2020 International Conference on Artificial Intelligence and Communication Technology (AICT 2020)

Intelligent RGV Dynamically Scheduling Strategy Based on Computer Games

Xudong Cai

School of Department of Computer Science and Technology, North China Electric Power University, Baoding 071003, China.

a719591339@gmail.com

Keywords: Static evaluation, Computer Games, Dynamic evaluation.

Abstract: The intelligent processing system plays an important role in production, completing repetitive and boring processing tasks, forming a factory assembly line, saving manpower and improving productivity. The design of dynamic scheduling strategy is the key to the "intelligence" of the processing system. A good dynamic scheduling strategy can save more time and complete more tasks under the same machine and time. Accumulation over time can greatly improve productivity.

1. Introduction

In this paper, we analyze the intelligent processing process and the detailed system flow, we design the optimal rail guide vehicle (RGV) operation scheme based on Computer Games and builds a dynamic scheduling model of the intelligent processing system based on static evaluation. We improve the model and algorithm of the material processing operation with only one procedure to the intelligent system of two processing procedures, and then deepened to the system in case of failure. Taking the static evaluation similar to the "scoring" principle, the model in complex cases is also efficient and highly correlated with the following models. The aim is to determine the dynamic scheduling strategy flexibly and accurately, design the algorithm concisely and efficiently, and simulate the results under each set of parameters. After the model is established, the advantages and disadvantages are analyzed, and the actual evaluation model is combined to propose specific directions for model improvement.

2. Problem analysis

2.1 Brief description of the problem

Figure 1 is a schematic diagram of an intelligent processing system. It consists of eight computer numerical control machine tools (CNC), one rail guide vehicle (RGV), one RGV linear track, and one feeding conveyor, one unloading conveyor and other auxiliary equipment. RGV is a self-driving smart car that can run freely on a fixed track. It can automatically control the moving direction and distance according to the instructions, and it has a robot arm, two mechanical grippers and a material cleaning tank, which can complete tasks such as loading and unloading and cleaning materials.

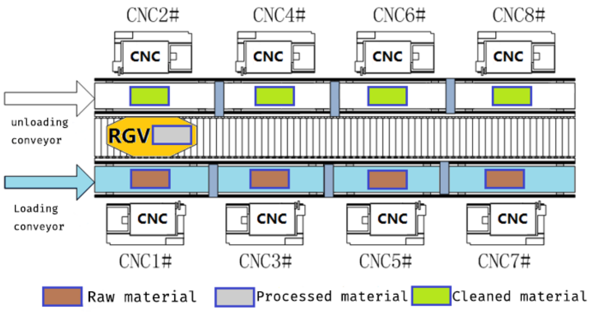

Fig.1 Schematic diagram of intelligent processing system

sThe material processing operation of one process is that each CNC is installed with the same tool, and the material can be processed on any CNC; And the material processing operations of the two procedures is that the first and second procedures of each material are processed by two different CNCs in turn; In the case of CNC, a failure may occur during processing (according to statistics: the probability of failure is about 1%), and the time for each troubleshooting (manual processing, unfinished material scrap) is between 10 and 20 minutes , Join the job sequence immediately after troubleshooting. It is required to consider the material processing operations of one operation and two operations separately.

2.2 Problem analysis

Obviously, one process and no failure are the simplest case, and a model can be established based on this. The rest of the cases can be solved efficiently and accurately by expanding the model.

First, the RGV receives the signal, moves, unloads, cleans, and places the finished product on the unloading conveyor. The key to improving efficiency is to shorten the waiting, moving and loading and unloading time. We use the analysis of the time axis to design the best scheduling scheme under certain circumstances (such as loading and unloading, cleaning) to make RGV work in the highest proportion of time.

3. Establishment of Model

3.1 Model assumptions

To build the model more accurately, we need to establish the following assumptions:

1. When the CNC is empty, it is assumed that the machine claw works in a fixed process and repeats the loading and unloading operations (there is no clinker, but the loading operation is still performed).
2. The analysis time of the demand signal after RGV finishes loading and unloading operations is extremely short and is ignored.
3. The time of failure can be any one second during processing and the probability of failure occurring every second is equal.

3.2 The main principle of the model

Assume that there are only two CNCs. Based on the principle of static evaluation, the effects of different CNCs can be selected for each step of the RGV for comparison based on relevant parameters (that is, the operation takes longer, the longer the time, the lower the efficiency). It can be observed that the dynamic scheduling strategy of RGV is similar to the principle of computer games, and the fitting degree is extremely high.

Nowadays the Mainstream "thinking mode" of mainstream chess and chess computers is to evaluate the situation directly caused by each legal move in the current situation, and then select the move corresponding to the situation with the highest "probability of winning". In other words, "accurately assessing the winning percentage of a given situation" is the core problem of mainstream computer chess players. It is very similar to the intelligent dynamic scheduling of this problem.

As shown in figure 2, the case is two simple CNCs, one RGV, and one processing time axis. White is the time that the machine is paused and waiting for operation. The shorter the time, the higher the efficiency. Blue is the RGV moving time. The shorter the time, the higher the efficiency. Red is the multi-purpose time for even CNC loading and unloading. The above three types of time should be shortened as much as possible, so that the higher the proportion of yellow and gray, the better.

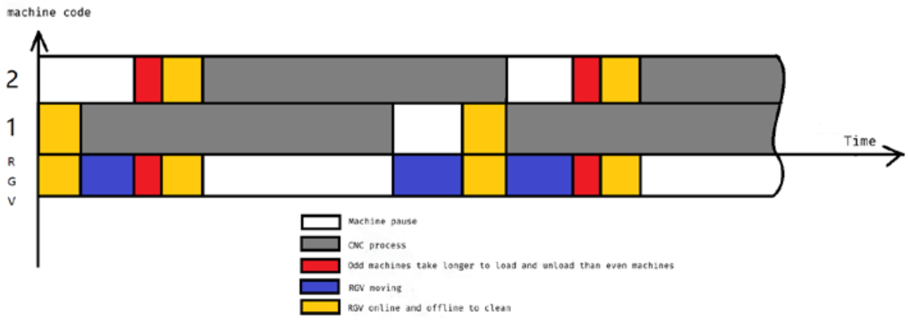

Fig.2 Timeline analysis

3.3 Modeling

In the entire process, the time for loading and unloading is only related to the machine number, and the time for odd-numbered loading and unloading is shorter than the time for even-numbered loading and unloading. The movement time depends on: | RGV position coordinate-target CNC position coordinate |. RGV will wait if all 8 CNCs are processing, or if there is no CNC that can continue processing the semi-clinker on the mechanical claw. After finishing putting the material on the conveyor, the RGV determine the CNC with the shortest T and move it. Judging formula:

$$T = move_t + change_t + wait_t$$

When movet = 0, changet corresponds to the odd machine number CNC, and $wait_t$ = 0, the T value is the smallest. The larger the T value, the longer the CNC waits for the RGV to feed, the lower the efficiency. After the judgment, RGV selects the CNC for loading and unloading and cleaning. After completion, it continues to judge the T of the next 8 CNCs and continues to select.

3.3.1 RGV workflow improvement

If the CNC completion time can be predicted, it can be divided into the following situations:

A. RGV moves to feed the CNC with odd / even numbers.

B. RGV moves, waiting for CNC demand signal, feeding for odd / even CNC.

Specific analysis:

The CNC finish the processing time is $work_t$, and send the CNC $wait_t$ of the demand signal.

When $wait_t <= move_t$:

odd CNC: $T = move_t$.

Even CNC:. $T = move_t + \Delta change$

When $wait_t => move_t$:

odd number CNC: $T = move_t + (work_t - move_t)$.

Even CNC: $T = move_t + (work_t - move_t) + \Delta change$.

The RGV scheduling strategy is optimized to minimize CNC waiting time. In the system process design, the optimal scheme has been reached.

3.3.2 Scheduling strategy of RGV in case one process

Overview on the timeline:

T1 $clean_t$ T2 $clean_t$ T3 $clean_t$ T4 $clean_t$ T5 $clean_t$ T6 $clean_t$ T7 $clean_t$ T8 $clean_t$ T9 $clean_t$ ……Tn $clean_t$

CNC: Idle ... Processing ... Idle ... Processing ... Idle ...

Specific dynamic scheduling strategy:

a. RGV start / completion After placing the finished product on the conveyor belt, judge the T of 8 CNCs and select the CNC with the smallest T value.

b. After time T, RGV will perform cleant for cleaning operation, and CNCt will perform CNCt for processing, so that this CNC will update the data.

$work_t = CNC_t$

Other CNC update data:

$work_t = work_t - T$

c. When the RGV cleant time is over, update all CNC workts and repeat process a. formula:

$work_t = work_t - clean_t$

Note: The minimum value of $work_t$ is 0, if it is negative, it will be automatically updated to 0.

3.3.3 Scheduling strategy of RGV in case two process and solution to CNC process and location allocation

In order to solve the CNC process types and distribution strategies, knife variables are added (1 indicates that the CNC performs the first process, and 2 indicates that the CNC performs the second process) to simulate 256 (28) cases. Obviously, all 8 CNCs are the first. Neither one operation nor the second operation can complete the processing task, so these two cases are excluded. Simulate the remaining 254 cases in turn, compare the efficiency and select the most efficient solution.

Overview on the timeline:

RGV: T_p T_q $clean_t$ T_p T_q $clean_t$ T_p T_q $clean_t$ T_p T_q $clean_t$ ...

CNC: Idle ... Processing ... Idle ... Processing ... Idle ...

Specific dynamic scheduling strategy:

a. RGV start / completion After placing the finished material on the conveyor, determine the T_p of the first process CNC of the processing p station, and select the CNC with the smallest T_p value.

b. After T time, RGV will perform T_q, and the first process CNC will perform CNCp for processing time, so that the CNC will update the data:

$work_t$ = CNCp

Simultaneously update the CNC workt of all the second processes, the formula:

$work_t = work_t - clean_t - T_p$

Simultaneously update the CNC workt of all the first processes, the formula:

$work_t = work_t - T_p$

c. After Tq time, RGV will perform cleant, and the second process CNC will process CNCq, so that the CNC will update the data:

$work_t$ = CNCq

Simultaneously update the CNC workt of all the second processes, the formula:

$work_t = work_t - T_q$

d. At the end of the cleaning operation, RGV updates all formulas of the first process:

$work_t = work_t - T_q - clean_t$

Repeat step a.

Note: The minimum value of workt is 0, if it is negative, it will be automatically updated to 0.

4. Summary

The model Can clearly analyze the factors that affect efficiency and rely on them to improve for different conditions. And the program is highly flexible, it can provide corresponding solutions for different situations. For example, when the processing time is shorter than the loading and unloading time, RGV can still choose the best solution for 8 CNCs. the model has a small amount of program calculation, fast program running, low error rate, RGV makes fast judgments, clear ideas, and is more efficient than actual RGV dispatching according to a certain rule and has better actual resilience. But Every coin has two sides, just like a basic chess computer, each time RGV only considers the optimal solution (CNC) for one step, and after executing one step, it continues to consider the optimal solution (CNC) for the next step. But the efficiency of the two optimal schemes is not necessarily the highest.

References

[1] Chen Hua. Model and Algorithm for Two Rail-guided Vehicles Scheduling Problem Based on a Partitioning Approach [J]. Industrial Engineering and Management,2014,19(06):70-77.

[2] Cha Zhenyuan,Li Jixing,Shen Runtao,Zhang Fenghua,Li Changcheng. Application of Intelligent Pan Rail Guided Vehicle [J]. Robot Technology and Application,2017(05):42-43.

[3] Qiao Fei,Wu Qidi. Real-time scheduling and fault scheduling of RGV in SJ-FMS [J]. Modular Machine Tool & Automatic Manufacturing Technique,1995(03):39-43.