Dynamic Scheduling Method of Multi-objective Job Shop Based on Reinforcement Learning

Zhenwei Zhang[1](✉), Lihong Qiao[2], and Zhicheng Huang[2]

[1] Beijing Institute of Radio Measurement, Beijing 100039, China
rotos@163.com
[2] School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China

Abstract. Aiming at the dynamic scheduling problem in workshop production, we propose a multi-objective scheduling method. By analyzing the actual dynamic scheduling problem, a mathematical model is constructed. Then the dynamic interference factors in the actual production environment are classified, and the interference intensity and its parameters are designed. On this basis, a dynamic scheduling oriented process model is established by using reinforcement learning and scheduling rules, and the design of its state space, state action value table and reward function is introduced. Finally, the model is trained and we analyze the simulation results of different methods. The results show that the dynamic scheduling method based on reinforcement learning has good performance under different periods and disturbance intensity, which shows this method is effective and feasible for dynamic scheduling problem.

Keywords: Reinforcement learning · Dynamic disturbance · Dynamic scheduling · Scheduling rules · Multi-objective

1 Introduction

Due to the complexity of the actual production process, uncertain factors such as equipment downtime, urgent orders, repairing due to quality problems, time adjustment and other factors are difficult to avoid. Job shop scheduling is often shown as complex dynamic scheduling, which needs to adjust the job plan at any time according to the changes of production conditions. For the dynamic scheduling problem, the traditional methods are simplifying the problem, ignoring the disturbance and uncertainty, and transforming the complex problem into a static scheduling problem. These methods need to be redesigned according to the current new state, and models are needed to readjusted according to the changes in the production environment. Since the disturbance of the system state is not considered in traditional methods, the actual production needs cannot be met. In addition, for large-scale production workshops, the order tasks

This work is supported by the National Key Research and Development Program of China under Grant 2018YFB1701800.

L. Zhang et al. (Eds.): CINT 2022, CCIS 1714, pp. 510–524, 2022.
https://doi.org/10.1007/978-981-19-8915-5_44

are complex and a great number of resources are involved, which will geometrically increase the difficulty of scheduling problems. For these large-scale and more complex scheduling problems, traditional methods are not easy to be applied. Therefore, it is of great significance to study the dynamic scheduling problem for actual production.

Scheduling rule is a priority allocation rule, which has the characteristics of low time complexity and high robustness, and is applicable to solve dynamic scheduling problems. In recent years, scheduling rules are often used to solve job shop scheduling problems. Durasevic et al. [1] studied the applicability of different scheduling rules, and found out the scheduling standards suitable for each scheduling rule by testing nine standards and four job types. Kuck et al. [2] proposed an optimization method based on adaptive simulation to select appropriate scheduling rules for production control in the case of equipment failure in complex manufacturing systems. Zhang et al. [3] proposed a semantic-based scheduling rule selection system, which associated scheduling rules with optimization objectives through semantic similarity and semantic expressions, and realized the generation of scheduling rule combinations for a given production target. Rolf et al. [4] presented a method of scheduling rule allocation in solving the hybrid flow shop problem with sequence-related setup times. Lee et al. [5] proposed a sequential search method to set appropriate weight sets for scheduling rules, and used decision trees and hierarchical clustering to improve search efficiency. Braune et al. [6] proposed a tree based scheduling priority rule generation method, which realized the decision-making of job allocation and machine sequencing through single tree and multiple.

Reinforcement learning algorithm is a kind of method that does not rely on samples. Compared with traditional intelligent algorithm, it has higher efficiency and generalization ability. Q-learning (QL) is one of the main methods of reinforcement learning. It is a model-free learning method, which can avoid the huge amount of computation of large-scale scheduling, and is applicable to dynamic scheduling problems. At present, there are more and more researches on reinforcement learning for scheduling problems. Bouazza et al. [7] selected reasonable equipment and process routes for the dynamic scheduling by improving the state-action value table of the reinforcement learning algorithm. Shahrabi et al. [8] used QL algorithm to find the appropriate parameters for problem with equipment failure and dynamic arrival of workpieces. Shiue et al. [9] studied the problem of real-time scheduling (TRS) and proposed a real-time scheduling system using a multiple scheduling rules (MDR) mechanism to ensure that the knowledge base (KB) can respond to changes in the workshop environment in real time. Wang [10] proposed an adaptive scheduling strategy, which avoided the blind search problem of the traditional method through dynamic greedy search, and realized the weighted iteration of Q function by defining state error, which improved the speed and accuracy of the learning algorithm. Qu et al. [11] proposed a multi-agent method for the scheduling of production system covering multiple types of products, equipment and labor, which could adaptively update production plans in real time. Chen et al. [12] proposed a method for flexible job shop scheduling, which took genetic algorithm as the key and intelligently adjusted its parameters based on reinforcement learning. Kardos et al. [13] proposed a new method to select machines according to real-time information, so as to reduce the delay time of the workpiece in production.

The current research on dynamic scheduling problems is mostly based on specific workshop scenarios, and the scheduling scheme is only suitable for special environments, which is not universal. By analyzing the above research, this paper proposes a multi-objective dynamic scheduling method based on QL. Through QL technology, the optimal scheduling strategy under dynamic disturbance is obtained, and the real-time matching between the scheduling strategy and the production environment is realized.

2 Description of Dynamic Scheduling Problem

Job shop dynamic scheduling is a complex optimization problem, which can be described as: n workpieces $Q = \{Q_1, Q_2, ..., Q_n\}$ are processed on m equipment $M = \{M_1, M_2, ..., M_n\}$. Each workpiece Q_i has its corresponding process route and process, and each process corresponds to an optional equipment set. At the same time, it is necessary to consider the disturbance factors in actual production, such as equipment failure, urgent orders, etc. The goal of scheduling is to select the appropriate processing equipment for the workpiece under various constraints and dynamic disturbances, to determine the processing sequence of the workpiece and its working time, and to continuously improve the scheduling index through optimization to meet the expected index requirements.

For the universality of the problem, the dynamic scheduling problem in this paper is based on the several conditions:

- The workpieces arrive dynamically, and the arrival times of the workpieces are random, regardless of the delivery time of the material. The processing time of the operation includes the preparation time.
- Each process of the workpiece corresponds to an optional equipment set, and only one of the equipment can be selected to complete the process.
- The process cannot be stopped halfway after it starts.
- Each equipment can only be used for the processing of one workpiece at the same time, and other workpieces are not allowed to preempt after the processing starts.
- Each workpiece has a definite process route, and the processing is carried out in the order specified in the process route. The next process can only be carried out after its previous process.
- The processing time of an operation has nothing to do with the process route.

To define the dynamic scheduling problem, the definitions of relevant parameters are shown in Table 1:

For the dynamic scheduling problem, the constraints can be described as follows:

$$TQ_{sij} + X_{ijk} \times TQ_{ijk} \leq TQ_{eij} \tag{1}$$

$$TQ_{eij} \leq TQ_{si(j+1)} \tag{2}$$

$$TQ_{eiL_i} \leq C_{\max} \tag{3}$$

Table 1. Parameter definition of dynamic scheduling problem

Symbol	Represention
Q_i	Workpiece i, $i = 1,2,\ldots n$;
Q_{ij}	The jth process of workpiece i, $j = 1,2,\ldots L_i$
M_k	The kth equipment, $k = 1,\ldots,m$
K_{ij}	Optional equipment set of Q_{ij}
H_{ij}	Number of equipment in the optional equipment set of Q_{ij}
TQ_{ijk}	Time of the jth process of P_i on equipment k
TQ_{sij}	The start time of Qij
TQ_{eij}	The end time of Q_{ij}
T_{ei}	Delivery time requirements of workpiece i
C_i	Actual completion time of workpiece i
G_i	Arrival time of workpiece i
C_{max}	Maximum makespan
T_a	Total operation amount of all workpieces
X_{ijk}	1, Q_{ij} processing on equipment k 0, Q_{ij} is not processed on device k
Y_{ijkhr}	1, Q_{ij} is processed before O_{hr} 0, Q_{ij} is not processed before O_{hr}
inf	Positive infinity

$$TQ_{sij} + TQ_{ijk} \leq TQ_{shr} + \inf \cdot (1 - Y_{ijkhr}) \tag{4}$$

$$TQ_{eij} \leq TQ_{si(j+1)} + \inf \cdot (1 - Y_{ikhr(j+1)}) \tag{5}$$

$$\sum_{k=1}^{H_{ij}} X_{ijk} = 1 \tag{6}$$

$$\sum_{i=1}^{n} \sum_{k=1}^{L_i} Y_{ijkhr} = X_{hrk} \tag{7}$$

$$\sum_{h=1}^{n} \sum_{r=1}^{L_h} Y_{ijkhr} = X_{ijk} \tag{8}$$

In the above constraints, Eqs. 1 and 2 represent the production route constraints of the workpiece; Eq. 3 represents the constraint of the completion time of the process; Eqs. 4 and 5 indicate that a piece of equipment can only be used for one process at the same time; Eq. 6 represents the exclusive constraint of a process; Eqs. 7 and 8 represent the usability constraints of the equipment.

The goal of defining dynamic scheduling is to facilitate the evaluation of the effect of scheduling. Common scheduling performance evaluation includes production efficiency, such as maximum makespan; Stability of production process, such as deviation index; Economic indicators, such as production cost, total processing energy consumption and so on. This paper uses the synthesis of multiple indicators as the evaluation indicators.

$$f_1 = \min(\max(C_i)) \tag{9}$$

$$f_2 = \min(\frac{1}{n}\sum_{i=1}^{n}(C_i - G_i)) \tag{10}$$

$$f_3 = \min(\max_{i=1}^{n}(\max(C_i - T_{ei}))) \tag{11}$$

Equation 9 represents the maximum makespan requirement, Eq. 10 represents the index of average flow time, and Eq. 11 represents the index of delayed delivery time. We construct the final scheduling performance index by synthesizing the above indicators, as shown in formula 12, and f is the objective function of comprehensive optimization.

$$f = \min F(f_1, f_2, f_3) \tag{12}$$

3 Dynamic Disturbance and Scheduling Rules

3.1 Analysis of Dynamic Disturbance Factors

The dynamic disturbance factors in actual production can be divided into indirect disturbance and direct disturbance according to their performance characteristics. Indirect disturbances, such as processing time deviation, poor material turnover, and equipment efficiency decline, etc., will affect the execution of scheduling only when these factors accumulate to a certain extent. Direct disturbance will significantly interfere with the scheduling and cause the adjustment of the plan. Direct disturbance can be divided into two types. One is related to resources, such as equipment failure, operation interruption, personnel absence, material shortage or delay, etc. The other is related to the workpiece, such as emergency order insertion, random arrival of workpiece, task cancellation, delivery date adjustment, working hours change, workpiece repair, etc. Common dynamic disturbance factors are shown in Table 2.

This paper mainly studies the direct disturbance, focusing on four typical disturbance factors: the dynamic arrival of the workpiece, urgent order, equipment maintenance and workpiece repair. In the actual production process, the workpiece arrives randomly. In this paper, the arrival time of the workpiece is set so that they are uniformly distributed. For urgent order, it can be set by proportion R_1 and advance its delivery date. The equipment is not available if it is under the maintenance period, and the disturbance can be set by the maintenance time proportion R_2. For the quality problems in actual production, it is achieved by setting a certain number of workpieces with a proportion of F for rework. Considering the difference of disturbance degree in actual production, this paper reflects it through different disturbance intensity.

Table 2. Classification of dynamic disturbance factors

General category	Subclass	Disturbance factor	Impact on production
Direct disturbance	Workpiece related	Random arrival of workpiece	Plan deviation
		Order change	Production adjustment
		Delivery date adjustment	Change the number of batch tasks
		Work hours change	Plan ahead or behind schedule
		Workpiece rework	Increase in production tasks
		Urgent order insertion	Subsequent task rescheduling
		Process change	Workpiece process route change
	Resource related	Equipment failure	Reduction in the number of equipment
		Operating disturbance	Postponement of related tasks
		Personnel absenteeism	Increased manpower load
		Material shortage	Delay of material waiting task
Indirect disturbance	Resource related	Poor material turnover	Continuous accumulation will disrupt the original production progress
		Equipment performance degradation	Deviation accumulation will affect the production schedule
	Time dependent	Processing time deviation	The accumulation of time error will affect the implementation of the plan

3.2 Scheduling Rules

The scheduling rules are to calculate the priority of the workpiece according to the selection of processing time, process quantity, delivery period, etc., and select workpiece to be processed for idle equipment.

For dynamic scheduling problems, the evaluation method based on scheduling rules can be used. This method is relatively easy to implement in the actual production environment. It belongs to an efficient closed-loop control method, so it can be used for real-time job shop scheduling. In the process of scheduling, we need to consider the

selection of equipment and the allocation of jobs. This paper analyzes the scheduling rules for these two types of problems, studies the performance differences of different scheduling rules under dynamic disturbance, and finally selects the rules with excellent performance. Common scheduling rules are shown in Table 3.

Table 3. Typical scheduling rules

Rule	Description
FIFO	Give priority to the workpiece that arrives first
SPT	The workpiece with the Minimum processing time is selected
LPT	The workpiece with the Maximum processing time is selected
EDD	The workpiece with the earliest delivery date is selected
MST	The workpiece with the least delay time is selected
MOR	Give priority to the workpiece with the most remaining operations
LOR	Give priority to the workpiece with the least remaining process
LRM	Give priority to the workpiece with the most remaining processing time
SRM	The workpiece with the least remaining processing time is selected

4 Dynamic Scheduling Problem Solving Based on Reinforcement Learning

4.1 State Space Definition

The job shop scheduling process is transformed through the state space to express the system environment of reinforcement learning. The definition of state needs to reflect the features and process of the scheduling environment, and it needs to be able to express different scenarios. In this paper, according to the state of the equipment and the workpiece. The state space is defined by means of feature vectors. The state space includes 5 features, which are shown as follows.

$$s_{k,1} = n_k / n \tag{13}$$

$$s_{k,2} = T_k^a / T^a \tag{14}$$

$$s_{k,3} = \sum_{j=1}^{L_i} TQ_{ijk} \Big/ \sum\sum TQ_{ij} \tag{15}$$

$$s_{k,4} = \sum_{h=j+1}^{L_i} TQ_{ih} \Big/ \sum_{j=1}^{L_i} TQ_{ij} \tag{16}$$

$$s_{k,5} = n_k^d \Big/ n \tag{17}$$

The above equations describe the current state of the system, where n_k represents the number of workpieces processed by the equipment k at the current moment; T_k^a indicates the number of processing operations of equipment k; n_k^d indicates the number of delayed workpieces processed by equipment k. Equation 13 reflects the distribution of processed workpieces on each equipment; Eq. 14 reflects the distribution of processes on the equipment; Eq. 15 reflects the proportion of processing time of the equipment; Eq. 16 reflects the proportion of remaining time of work in process; Eq. 17 reflects the distribution state of delayed workpieces.

The state values constitute a vector [s11, s12, …, s15, s21, s22,…, s25, s31,…, sm5]. The state vector can be transformed into a state value located in a certain numerical interval (such as [0,100]) by using neural network, and the state value can be used as a criterion to distinguish the state of the scheduling environment.

4.2 *Q*-value Table

The action space of QL can be expressed as the scheduling behavior in the current state. This paper selects seven typical scheduling rules as the action space of QL. According to the aforementioned priority rules, the $Q(s, a)$ table can be established by combining the state values. In this paper, 11 states are used to construct the $Q(s, a)$ table, as shown in Table 4.

Table 4. $Q(s, a)$ table

State	Range	Scheduling rule						
		FIFO	SPT	LPT	EDD	MST	LRM	SRM
0	$T_v = 0$	0	0	0	0	0	0	0
1	$0 \leq T_v \leq 10$	$Q(1,1)$	$Q(1,2)$	$Q(1,3)$	$Q(1,4)$	$Q(1,5)$	$Q(1,6)$	$Q(1,7)$
2	$10 < T_v \leq 20$	$Q(2,1)$	$Q(2,2)$	$Q(2,3)$	$Q(2,4)$	$Q(2,5)$	$Q(2,6)$	$Q(2,7)$
3	$20 < T_v \leq 30$	$Q(3,1)$	$Q(3,2)$	$Q(3,3)$	$Q(3,4)$	$Q(3,5)$	$Q(3,6)$	$Q(3,7)$
4	$30 < T_v \leq 40$	$Q(4,1)$	$Q(4,2)$	$Q(4,3)$	$Q(4,4)$	$Q(4,5)$	$Q(4,6)$	$Q(4,7)$
5	$40 < T_v \leq 50$	$Q(5,1)$	$Q(5,2)$	$Q(5,3)$	$Q(5,4)$	$Q(5,5)$	$Q(5,6)$	$Q(5,7)$
6	$50 < T_v \leq 60$	$Q(6,1)$	$Q(6,2)$	$Q(6,3)$	$Q(6,4)$	$Q(6,5)$	$Q(6,6)$	$Q(6,7)$
7	$60 < T_v \leq 70$	$Q(7,1)$	$Q(7,2)$	$Q(7,3)$	$Q(7,4)$	$Q(7,5)$	$Q(7,6)$	$Q(7,7)$
8	$70 < T_v \leq 80$	$Q(8,1)$	$Q(8,2)$	$Q(8,3)$	$Q(8,4)$	$Q(8,5)$	$Q(8,6)$	$Q(8,7)$
9	$80 < T_v \leq 90$	$Q(9,1)$	$Q(9,2)$	$Q(9,3)$	$Q(9,4)$	$Q(9,5)$	$Q(9,6)$	$Q(9,7)$
10	$90 < T_v \leq 100$	$Q(10,1)$	$Q(10,2)$	$Q(10,3)$	$Q(10,4)$	$Q(10,5)$	$Q(10,6)$	$Q(10,7)$

For state 0, that is, the scheduling action has not yet started. This state is empty and is also the initial state, so its value is 0.

4.3 Design of Reward Function

For the reward function *R*, its construction needs to consider the performance indicators of the scheduling system, and the function needs to reflect the impact of action selection on the scheduling results. Therefore, the design of R needs to reflect not only the immediate reward of the action, but also the cumulative impact on the production cycle, and it should also be suitable for scheduling problems of all sizes. For the production efficiency indicators related to time, because these indicators are related to the utilization of equipment, it can be considered to take the working state of equipment as a reward and punishment function, and the equipment state can be defined as follows.

$$\delta_i(t) = \begin{cases} -1, & \text{Equipment } i \text{ is idle at time } t \\ 0, & \text{Equipment } i \text{ is in working state at time } t \end{cases}$$

The reward and punishment function are as follows.

$$r_k = \frac{1}{m} \sum_{i=1}^{m} \int_{\tau=t_{k-1}}^{t_k} \delta_i(\tau) \tag{18}$$

In Eq. 18, m is the number of devices, and r_k represents the reward when it transitions from s_{k-1} to s_k. The absolute value of r_k is the same as the average time that each device is idle when the two states are transferred. It can be seen that the production cycle is negatively correlated with the cumulative return.

$$\begin{aligned} R_{\mathrm{a}} &= \sum_{k=1}^{K} r_k = \frac{1}{m} \sum_{k=1}^{K} \sum_{i=1}^{m} \int_{\tau=t_{k-1}}^{t_k} \delta_i(\tau) = \frac{1}{m} \sum_{i=1}^{m} \int_{\tau=0}^{C_{\max}} \delta_i(\tau) \\ &= -\frac{1}{m} \sum_{k=1}^{m} (C_{\max} - \sum_{i=1}^{n} \sum_{j=1}^{L_i} TQ_{ijk}) = \frac{1}{m} \sum_{k=1}^{m} \sum_{i=1}^{n} \sum_{j=1}^{L_i} TQ_{ijk} - C_{\max} \end{aligned} \tag{19}$$

In Eq. 19, R_a represents the cumulative return. It can be seen that the smaller the C_{max}, the greater the cumulative return R_a.

5 Case Study

To verify the effectiveness of the method, we carry out simulation analysis through MATLAB. The QL algorithm parameters settings: $\alpha = 0.05$, $\beta = 0.9$, $\varepsilon = 0.15$. The initial state-action reward value is zero. For the experimental data, the number of processes of a single workpiece is between 1 and 4, the total number of equipment is 10, the process processing time is between 10 and 25, and the workpiece cache is 100. The processing time, the time interval of arrival, and the required completion time data all meet the normal distribution. The dynamic disturbance parameter settings are shown in Table 5. The training data of QL is randomly generated by the system, with a total of 100000 pieces of workpiece data (the unit of time-related data is hour). In the test phase, seven sets of workpiece data such as 300, 600, 1000, 1500, 2000, 2500 and 3000 are used

Table 5. Dynamic disturbance parameter setting

Parameter	Setting
Time interval from workpiece to random arrival	Mean 15
Date of delivery	Tight period: working hours of workpiece × 3
	Medium period: work hours of workpiece × 5
	Loose period: working hours of workpiece × 8
Disturbance intensity $S(\%)$	Strength 1: $S(R_1, R_2, F) = [2]$
	Strength 2: $S(R_1, R_2, F) = [4]$
	Strength 3: $S(R_1, R_2, F) = [6]$

Table 6. Performance comparison of different rules and QL

Rule number of workpieces	FIFO	SPT	LPT	EDD	MST	LRM	SRM	QL
300	4681	4827	4560	4644	4263	4580	5120	4525
600	8835	9030	8875	8864	8531	8826	9275	8690
1000	14680	14501	14732	14680	14237	14653	15023	14271
1500	23684	22561	22490	23661	21649	23598	24061	21701
2000	31553	29553	30642	31508	29009	30801	31609	28902
2500	36952	36025	36201	36891	35803	36538	37568	35355
3000	43725	42845	42339	43356	42647	43228	44187	42208

respectively, and a single scheduling rule and the QL lgorithm in this paper are used for scheduling.

Table 6 shows the makespan data obtained by different scheduling rules and the QL algorithm in this paper. It can be seen that the advantage of QL is not obvious when the number of workpieces is small, but with the increase of the number of workpieces, the performance advantage of QL scheduling becomes obvious. Through QL, the frequency of the system selects different rules in different delivery periods is shown in Table 7.

It can be seen that there are differences in the selection frequency of scheduling rules. The rule of MST, EDD and LPT are used more frequently under tight period, and MST, LPT, EDD and SPT are used more frequently under medium period, and MST and SPT are used more frequently under loose period. MST has the highest frequency of use under the three periods. These rules with higher frequency contribute the most to the solution of QL algorithm, while other rules are used less frequently and contribute less to the solution of the system.

Table 7. Selection frequency of dispatching rules

Rule period	FIFO	SPT	LPT	EDD	MST	LRM	SRM
Tight	2.7%	17.7%	20.9%	23.5%	29.5%	2.8%	2.9%
Medium	3.0%	18.2%	21.3%	19.6%	32.3%	3.0%	2.6%
Loose	2.4%	26.4%	13.1%	11.4%	38.1%	4.7%	3.9%

Table 8 analyzes the comparison of the time limit and completion of QL and MST under the three periods. It can be seen that QL has better effect in the actual time limit, the tardiness and the advance.

Table 8. Comparison of time limit and completion under three periods

Workpiece	Period	Method	Planned duration	Actual construction period		Delayed completion		Early completion	
			Mean value	Mean value	Mean square deviation	Mean value	Mean square deviation	Mean value	Mean square deviation
1000	Tight	QL	213	244.5	625.6	832.9	854.7	123.5	54.9
		MST	213	290.3	854.2	791.8	1542.7	100	62.2
	Medium	QL	391	196	431.3	550.8	697.3	206.3	122.1
		MST	391	297.8	906.7	1152.4	1764.6	162.4	104.6
	Loose	QL	568	170	314.4	514.3	502	285.4	189
		MST	568	307.7	910.9	1958.9	1908.2	351.6	136.4
2000	Tight	QL	183	267.5	946.2	855.8	1188.6	106.9	51.9
		MST	183	336.9	1100	711.2	1481.1	93.8	54
	Medium	QL	336	202.8	571.2	656.8	1020.3	180.8	95.3
		MST	336	334	1114.2	908.5	1842.3	138.8	96.2
	Loose	QL	488	174.4	464.9	712.4	865.9	251.2	156.4
		MST	488	319.4	1068.3	1525.1	2280.7	279.5	128.4

Figure 1 shows the tardiness of QL and MST under different workpiece cache, including tight period and medium period. It is obvious that the tardiness of two methods increase with the expansion of the cache capacity, and the tardiness of tight period is worse than that of medium period. In addition, the QL curve rises gently than MST in both periods. In the medium period, the MST curve quickly crosses the QL curve when the workpiece cache capacity reaches 70, and it crosses the QL curve when the capacity is only 30 in the tight period.

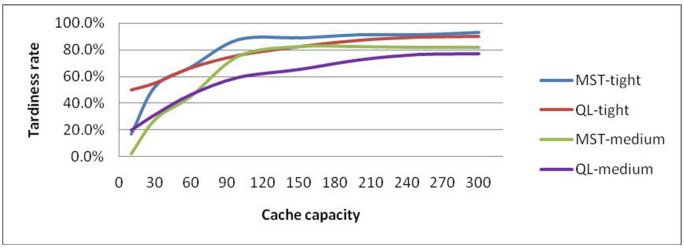

Fig. 1. Comparison of tardiness rate under different cache capacity

Figures 2, 3 and 4 shows the comparison between QL and MST in terms of the change of overdue rate with the number of workpieces under different disturbance intensity and different periods. It can be seen that under the three periods, the overdue rate of both methods increases with the increase of disturbance intensity. When the number of workpieces increases from 200 to 1000, the overdue rate of each period and disturbance intensity increases rapidly. The higher the disturbance intensity, the greater the slope of the curve. After the number of workpieces exceeds 1000, the overdue rate decreases slightly, but remains at a high level. During this period, the overdue rate of QL is generally lower than that of MST. After the number of workpieces exceeds 2000, the overdue rate of QL and MST gradually decreases and stabilizes, and it decreases faster using QL under tight period and high disturbance intensity. By comparison, it is obvious that QL has better effect under tight period and high disturbance intensity.

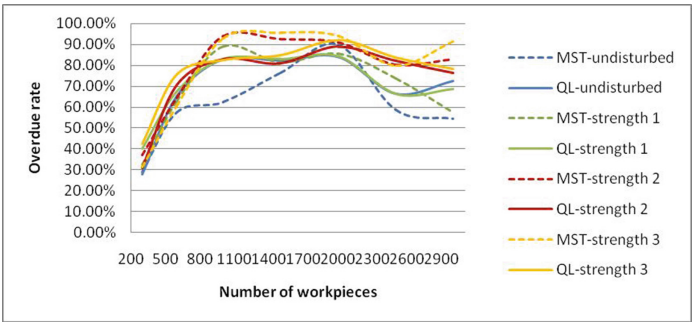

Fig. 2. Comparison of over time of different disturbance intensities under tight period

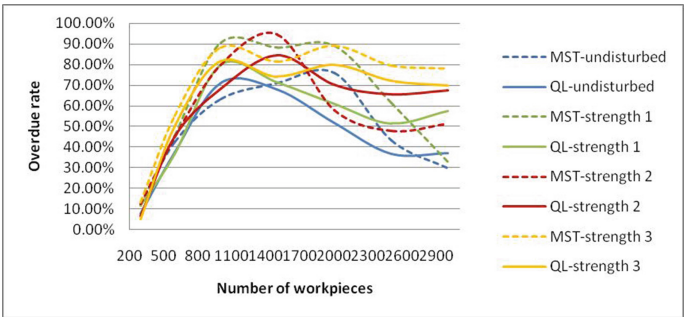

Fig. 3. Comparison of over time conditions of different disturbance intensities under medium period

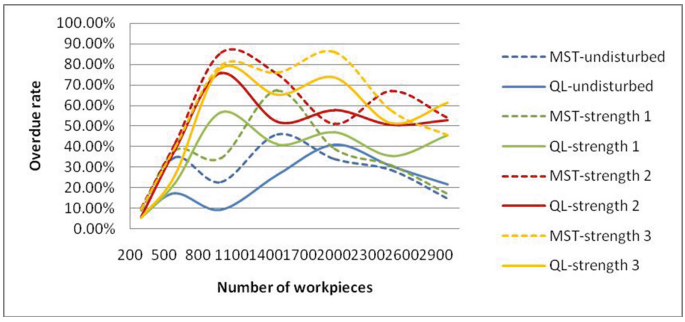

Fig. 4. Comparison of over time conditions of different disturbance intensities under loose period

Figure 5 is a comparative analysis of the maximum makespan of QL, GA and PSO under different workpiece cache capacities in the medium period, in which the number of the workpieces is 1000, the number of genetic algorithm population is 20, and the crossover and mutation parameters are 0.4 and 0.2, the acceleration index is 1.5. The particle size of PSO is 40, the acceleration factor is 2, the inertia weight is 0.5, the maximum particle speed is 0.7, and the number of iterations is 100. From Fig. 5 we can see that when the workpiece is 1000, GA and PSO have advantages over QL in terms of completion time optimization. However, from Fig. 6, the scheduling running time of GA and PSO is much higher than that of QL. When the workpiece cache is low, GA method is about 12 times the running time of QL. When the cache capacity is 80–150, the running time decreases slightly, and when the cache capacity is more than 150, it enters an upward trend. The performance of PSO is worse, and the running time increases sharply after the cache capacity exceeds 40. The running time of QL is always maintained at 2 s, which is only related to the number of workpieces and has nothing to do with the cache. It can be seen that QL is slightly weaker than GA and PSO in terms of completion time, but QL has a better time efficiency advantage in the case of large number of workpiece production and large workpiece cache.

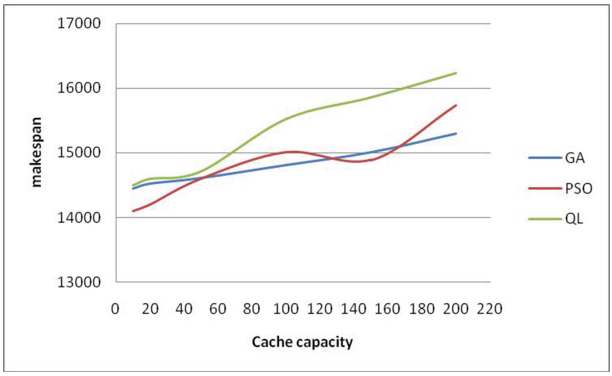

Fig. 5. Comparison of makespan of three algorithms

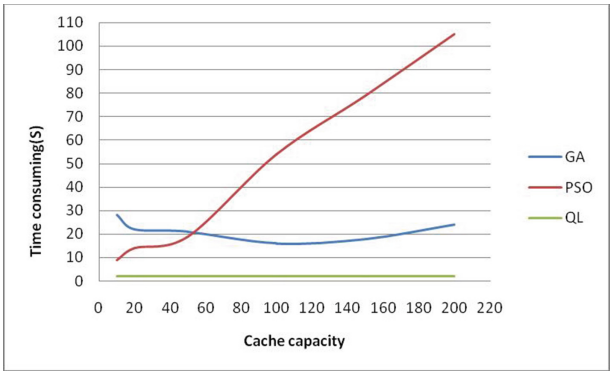

Fig. 6. Comparison of running time of three algorithms

6 Conclusion

This paper studies the dynamic scheduling problem based on reinforcement learning technology and scheduling rules. Typical production disturbances are classified and described through disturbance parameters, the dynamic scheduling problem and its optimization objectives can be consequently represented. On this basis, the state space, Q-value table and reward function of reinforcement learning scheduling are designed. The algorithm is carried out with simulation analysis by using the example data. Through the case study, the method proposed in this paper shows far superior to the traditional intelligent algorithm in time efficiency, and also has a good dynamic scheduling effect. This paper provides a new idea for large-scale job shop dynamic scheduling. For the selection of equipment, this paper adopts the method of man-hour priority, which still lacks the overall analysis of man-hour and load under the complete process route, and further in-depth research can be continued from this direction in the future.

References

1. Đurasevic, M., Jakobovic, D.: A survey of dispatching rules for the dynamic unrelated machines environment. Expert Syst. App. **113**, 555–569 (2018). https://doi.org/10.1016/j.eswa.2018.06.053
2. Kuck M, Broda E, Freitag M, et al. Towards adaptive simulation-based optimization to select individual dispatching rules for production control. In: 2017 Winter Simulation Conference, WSC, pp. 3852–3863. IEEE, Las Vegas (2017)
3. Zhang, H., Roy, U.: A semantics-based dispatching rule selection approach for job shop scheduling. J. Intell. Manuf. **30**, 2759–2779 (2018)
4. Rolf, B., Reggelin, T., Nahhas, A., et al.: Assigning dispatching rules using a genetic algorithm to solve a hybrid flow shop scheduling problem. Procedia Manuf. **42**, 442–449 (2020)
5. Lee, J.H., Kim, Y., Yun, B.K., et al.: A sequential search method of dispatching rules for scheduling of LCD manufacturing systems. IEEE Trans. Semicond. Manuf. **33**(4), 496–503 (2020)
6. Braune, R., Benda, F., Doerner, K.F., et al.: A genetic programming learning approach to generate dispatching rules for flexible shop scheduling problems. Int. J. Prod. Econ. **243**, 108342 (2022)
7. Bouazza, W., Sallez, Y., Beldjilali, B.: A distributed approach solving partially flexible job-shop scheduling problem with a Q-learning effect. IFAC Papersonline **50**(1), 15890–15895 (2017)
8. Shahrabi, J., Adibi, M.A., Mahootchi, M.: A reinforcement learning approach to parameter estimation in dynamic job shop scheduling. Comput. Indus. Eng. **110**(aug), 75–82 (2017)
9. Shiue, Y.R., Lee, K.C., Su, C.T.: Real-time scheduling for a smart factory using a reinforcement learning approach. Comput. Indus. Eng. **125**(Nov), 604–614 (2018)
10. Wang, Y.: Adaptive job shop scheduling strategy based on weighted Q-learning algorithm. J. Intell. Manuf. **31**, 417–432 (2018)
11. Qu, S., Wang, J., Govil, S., et al.: Optimized adaptive scheduling of a manufacturing process system with multi-skill workforce and multiple machine types: an ontology-based, multi-agent reinforcement learning approach. Procedia CIRP **57**, 55–60 (2016)
12. Chen, R., Yang, B., Li, S., et al.: A Self-Learning Genetic Algorithm based on Reinforcement Learning for Flexible Job-shop Scheduling Problem. Comput. Indus. Eng. **149**(1993), 106778 (2020)
13. Kardos, C., Laflamme, C., Gallina, V., et al.: Dynamic scheduling in a job-shop production system with reinforcement learning. Procedia CIRP **97**(1), 104–109 (2021)