC2→CNC3→CNC4→CNC5→CNC6→CNC7→CNC8 and the total number of completed materials is M = 0;
Step2: RGV immediately identify the number of CNC issue the requirements: if time is more than one of completion time, and RGV did not receive demand signal from that CNC, then the completion time of the CNC is rewritten as infinity and then execute Step2. Otherwise, execute Step3;
Step3: RGV immediately identify the number of CNC issue the requirements:
① If none of job instructions are received, RGV waits until the next job instruction;
② If only 1 CNC issues instructions, the dispatch is carried out in accordance with principle 1, then perform the Step2;
③ If 2 CNC issues are issued, the dispatch is carried out in accordance with principle 2, then perform Step2;
④ If 3 CNC issues are issued, the dispatch is carried out in accordance with principle 3, then perform Step2;
⑤ If 4 or 5 CNC issue instructions, the dispatch is carried out in accordance with principle 4, then perform Step2;
⑥ If 6, 7 or 8 CNC issued instructions, the dispatch is carried out in accordance with principle 5, 6 or 7, then perform Step2;
Step4: If time ≥ 3600*A, the current total number of completed items is recorded as M and the algorithm stops. Otherwise, execute Step2.

3.4 Semi-intelligent dual process with fault dynamic scheduling model
RGV records the time of each CNC starting to work, too. RGV always compares the current time with the expected time:
① If the item is in the first process, and the current time is equal to the expected completion time, the total production number $M = M + 1$;
② If the item is in the first process, and the current time is over the expected completion time, it indicates that the CNC of that item failed before, $M = M$;
③ If the item is in the second process, and the current time is equal to the expected completion time, $M = M$;
④ If the item is in the first process, and the current time is over the expected completion time, the total production number $M = M - 1$.

Semi-intelligent two-process fault-free algorithm:
Step1: According to the balance principle, determine the number of rare CNC; According to the m-Central principle, determine their arrangement;
Step2: The system is powered on. RGV is in the initial position and all CNC are in idle state. The initial time time = 0, and the number of completed material M = 0;
Step3: Check whether the working time is over A; if so, the system job stops, the algorithm scheduling ends; if not, the Step4 is executed;
Step4: Check whether CNCs in the first or second channel are sending requirements at the same time; if so, perform Step5; if not, return to Step3

Step5: For CNC in the first process, transform questions into a "semi-intelligent single-process trouble-free algorithm" and schedule according to the principles; Then calculate the current RGV working time;

Step6: Compare the current predicted completion time with the current schedule; if they are equal, then \( M = M + 1 \); Otherwise, M is not changed;

Step7: For CNC in the second process, transform questions into a "semi-intelligent single-process trouble-free algorithm" and schedule according to the principles; Then calculate the current RGV working time;

Step8: Compare the current predicted completion time with the current schedule; if they are equal, then M is not changed; Otherwise, \( M = M - 1 \);

Step9: Serve the selected CNC and calculate the working time, then returns the Step3.

4. Simulation results and validity of the algorithm

Table 1: Validation data sheets for the operation parameters of the intelligent machining system

| System operating parameters                      | Group 1 | Group 2 | Group 3 |
|--------------------------------------------------|---------|---------|---------|
| Time for RGV to move 1 unit                      | 20      | 23      | 18      |
| Time for RGV to move 2 unit                      | 33      | 41      | 32      |
| Time for RGV to move 3 unit                      | 46      | 59      | 46      |
| Time for CNC to complete a single process        | 560     | 580     | 545     |
| Time for CNC to complete the first process       | 400     | 280     | 455     |
| Time for CNC to complete the second process      | 378     | 500     | 182     |
| Time for RGV to load and unload for CNC1, 3, 5 or 7 once | 28      | 30      | 27      |
| Time for RGV to load and unload for CNC2, 4, 6 or 8 once | 31      | 35      | 32      |
| Time for RGV to complete a material cleaning operation | 25      | 30      | 25      |

Note: each shift for 8 hours.

CNC sends out the demanding signal after processing, and there is an idle time between waiting for RGV to come forward and feed. In this paper, the principle of designing the dynamic scheduling algorithm is to minimize the average idle time of each CNC. Based on this idea, this paper designs a method to verify the effectiveness of the model: the ideal machining time of CNC ratio the actual machining time.

Calculate the total number of completed materials, total of three groups of given data list is as follows:

Table 2: Simulation results of four algorithms under three sets of data

|                  | Group 1 | Group 2 | Group 3 |
|------------------|---------|---------|---------|
| Single process   |         |         |         |
| without failure  | 357     | 337     | 367     |
| Double process   |         |         |         |
| without failure  | 210     | 144     | 187     |
| Single process   |         |         |         |
| with failure     | 352     | 336     | 364     |
| Double process   |         |         |         |
| with failure     | 201     | 140     | 184     |

When calculating the effectiveness of the model, the amount of waste caused by the fault is small
and can be neglected compared with the total number of materials, so the efficiency value does not change much. Therefore, this paper only calculates the efficiency value of single process without failure and double process without failure, as shown in list below.

| Table 3: Two fault-free algorithms are efficient in three sets of data |
|---------------------------------------------------------------|
| Group 1 | Group 2 | Group 3 |
| Single process without failure | 96.63% | 95.68% | 96.71% |
| Double process without failure | 79.45% | 73.77% | 78.86% |

This paper can be concluded that: in the "single process trouble-free RGV dynamic scheduling algorithm", "double process trouble-free RGV dynamic scheduling algorithm", "single process of RGV containing fault dynamic scheduling algorithm" and "double process of RGV containing fault dynamic scheduling algorithm", the total efficiency of CNC values to achieve a higher standard, namely every CNC can be fully used, and confirms the practicability and validity of the algorithm.

5. Improved Model -- -- -- -- - Fully Intelligent RGV Dynamic Scheduling Model

In this paper, a kind of fully intelligent RGV dynamic scheduling model is proposed, namely the RGV knows the whole producing process of CNC and advances their real-time situation (RGV can early moves to the position wait for the next work).

As a result, RGV system requires with a register and internal processor, register is used to continually check the start time for each of the process, and use the processor to calculate the completion time of the material. Consider the minimum combination of ‘Time prior material still needs+ Time for RGV to move + Time for working scheduling’ for the first job scheduling.

When time reaches expected completion time for a certain material, CNC of that material did not send a signals, the CNC fails, and will be removed from the job queue (change its expected job completion time into infinity).

Fully intelligent RGV dynamic scheduling algorithm

Take single process trouble-free as an example:

**Step1:** After the system is powered, the initial position of RGV is set within CNC1 and CNC2 with the initial working time defined as . RGV starts the first round of job scheduling in the order of \( \text{CNC1} \rightarrow \text{CNC2} \rightarrow \text{CNC3} \rightarrow \text{CNC4} \rightarrow \text{CNC5} \rightarrow \text{CNC6} \rightarrow \text{CNC7} \rightarrow \text{CNC8} \).

**Step2:** Calculate the current working time of RGV , and RGV immediately judge the number of CNC issuing the demand instructions. If none, turn Step3; otherwise, turn Step4;

**Step3:** Calculate total time of every CNC:

\[
T_i = T_e + T_m
\]

\( T_e \) is the total time; \( T_e \) is the difference between the time and time required by CNC next time; \( T_m \) is the time the RGV moved to the CNC.)

RGV choses CNC that needs totally least waiting time to serve, then turn Step2;

**Step4:** Transform the problem into ‘Semi-intelligent single process without failure’, namely, according to the number of CNC, schedule in accordance with the principles primarily

Computing new ‘time’ currently, that is,

\[
time = time + T_m
\]

And expecte whether there is a new CNC to send the demand signal before ‘time’:

(1) If not, RGV directly to serve primary CNC, then turn Step2;

(2) If yes, renew the CNC machining time \( p'_{\text{new}} \).
\[ p_{\text{new}} = p_{\text{new}} + T_r \]  
(5)

Transform the question into ‘Semi-intelligent single process without failure’. According to the number of CNC, schedule in accordance with the principles, then turn Step2

**Step5:** If \( time \geq 3600*A \), algorithm stops

### 6. Conclusion

The scheduling of each step of RGV starts from the perspective of making full use of CNC processing time, so that each step of operation is the optimal solution meeting the current situation. In order to achieve the scheduling problem of two processes, this paper considered different situations of the combination of RGV and CNC, and determined the producing proportion of CNC in the first and second process as identical as possible in the same time. In terms of the arrangement of rare CNC, this paper aims at radiating as much as possible to non-rare CNC as each rare CNC can, and calculates the sum of the time it takes for rare CNC to traverse non-rare CNC under different arrangements, and takes the arrangement with the minimum time as the actual arrangement.

However, the scheduling of RGV is only related to the CNC that sent the request signal before, and the CNC that sent the request during the operation cannot affect the scheduling decision, which may cause the decision deviation. This paper also puts forward the improved model in chapter 6, the completion time and decision making in advance can be predicted by the way to overcome this kind of problem. With the automation and intelligence of production, the application of RGV dynamic scheduling in production is more and more extensive, and the more efficient and more stable RGV dynamic scheduling model will also be explored constantly.

### Acknowledgements

This paper was financially supported by National College Students Innovation and Entrepreneurship Training Program of China---project number: 201810294084).

### References

[1] Hongqu Zhou, Huizhen Zhao. Hybrid discrete particle swarm optimization algorithm in the application of mixed flow assembly line production scheduling [J]. Journal of chongqing university of science and technology (natural science edition), 2015, (3) : 58-64. The DOI: 10.3969 / j.i SSN. 1674-8425 (z). 2015.03.012.

[2] Kang Jiang, Huaitian Zhuang, Kai Xian. The discrete manufacturing system for production scheduling algorithm research and design? [J]. Journal of combination machine tools and automatic processing technology, 2016, (11) : 154-156160. The DOI: 10.13462 / j.carol carroll nki mmtamt. 2016.11.041.

[3] Lijun Fu, Chong Zhou. Elevator group control system of the optimal scheduling simulation [J]. Computer simulation, 2012, (4) : 263-267. DOI: 10.3969 / j,i SSN. 1006-9348.2012.04.064.

[4] Jiyang Ji, Chang’an Zhu. Production scheduling based on Petri net and simulated annealing algorithm [J]. Journal of jiangsu university of science and technology (natural science edition), 2011, 25 (4) : 346-349. The DOI: 10.3969 / j,i SSN. 1673-4807.2011.04.009.

[5] Xiao Xiongwu, Guo Bingxuan, Li Deren, Et al. Multi-view Stereo Matching Based on self-adaptive Patch and Image Grouping for Multiple Unmanned Aerial Vehicle Imagery[J].Remote Sensing,2016,8(2). DOI:10.3390/rs8020089.

[6] Kang Jiang, Kai Xian, Yu Hao. Engineered. Split the MTO enterprises of production scheduling based on the order and the algorithm research [J]. Manufacturing technology and machine tools, 2018, (3) : 143-148. The DOI: 10.19287 / j.carol carroll nki. 1005-2402.2018.03.029.

[7] Shusen Yang, Shunsheng Guo. Production scheduling based on hybrid genetic algorithm research [J]. Journal of mechanical manufacturing, 2017 zhongguo kuangye daxue (10) : 108-111116. DOI: 10.3969 / j,i SSN. 1000-4998.2017.10.031.