IOP Conference Series: Earth and Environmental Science

PAPER • OPEN ACCESS

Based on Intelligent RGV Dynamic Scheduling Model of Particle Swarm Optimization

To cite this article: Yunlong Jia 2019 *IOP Conf. Ser.: Earth Environ. Sci.* **252** 052135

Based on Intelligent RGV Dynamic Scheduling Model of Particle Swarm Optimization

Yunlong Jia

Shandong Jianzhu University

Abstract. The dynamic scheduling of intelligent RGV is an important factor that affecting the production efficiency of intelligent processing systems, and it plays an important role in manufacturing enterprises to improve production efficiency. This paper analyzes the dynamic scheduling problem of intelligent RGV by establishing a reasonable RGV dynamic scheduling model. First of all, it starts from the case where the machine does not malfunction, and the processing of one process and two processes is expressed by a 0-1 integer plan respectively. Secondly, the shortest time to start processing is the objective function. The single-process RGV static scheduling model and the dual-process RGV static scheduling model based on 0-1 integer plan are established respectively, and the model is solved by particle swarm optimization. On the basis of the RGV static scheduling model, the machine failure condition of the CNC is regarded as the state of continuous operation of the CNC by the case of the machine failure. This paper passes the original model in the case of possible machine failure. The constraint conditions are added, and the single-process RGV dynamic scheduling model and the dual-process RGV dynamic scheduling model are established. Finally, the practicality and effectiveness of the built model and algorithm are verified by numerical experiments, and the simulation experiments of the two models are carried out using eM-Plant software. The models and algorithms established in this paper are effective research and application of dynamic scheduling methods and optimization techniques, which play an important role in manufacturing enterprises to improve production efficiency and reduce costs.

1. Introduction

The use of RGV scheduling in intelligent three-dimensional chemical plants is mainly used in two area: On the one hand, it is the intelligent production that is used in the modern manufacturing industry to match the automated production line. It can realize the rational use of equipment and implement the low-carbon and environmentally-friendly production road through the improvement of production efficiency.On the other hand, in order to meet the development needs of enterprises, and build intelligent warehouses and improve transportation capacity in the era of unprecedented prosperity of electronic commerce, intelligent RGV scheduling is an important factor affecting the production efficiency of intelligent processing systems. The research and application of effective dynamic scheduling methods and optimization techniques play an important role in manufacturing enterprises to improve production efficiency and reduce production costs. The intelligent RGV dynamic scheduling strategy effectively solves the efficiency problem and the efficiency of transportation [1-5]. The particle swarm optimization is an intelligent evolutionary algorithm. Although it has the disadvantages of being limited to the local optimal solution and the slowness of the mature speed, the algorithm has the fast search ability and it is beneficial to the optimal solution under multiple targets. At the same time, this The universality of the algorithm is better, and it is suitable for dealing with the problems and conditions of various situations, and it is easy to integrate the advantages of other algorithms, and it can better solve the problems raised. Therefore, based on the improvement and optimization of the traditional particle swarm optimization, the RGV dynamic optimization scheduling model and algorithm based on particle swarm optimization are established, and the best moving route or processing route of RGV is obtained.

2. Models and Assumptions of System

2.1. Model of System

Figure 1 is a schematic diagram of an intelligent machining system consisting of eight computer numerical control machine tools, including one RGV (rail-type automatic guided vehicle), one RGV linear track, one feeding conveyor and one feeding conveyor and other auxiliary equipment.

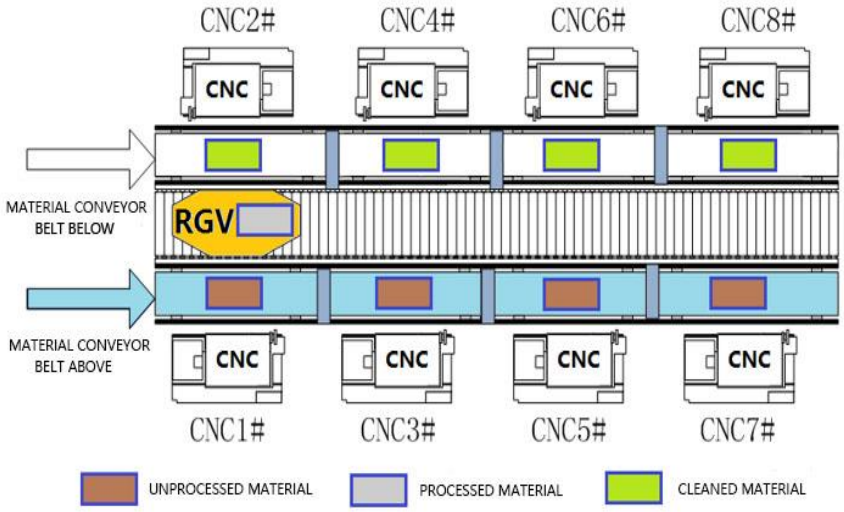

Figure. 1 Schematic diagram of intelligent processing system

The RGV is an unmanned in the picture, 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 comes with a robot arm, and two mechanical grippers and a material cleaning tank, which can complete the tasks of loading and unloading and cleaning materials.

The specific operation process of the intelligent system as follows: (1) the RGV is in the initial position, and all the CNCs are in an idle state after the intelligent processing system is powered on. (2) Under normal working conditions, if a CNC is in an idle state, a loading demand signal is sent to the RGV. Otherwise, the CNC is in a machining operation state, and a demand signal is sent to the RGV immediately after the machining operation is completed. (3) After receiving the demand signal of a certain CNC, RGV will determine the order of loading and unloading of the CNC, and sequentially load and unload the jobs. According to the demand instruction, the RGV runs to a certain CNC where work is required, and the loading conveyor sends the raw material directly to the front of the CNC for RGV loading operation. The time required for the RGV to be an even numbered CNC to load and unload at one time is greater than the time required for the odd-numbered CNC to load and unload at one time. (4) After the RGV completes the loading and unloading operation for a CNC, the robot arm is rotated, and the clinker of one robot is moved to the top of the cleaning tank for cleaning operation (only the processed clinker is cleaned). (5) After completing an operation task, RGV immediately determines the execution of the next operation instruction. At this time, if no other job command is received, the RGV waits in place until the next job command. After a CNC completes the processing task of a material, it immediately sends a demand signal to the RGV. If the RGV fails to arrive at it immediately, so the CNC will wait. (6) The system repeats (3) to (5) repeatedly until the system stops working, and the RGV returns to the initial position.

2.2. Hypothesis of Problem

It is assumed that the CNC installed on both sides of the loading conveyor and the unloading conveyor are equidistantly arranged, and each CNC can only install one tool to process one material at a time. If the material processing process requires two processes, and different CNC installations are required to machine different tools separately, and the tool cannot be changed during the machining process. The first and second processes need to be processed sequentially on different CNCs, and the completion time is different. Each CNC can only complete one of the processes. RGV can move and stop waiting in linear orbit according to the command, and it can continuously move one unit (distance between two adjacent CNCs), and two units (distance between three adjacent CNCs) and three units (four the distance between the adjacent CNCs of the station). RGV can only perform one of moving, stopping waiting, loading and unloading and cleaning at the same time. The loading conveyor consists of n segments with one segment before the odd numbers CNC1#, 3#, and up to 2n-1#. The unloading conveyor consists of free segments with one segment before the even numbers CNC2#, 4#, and up to 2n#. The loading and unloading conveyor belt is controlled by the sensor and can only be driven in the same direction, which can be linked or independently.

3. RGV scheduling model and algorithm for one operation

3.1. RGV dynamic scheduling model in one process

The processing optimization scheduling problem in the RGV intelligent processing system can be described as completing the necessary production tasks within a certain time range and certain constraints, and improving the efficiency and cost as much as possible within the scope of the conditions to attain production efficiency. Therefore, we can also approximate such problems as an optimization process for constraints. According to the RGV moving direction, distance, CNC number and other issues during processing, the RGV machining path optimization mathematical model is established, as shown below:

The objective function of RGV machining path optimization is:

$$MinR = \sum_{k=1}^{k}\left(\sum_{i=1}^{i_k} b_{r_k^{i-1}, r_k^{i}} + b_{r_k^{i_k},0}\right) \times route(i_k) \tag{1}$$

In the above formula, nk indicates whether CNCk# (k=1, 2, 2n) participates in processing materials at this moment. When ik=0, it means that CNC does not participate in processing materials at this moment, and ik=1 participates materialof processing at this moment.

$$route(i_k) = \begin{cases} 1, i_k = 0 \\ 0, i_k = 1 \end{cases} \tag{2}$$

The constraints of the RGV processing path scheme are:

$$\begin{cases} \sum_{i=1}^{\mathrm{i}_k} p_{r_k^i} \le pk; \mathrm{i}_k \ne 0 \\ \sum_{i=1}^{\mathrm{i}_k} b_{r_k^i, r_k^i} + b_{r_k^{\mathrm{i}},0} \le B_k; \mathrm{i}_k \ne 0 \\ R_{k1} \cap R_{k2} = \varphi; k_1 \ne k_2 \\ \bigcup_{k=1}^{K} R_k = \{1, 2, \cdots, M\}; 0 \le \mathrm{i}_k \le M \\ T_{\mathrm{start}\,2} - T_{\mathrm{start1}} \ge \Delta T_{\mathrm{move}} + \Delta T_{\mathrm{c}} + \Delta T_{\mathrm{m}} \\ T_{\mathrm{startf2}} = T_{\mathrm{startl1}} \end{cases} \tag{3}$$

Where B_k represents the maximum value of the RGV processing path, R_k represents the set of RGV going to the CNCk# path scheme, r^k_j represents the order of the RGV going to the CNCk# processing path is j, T_{start2} represents the blanking start time, T_{start1} represents the loading start time, ΔT_{move} represents the RGV moving time , ΔT_{c} represents a loading and unloading time, ΔT_{m} represents the time when a material is processed, T_{startf2} represents the starting time of the blanking of the previous material, T_{startl1} represents the loading start time of the next material.

3.2. Particle swarm optimization

It is easy to know from the form of the objective function (1) that RGV is required to meet the processing in a moving process at a certain time and efficiently. At this time, the RGV moving machining path is required to be the shortest, so it is necessary to find an optimal route that satisfies each constraint at the same time. In order to solve the above-mentioned RGV dynamic scheduling optimization model, based on the improvement and optimization of the traditional particle swarm optimization algorithm, the algorithm for obtaining the best RGV moving route or processing route is designed, including:

Optimization of speed and position formulas:

Suppose that there are N particles forming a community, and one D-dimensional vector is represented by the i-th particle in a D-dimensional target search space:

$$X_i = (x_{i1}, x_{i2}, \cdots, x_{iD}), i = 1, 2 \cdots, N \tag{4}$$

The motion velocity of the. i-th particle is also a D-dimensional vector, which is recorded as

$$V_i = (v_{i1}, v_{i2}, \cdots, v_{iD}), i = 1, 2 \cdots, N \tag{5}$$

The individual extremum is the optimal position that the. i-th particle has searched so far, which is recorded as

$$P_{best} = (p_{i1}, p_{i2}, \cdots, p_{iD}), i = 1, 2 \cdots, N \tag{6}$$

The global extremum is the optimal position that the entire particle swarm has searched so far, which is recorded as

$$g_{best} = (p_{g1}, p_{g2}, \cdots, p_{gD}), i = 1, 2 \cdots, N \tag{7}$$

When you find these two optimal values, the particles change their speed and position according to the following formula.

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \tag{8}$$

$$v_{id}(i+1) = w \times v_{id}(i) + c_1 \times rand() \times (P_{best} - x_{id}(\mathrm{i}) + c_2 \times rand() \times (g_{best} - x_{id}(i)) \tag{9}$$

$$x_{id}(i+1) = x_{id}(i) + v_{id}(i+1) \tag{10}$$

$$w' = w_{\min} + (w_{\max} - w_{\min}) \exp\left(-\frac{t}{T_{\max} - t}\right) \tag{11}$$

The index of the evaluation particle is the fitness value, and the fitness function determines the fitness value. For the RGV moving machining path scheduling optimization problem and the more constraints, the fitness function is expressed as the following form:

$$f(x) = \sum_{k=1}^{K} (\sum_{i=1}^{n_k} b_{r_k^{i-1}, r_k^i} + b_{r_k^{nk}, 0}) \times route(n_k) \tag{12}$$

3.3. Mobile processing flow of RGV

In this paper, the inertia weight w of the traditional algorithm is optimized and improved, and the dynamic inertia weight w nonlinear variation particle swarm optimization algorithm is proposed. The RGV mobile machining process based on the improved particle swarm optimization algorithm is designed:

1: His parameters are initialized and assigned to Tmax, factors c1, c2, wmax, wmin and particle swarm size n. Where wmax and wmin are the minimum and maximum values of w respectively, and Tmax is the maximum number of iterations.

2: The population particles are initialized, and each CNC is numbered and the RAG is initialized and encoded at each CNC point walking path order.

3: Each particle is decoded and the RGV moving processing path is generated after decoding, and the fitness value is solved according to the position of each CNC demand point by using the equation (6) as the fitness function, and multiple conditions satisfying the formula (3) are required, if the result does not satisfy the condition, the tracking is performed again.

4: Compare each particle in the particle swarm with the past optimal value Pbest and the past optimal value of the set population, and use this to determine whether to update the two optimal values.

5: Apply (5) to optimize the w value and find the next moving direction of the particles.

6: Determine whether the preset T_{max} is reached and whether the end condition is met. Enter this position if it is satisfied, otherwise, the process skip to 3.

7: The optimal position of the decoding algorithm is obtained, and the RGV optimal moving path scheme is obtained.

The above mobile processing flow of RGV is shown in Figure 2.

4. Scheduling Model and Algorithm of RGV for Two Processes

4.1. Scheduling Model of RGV for Two Processes

When the material processing process requires two processes, since the first and second processes of each material are sequentially processed by two different CNCs, and in order to give the RGV dynamic scheduling model and the corresponding solving algorithm, so this paper adopts two-layer coding. The genetic particle swarm optimization algorithm is used to establish the optimization model. The RGV scheduling strategy is given.

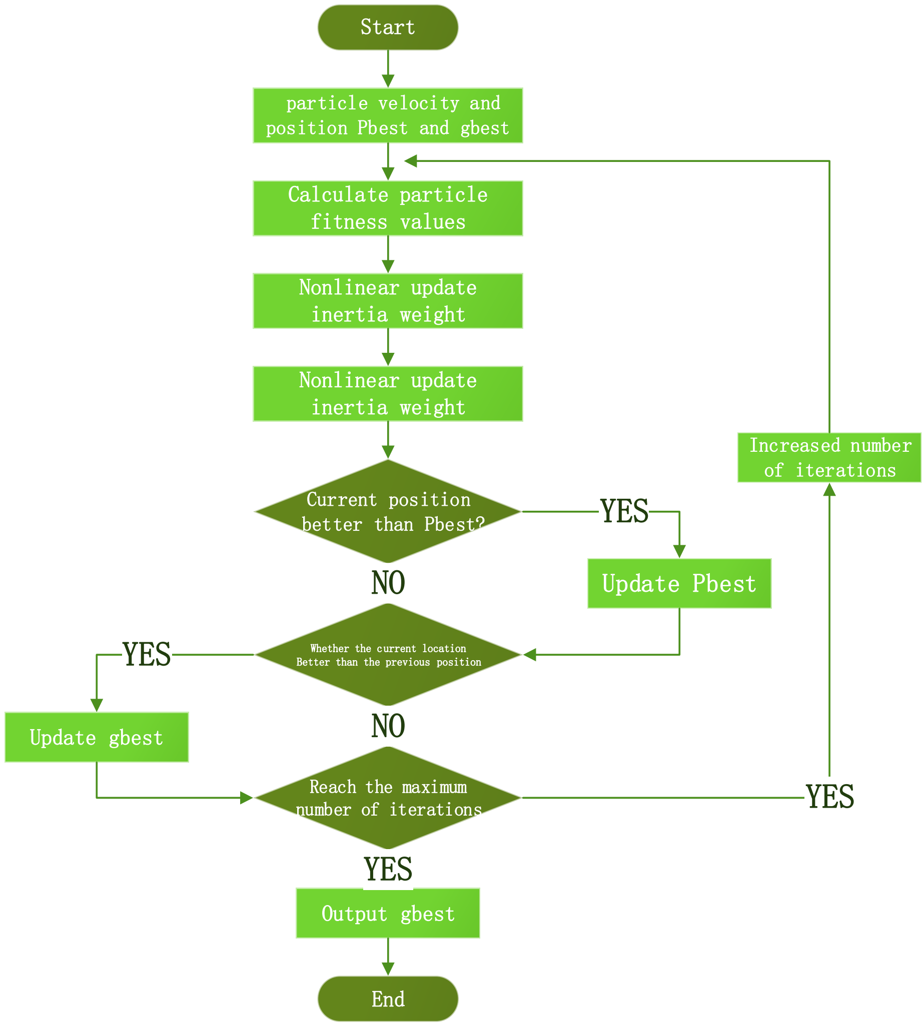

Figure. 2 Flow chart of optimal solution

The first step in optimizing the model is satisfing the following constraints and consider the actual conditions and conditions of the processed material.

$$\sum_{i=1}^{p} y_{si} = 1, s = 1, 2, ..., 2\,\mathrm{n} \tag{13}$$

The second step in optimizing the model is to minimize all process time for processing a given material, so the objective function can be set.

$$F(y)_{\min} = \sum_{i=1}^{p}\sum_{j=1}^{2} D_{ij} \tag{14}$$

The self-updating process in optimization is:

$$Q_i^{k+1} = X\left[f\left[X\left[g\left[X\left[c\left(Q_i^k, ch_i^k \right) \right], gh^k \right] \right] \right] \right] \tag{15}$$

Q_i^{k+1} Represents the position of the. I-th particle in the k-th iteration, and ch_i^k represents the individual extremum of the particle i, and gh^k represents the self-renewal in the population extremum optimization.

The self-renewal process for each part is:

$$M_i^k = X\left[c\left[Q_i^k, ch_i^k \right] \right] = \begin{cases} c\left[Q_i^k, ch_i^k \right] \\ ch_i^k \end{cases} \tag{16}$$

$$S\left[c\left[Q_i^k, ch_i^k \right] \right] > S\left[ch_i^k \right] \tag{17}$$

$$S\left[c\left[Q_i^k, ch_i^k \right] \right] \le S\left[ch_i^k \right] \tag{18}$$

Change the position by the extreme value:

$$\begin{aligned} F_i^k &= X\left[g\left[M_i^k, gh^k \right] \right] \\ &= \begin{cases} g\left[M_i^k, gh^k \right] \; S\left[g\left[M_i^k, gh^k \right] \right] > S\left[gh^k \right] \\ gh^k \;\; S\left[g\left[M_i^k, gh^k \right] \right] \le S\left[gh^k \right] \end{cases} \end{aligned} \tag{19}$$

The position of the particle itself is self-changing in the process of finding optimization:

$$Q_i^{k+1} = X\left[f\left[F_i^k \right] \right] =\le \begin{cases} f\left[F_i^k \right] \; S\left[f\left[F_i^k \right] \right] > S\left[F_i^k \right] \\ F_i^k \;\; S\left[f\left[F_i^k \right] \right] \le S\left[F_i^k \right] \end{cases} \tag{20}$$

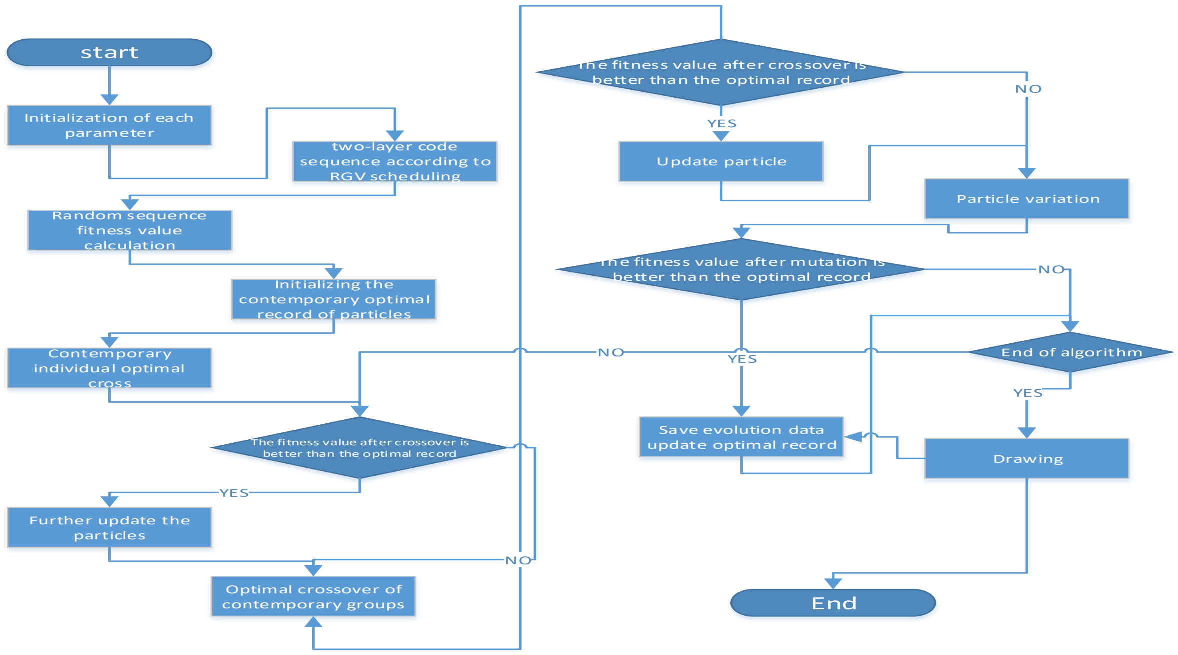

Figure. 3 Flow chart of the optimal crossover algorithm

5. Numerical test and results of simulation

5.1. Numerical experiment of RGV scheduling in one process

In a process, the experiment takes n=8, and there are 8 CNCs in total, and each shift is continuously operated for 8 hours. The MATLAB software can solve the problem by using the algorithm designed in the previous example to obtain the results as shown in Table 1:

Table. 1 Numerical results table in one process

Process material serial number	Process CNC number	Feeding start time	Cutting start time	Process material serial number	Process CNC number	Feeding start time	Cutting start time
1	1	0	562	16	1	1145	1728
2	3	49	641	17	3	1225	1807
3	5	95	719	18	5	1304	1887
4	7	143	798	19	7	1382	1967
5	8	173	-	20	2	1486	2072
6	4	224	978	21	4	1562	2147
7	6	275	1054	22	6	1638	2223
8	2	326	902	23	1	1728	2311
9	1	562	1145	24	3	1807	2391
10	3	641	1225	25	5	1887	2471
11	5	718	1304	26	7	1967	2549
12	7	799	1382	27	2	2071	2652
13	2	901	1487	28	4	2148	2728
14	4	978	1562	29	6	2223	2807
15	6	1055	1638	30	1	2310	2893

From the above results of table, it can be seen that the total number of processed materials is 374. Taking the first 30 sets of data, and it is found that the service efficiency of RAG affects the production efficiency, and the hand of RAG is not enough to serve 8 CNCs. Therefore, we find the shortest Path and optimal service quantity to improve production efficiency in this process. After MATLAB runs the program, we find that the order and time of processing CNC are fixed after the third cycle based on these data and only serve 7 CNCs, which achieves production efficiency overall optimal effect.

5.2. Numerical experiment of RGV scheduling in two processes

The same process is similar, we are taking n=8, and there are 8 CNCs in total, and each shift is continuously operated for 8 hours. The MATLAB software can solve the problem by using the algorithm designed in the previous example to obtain the results as shown in the table:

Table. 2 Numerical results of the two processes

Process material serial	CNC number of	Feeding start time	Cutting start time	CNC number of	Feeding start time	Cutting start time
1	1	0	409	3	53	814
2	5	111	442	8	222	-
3	7	164	576	6	275	-
4	1	409	641	4	333	933
5	5	442	1124	2	386	1097
6	7	576	-	3	814	1371
7	1	641	1239	4	933	1494
8	1	978	1540	2	1097	1661
9	5	1124	1693	3	1371	1945
10	1	1239	1812	4	1494	2067
11	1	1540	2113	2	1661	2236
12	5	1693	2268	3	1945	2521
13	1	1812	2384	4	2067	2644
14	1	2113	2694	2	2236	2815
15	5	2268	2843	3	2521	3095

Through the analysis of the first 15 groups of data, and it can be found that after a certain period of time, RGV reaches the balance of average moving distance and capacity, and achieves the most efficient situation. At this time, only 5 CNCs are in working state, and the remaining 3 CNCs are idle. Because the situation of the two processes is actually a promotion of the situation at the time of a process. When the capacity difference between different processes of a single device is large, it is limited by the arrangement of the devices, there is only one CNC left and right, so RGV will inevitably there is movement, and the RGV moving distance increases. However, at this time the number of RGVs (only one RGV) is limited, and the service capability has not changed. In order to obtain the maximum capacity, more devices will be idle.

5.3. Model verification

The eM-Plant software itself has strong path planning capability, and the simulation results are better, and the fault can be set, which is in line with the simulation requirements of this paper. As shown in Figure 4, a model based on particle swarm optimization and a simulation model based on two-layer genetic algorithm are established. The algorithm of the simulation model solves the multi-process problem with different productivity between different steps through hierarchical operation. The generated Gantt chart can display the processing activities to be performed in chronological order, so as to solve the problem that the path node changes with time. Through the comparison between the results obtained by the model and the simulation results, the results obtained by the two are relatively close.

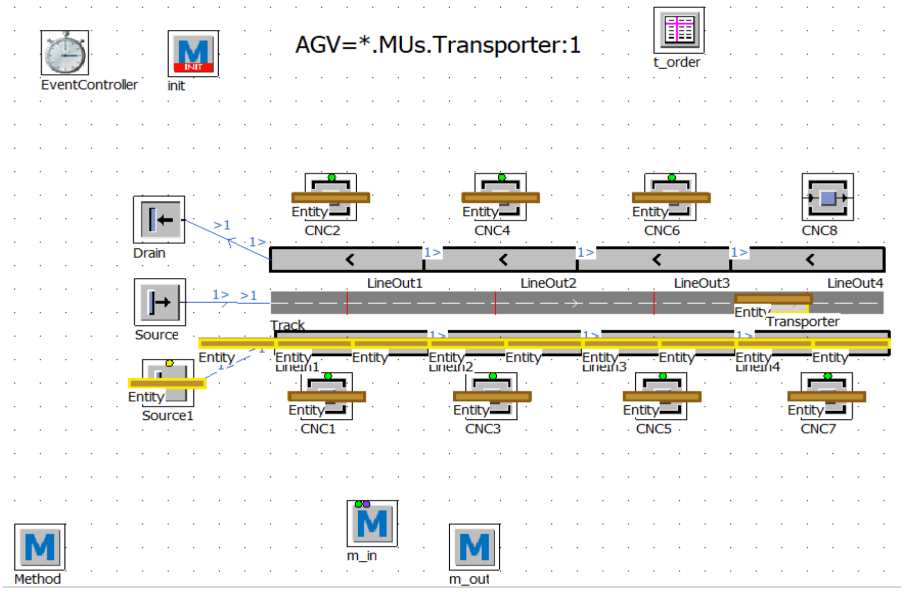

Figure. 4 eM-Plant software simulation diagram

6. Conclusion

In this paper, the dynamic scheduling problem of intelligent RGV is analysed by establishing a reasonable RGV dynamic scheduling model for different work piece processing methods. Firstly, a single-process RGV static scheduling model based on 0-1 integer programming and a dual-process RGV static scheduling model are established for one process and two processes, and the particle swarm algorithm is used, and the model was solved in the case of machine-free operation. Secondly, a single-process RGV dynamic scheduling model and a dual-process RGV dynamic scheduling model are established by adding constraints to the static model in the case that the machine is faulty. Finally, the practicality and effectiveness of the built model and algorithm are verified by numerical experiments, and the simulation experiments of the two models are carried out using eM-Plant software.

References

[1] Hou Yuanbin, Xue Fei, Zheng Maoquan, Fan Rong.Pipeline Optimization Control Based on Multi-layer Coded Genetic-Particle Swarm Optimization Algorithm [J].Journal of Mechanical Engineering,2015,51(09):159-164.
[2] Chen Hua,Sun Qiyuan.Study on the scheduling of linear reciprocating 2-RGV system based on TS algorithm [J].Industrial Engineering and Management,2015,20(05):80-88.
[3] Zhang Guiqin, Zhang Yangsen.Intelligent Scheduling Algorithm for Linear Reciprocating Orbital Automatic Guide Vehicle [J].Computer Engineering,2009,35(15):176-178+181.
[4] WANG Hua-dong,LI Wei.Research on Logistics Distribution Path Optimization Based on Particle Swarm Optimization Algorithm [J].Computer Simulation,2012,29(05):243-246.
[5] Wang Jinxi, Sun Qinghong, Chen Nan, et al. Research and development trend of key technologies of AGVS[J]. Manufacturing Automation, 2002, 24(1): 36-38.