A Q-Learning-based Deep Deterministic Policy Gradient Algorithm for the Re-entrant Hybrid Flow Shop Joint Scheduling Problem with Dual-gripper

Jiajia Wang, Hongming Zhou, Jian Guo*, Hongwei Si, Xu Chen, Minghai Zhang, Yaqi Zhang, Guixuan Zhou

Abstract—**This paper addresses the re-entrant hybrid flow shop joint scheduling problem with dual-gripper (RHFSJSPD). In RHFSJSPD, multiple machines and dual-gripper robots are capable of simultaneously handling different workpieces, which gives rise to a complex problem involving job sequencing, task allocation and speed control across multiple workstations. It is characterized by a higher state space dimension and finer action division, thus requiring an efficient integrated scheduling approach. Traditional intelligent algorithms often struggle with the dynamic, combined discrete, and continuous challenges of RHFSJSPD. First, an integrated production and transportation mathematical model aimed at minimizing the maximum completion time and total delay is established to describe RHFSJSPD. Then, a Q-learning-based Deep Deterministic Policy Gradient (DDPG) algorithm(Q-DDPG) is developed to handle RHFSJSPD. The algorithm employs Q-learning for macroscopic job scheduling and DDPG for microscopic dual-gripper control. Greedy strategies are combined with heuristic rules to enhance the search efficiency of Q-learning. The convergence rate of DDPG in the microscopic continuous action space is improved by means of target networks and experience replay. Finally, to evaluate the efficiency of Q-DDPG, two comparative numerical experiments are conducted on a benchmark of 30 instances of RHFSJSPD with different scales. The statistical results reveal that Q-DDPG surpasses the other three Q-learning-based variant algorithms and achieves the highest utilization rates of machines and dual-gripper robots in the actual scheduling process. Furthermore, the integrated scheduling approach leads to an average reduction of 9.41% in makespan and 16.14% in total delay time.**

Index Terms—**re-entrant flow shop, Q-learning algorithm, dual-gripper, multi-agent system.**

Manuscript received October 22, 2024; revised March 28, 2025.

This work was supported by the General Research Project of the Zhejiang Provincial Department of Education (Grant No. Y202249058) and Wenzhou Science and Technology Major Project (Grant No. ZG2023030).

Jiajia Wang is a postgraduate student of the School of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (e-mail: 1569988920@qq.com).

Hongming Zhou is a professor of the School of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (e-mail: zhm69314@163.com).

Jian Guo is an associate professor of College of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (corresponding author, e-mail: 20210701@wzu.edu.cn).

Hongwei Si is a postgraduate student of College of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (e-mail: s1780797785@163.com).

Xu Chen is a Senior Operator of Senken Group Co., Ltd., Wenzhou, Zhejiang, 325029, China (e-mail: chenxu@senken.cn).

Minghai Zhang is a Senior Operator of Senken Group Co., Ltd., Wenzhou, Zhejiang, 325029, China (e-mail: zhangminghai@senken.cn).

Yaqi Zhang is a postgraduate student of College of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (e-mail: 15776958821@163.com).

Guixuan Zhou is a postgraduate student of College of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, Zhejiang, 325035, China (e-mail: 2087153995@163.com).

I. Introduction

In recent years, the flow shop scheduling problem (FSSP) has been fundamental in operations research and industrial engineering, with a primary goal of optimizing job sequencing to enhance efficiency and minimize costs [1]. However, as manufacturing technology advances, the processes and standards in semiconductor fabrication and automotive production are becoming increasingly intricate. Sometimes, workpieces must enter the production system several times for processing or inspection to fulfill requirements [2], giving rise to the Re-entrant Flow Shop Scheduling Problem (RFSSP) [3]. Consequently, there is a growing demand for robust RFSSP solutions that can seamlessly integrate with these advanced manufacturing processes and improve overall operational performance [4]. This is crucial for boosting production rates and shortening delivery times across various industries. The complexity of RFSSP lies in the dual challenges of coordinating job sequencing and re-entrant timing within the inherent variability and unpredictability of contemporary manufacturing. The inherent complexity of RFSSP positions it as a critical area of research, highlighting the need for advanced scheduling algorithms that can effectively manage the intricacies of re-entrant processes.

RFSSP presents a variety of intricate challenges that set it apart from the conventional FSSP. A major challenge is developing effective scheduling strategies to determine the sequence of job processing across machines, especially when jobs may re-enter the same machine multiple times. Such strategies demand a comprehensive understanding of job dependencies and machine availability [5]. Another significant challenge is resource allocation. Due to limited resources, RFSSP requires intelligent scheduling to optimize production capacity utilization. This task entails distributing the workload evenly across machines and ensuring timely resource allocation to jobs to avoid bottlenecks and downtime [6]. Additionally, process control poses research challenges, as its complexity escalates with the system's

reentrant nature. Managing the re-entrant flow of jobs involves optimizing to prevent deadlocks, reduce wait times, and enhance throughput, all while keeping the production system dynamic and responsive [7]. As RFSSP becomes increasingly complex, particularly regarding job re-entry and resource allocation, there is a growing demand for advanced scheduling systems that can effectively manage both production processes and the use of automation resources.

To address these issues, the use of automation tools, such as dual-gripper robots, has become essential in contemporary manufacturing systems. These systems are recognized for their ability to significantly improve production efficiency by providing enhanced operational flexibility [8]. The effective deployment of automated workpiece transportation systems has increased the need for coordinated scheduling methods that optimize both production and transportation resources. These methods not only simplify job sequencing but also strategically manage the use of these resources. They ensure the smooth coordination of processing and transportation activities, which is crucial for maximizing efficiency in production systems [9]-[10]. In re-entrant hybrid flow shops utilizing dual-gripper robots, production sequencing and workpiece transportation are vital elements, and their efficiency directly influences the overall operational efficiency of the workshop. Conventional scheduling approaches usually tackle these two components in stages [11] or separately [12]-[13], neglecting the transportation time of workpieces. This results in a lack of information sharing between the two components, which impedes the achievement of integrated scheduling optimization.

Research on production scheduling problems has been conducted for more than half a century. The emergence and growth of new disciplines and optimization methods in applied mathematics, computer science, operations research, artificial intelligence, and engineering have led to significant advancements in production scheduling research for various manufacturing systems [14]. However, the majority of studies on scheduling issues focus on conventional job shop or flow shop, with current research findings on optimization for RFSP as follows.

Chamnanlor et al. proposed the GACO algorithm to solve the reentrant hybrid flow shop scheduling problem with time window constraints [15]. Johanna and Richard conceptualized a dual-resource-constrained re-entrant flexible flow shop problem, drawing inspiration from the screenprinting sector. They employed a constraint programming model to effectively assess the proposed heuristic approaches and also introduced a hybrid genetic algorithm to expedite the search for better solutions [16]. Jun D et al. developed an enhanced hybrid Salp Swarm algorithm (SSA) and NSGA-III algorithm aimed at addressing the triple objective reentrant shop scheduling problem [17]. Mousavi et al. addressed the re-entrant hybrid flow shop scheduling problem, which incorporates setup times and position-dependent learning effects, by proposing a bi-objective optimization algorithm based on genetic algorithms [18]. Topaloglu et al. developed a modified shifting bottleneck heuristic (MSBH) for the reentrant job shop scheduling problem (RJSSP) with makespan minimization objective [19]. Rong H et al. employed a differential evolutionary algorithm to solve the RFSP problem, aiming to minimizing total drag time as the objective to solve the RFSP problem [20]. Furthermore, numerous researchers have investigated the integrated scheduling problem of RFSP. For instance, Eghonghon et al. conducted research on sequence and time dependent maintenance Scheduling, based on the application of two-stage re-entrant flow shop in industry. They formulated a mixed-integer programming model to more accurately describe the problem and applied heuristic methods for the integrated scheduling of both operational and maintenance tasks [21]. Ying et al. introduced an iterative Pareto greedy algorithm to solve the RFSP problem, targeting minimization of completion time and total delay [22].

The studies mentioned primarily focus on using single-gripper systems for job transportation, but in practice, companies are likely to adopt dual-gripper systems to transport jobs. Hurink and Knust analyzed the complexity of a specific flow shop scheduling problem involving a single robot and unlimited buffer space [23]. Carlier et al. solved a robotic flow shop scheduling problem with various part types and a single robot by developing a branch-and-bound algorithm alongside a genetic algorithm [24]. Paul et al. designed an effective heuristic algorithm to address a hoist scheduling problem dealing with time windows, aiming to minimize maximize the completion time [25]. Che et al. solved a robust optimization problem related to a cyclic hoist scheduling problem, accounting for variations in hoist transport time, and established a mixed-integer programming model to optimize cycle time and robustness, ultimately identifying the Pareto optimal solution [26]. Drobouchevitch et al. considered the scheduling problem of cyclic production in dual-gripper robot cell without buffers, focusing on identical parts, and provided a framework for modeling this type of production system [27]. Sriskandarajah and Shetty investigated the complexity and solvability of the part ordering problem in a dual-machine dual-gripper robot cell, identifying all potentially optimal robot motion sequences [8]. Geismar et al. analyzed a dual-gripper robot cell that processes identical parts with parallel machines at each processing stage, deriving a lower bound on the throughput and an optimal solution under specific practical conditions [28]. Gultekin et al. considered cyclic scheduling for multiple parts on a robot cell, proposing both a mathematical planning model and a metaheuristic model [29]. Dawande et al. addressed the issue of spaced robot cells with dual-gripper robots handling identical parts, ensuring that completed parts exit the machine within a designated time frame, and achieved an asymptotically optimal solution in polynomial time for two extreme scenarios: the no-wait cell and the free-picking cell [30].

Previous studies have demonstrated that dual-gripper systems can lead to greater productivity than single-gripper systems. Moreover, integrated scheduling methodologies provide solutions that are more proximate to the optimal ones when compared with stage-wise or factor-specific scheduling methodology. Therefore, this research aims to address the re-entrant hybrid flow shop scheduling problem with dual-gripper systems through an integrated scheduling method that encompasses both production and transportation aspects. The complexity of re-entrant flow shops is heightened by the addition of dual-gripper robots, which

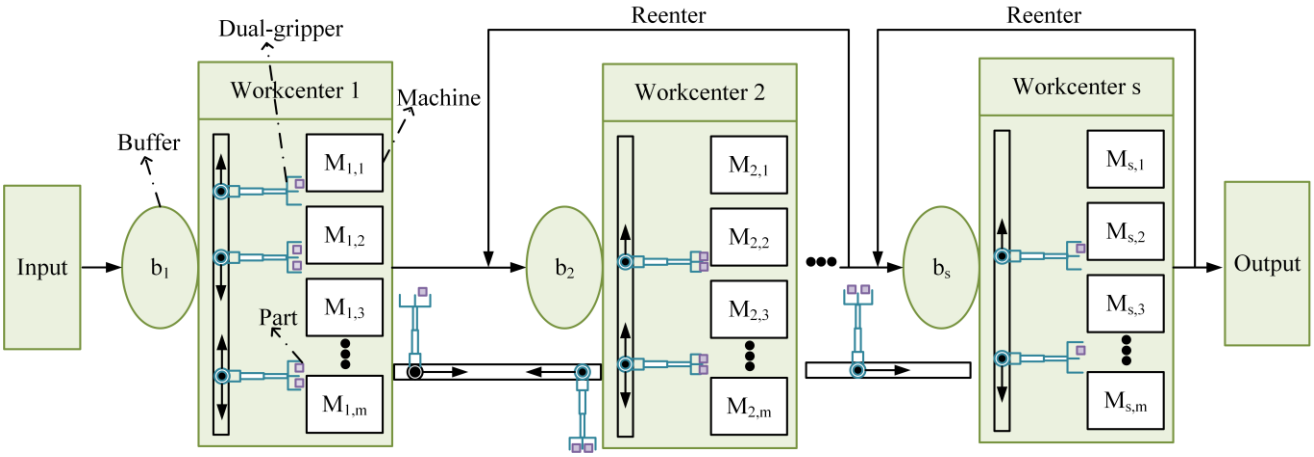


Fig. 1. The illustration of the re-entrant hybrid flow shop with dual-gripper

introduce further dynamic constraints, further complicating the problem. To tackle this more complex integrated problem, an integrated scheduling strategy based on an improved Q-Learning algorithm is proposed. This strategy effectively navigates the intricate interactions between job processing and dual-gripper system scheduling, optimizing the job processing workflow, maximizing resource use, and enhancing overall production efficiency.

II. Description and Definition of the Re-entrant Hybrid Flow Shop Joint Scheduling Problem with Dual-gripper

The joint scheduling problem discussed in this research is derived from studies on scheduling in re-entrant hybrid flow shops, taking into account the transportation resource constraints represented by dual-gripper robots. It examines how the routing and movement features of these dual-gripper robots, which handle workpiece transfers, affect the production process.

The dynamic joint scheduling can be decomposed into three distinct static scheduling components:

(1) Sorting multiple pending jobs for processing.

(2) Selecting processing machines within a specific workstation for each job.

(3) Determining suitable dual-gripper robots for job transfers.

(4) Planning the moving paths of the dual-gripper robots.

The re-entrant hybrid flow shop joint scheduling problem with dual-gripper (RHFSJSPD) is described as follows: In a production line, there are S work centers, each corresponding to a process. Each process is equipped with m_s (m_s>0) unrelated parallel machines for processing, and the processing time of any job in any work center is fixed. The processing sequence among jobs is not fixed. Although there is no absolute restriction on the job sequence, the process-flow sequence of each job along the process path is subject to specific constraints. Some special workpieces may need to re-enter a certain process for repeated processing due to their specific processing requirements. Additionally, if problems are detected in the quality inspection stage, the workpieces will also be sent back to a previous process for re-processing. This re-entry phenomenon indicates that a job may undergo multiple repeated processing iterations within a single process. Different jobs are processed in different sequences at different levels. In each workstation and between different workstations, there could exist more than one dual-gripper robot D_i responsible for workpiece transfer. When the workpiece j_i requires transfer, the central control system designates the dual-gripper robot D_i that has either both grippers idle or one gripper idle to execute the transfer task. Subsequently, D_i conducts the workpiece transfer in accordance with the task priority. As illustrated in Fig. 1, the workpieces are introduced via the input terminal and may re-enter a prior process for subsequent processing upon completion of a certain operation. Each process is equipped with a buffer. In the event that a workpiece enters a process while the equipment is still occupied, it is placed into the buffer and waits for its turn to be processed. The dual-gripper robot is responsible for transferring workpieces between different workstations and the machines at each workstation.

Based on the above analysis of workshop layout and operation mechanism, this paper proposes the following hypotheses for RHFSJSPD:

(1) At any given time, each job can be processed on only one machine, and each machine can process only one job at a time to prevent resource contention.

(2) During the production process, it is assumed that all workpieces, processing machines, and dual-gripper robots are available, and the buffer capacity between adjacent workstations is considered to be unlimited.

(3) The production process for all workpieces to manufacture a specific product is fixed. However, quality inspection steps exist between certain processes. Defective workpieces will return to designated processes for reprocessing.

(4) The dual-gripper robot can carry a maximum of two workpieces at the same time, with each gripper capable of holding only one workpiece. The loading and unloading durations of the dual-gripper robots, as well as the reaction and processing times of all machines and robots, are not taken into account.

(5) The start-up and shut-down times of the machines have no impact on the overall production scheduling, and are regarded as negligible factors in the scheduling model. That is, once the machine is ready for processing, it can immediately start working, and when the processing is completed, it has no subsequent impact on the production process in terms of time.

(6) The transfer paths between workstations for the dual-gripper robots are pre-determined and fixed. There are

no unexpected obstacles or changes in the transfer routes during the transfer process, ensuring the smooth transfer of workpieces between different workstations and machines.

Based on the above problem description and assumptions, the symbols and variables are defined in Table I.

TABLE I
SYMBOLS AND MEANINGS

Symbol	Meaning
N	Total number of jobs
S	Number of sequences, not including inputs and outputs
MN	Total number of machines
M_s	Number of equipment in the s th process
P_s	The time the job is processed in process s
$C_{j,s,t}$	End time of job j in process s in level t
$F_{j,s,t}$	Entry time of job j in process s in level t
C_{max}	Time for all jobs to enter the output
$L_{s,m,s',m'}$	The time required to move the job from equipment m of process s to equipment m' of process s'
φ	Grippe and release time of job by robot grippers
θ	Robot gripper reset time
$X_{j,s,m,t}$	1 Table equation at level t where job j is processed on equipment m in process s, otherwise 0
$E_{j,t}$	1 means that the job does not meet the machining requirements after the t th level of machining, otherwise, it is 0
$R_{s,j}$	1 means that the j th gripper of the robot gripper holds the job at process s, otherwise it is 0
BM	A large enough positive number

III. Mathematical Model

In the joint scheduling problem research, the objective is to sort jobs according to their processing sequence and select appropriate processing machines and dual-gripper robots. This approach aims to reduce bottlenecks resulting from issues like the congestion of re-entering workpieces and delays in transfers, ultimately minimizing both the completion time C_{max} and the total delay time TD.

The objective function represented by Equation (1) aims to minimize the weighted sum of the total delay time and the maximum makespan, where μ denotes the weight coefficient of the completion time, and d_j represents the delay time of job j.

$$F = \mu \times max(C_{\max}) + (1-\mu) \times \sum_{j=1}^{N} \max(0, C_{j,S+1,T} - d_j) \tag{1}$$

Process priority constraints:

In practical production scenarios, the processing of job is subject to the sequential constraints as represented in Equation (2). At the same level, a job may proceed to the subsequent process only upon the completion of the preceding process.

$$F_{j,s,t} + BM(X_{j,s\_1,m',t} - X_{j,s,m,t}) \geq C_{j,s-1,t} + L_{s-1,m',s,m} + 2(\varphi + \theta) \tag{2}$$

To prevent the waste of production resources, the processing time of job is constrained as Equation (3). The job is processed at most once in one process at the same level.

$$\sum_{s=1}^{S} \sum_{m=1}^{M_s} P_s * X_{j,s,m,t} \leq C_{j,S,t} \tag{3}$$

To ensure the integrity of workpiece processing, it is essential for each workpiece to go through all processes corresponding to the specified levels, as shown in Equation (4).

$$\sum_{s=1}^{S} X_{j,s,m,1} = S \tag{4}$$

Quality inspection and other reasons may result in a situation where a job is not required to undergo processing at a certain level and needs to be transferred to the subsequent process. Equation (5) is employed to determine the turning conditions in different situations, and Equation (6) is used to determine the process to which the job is transferred.

$$F_{j,s,t+1} - BM\left(1 - E_{j,t}\right) > 0 \tag{5}$$

$$F_{j,s,t} + BM\left[2 - \left(X_{j,s,m',t} + X_{j,S,m,t-1}\right)\right] > C_{j,s,t-1} + L_{S,m,s,m'} + 2(\varphi + \theta) \tag{6}$$

The re-entrant caused by problems in the quality inspection is assigned a different priority compared to the initially scheduled processing operations, and Equation (7) ensures that the jobs in the higher-level operations of the same process are preferentially processed.

$$C_{j,s,t-1} - BM(X_{j',s,t-1} - X_{j',s,t}) \leq C_{j',s,t} \tag{7}$$

Job to dual-gripper assignment constraints:

In the process of transferring workpieces, Equation (8) is required to limit the number of workpieces grasped by the dual-gripper at any time to be no more than 2, so as to ensure the normal operation of the dual-gripper. Furthermore, to ensure the efficiency of output, Equation (9) stipulates that the number of jobs grasped by the dual-grippers at the output terminal must consistently be greater than 0.

$$\sum_{s=1}^{S} R_{s,j} \leq 2S \tag{8}$$

$$\sum_{j=1}^{2} R_{S,j} > 0 \tag{9}$$

To prevent ambiguity and conflict in the job handling process, Equation (10) specifies that each job can only be transported by one dual-gripper robot at any given time. This regulation guarantees that there is a designated entity responsible for the handling of each job.

$$\sum_{d=1}^{D} x_{i,d,t} = 1, \forall i \in \{1, 2, \ldots J\} \tag{10}$$

Job to processing machine assignment constraints:

To prevent conflicts in machine resource allocation, Equation (11) restricts that a workpiece can only be processed by one machine at the same time. Equation (12) specifies that a machine can process at most one workpiece concurrently.

$$\sum_{m=1}^{M_s} X_{j,s,m,t} = 1 \tag{11}$$

$$\sum_{j=1}^{J} X_{j,s,m,t} = 1 \tag{12}$$

Dual-gripper dynamics constraints:

In the process of conveying workpieces with a dual-gripper robot, the kinetic energy is affected by the movement speed and load. To ensure the safe operation of the dual-gripper robot and avoid damage to components, such as the motor, which could arise from excessive kinetic energy, it is necessary to limit its kinetic energy. Equation (13) demonstrates that the kinetic energy of the dual-gripper robot

is constrained to remain below a specified maximum threshold in order to prevent potential damage to the motors. Where G_d is the gravity of the dual-gripper robot itself, G_j is the gravity of the job, v is the movement speed of the dual-gripper robot, and the speed is obtained by arbitrarily taking a linear value within the interval. Additionally, E_{max} refers to the maximum allowable kinetic energy.

$$\frac{(G_d + \sum G_j \times x_{i,d,t}) \times v^2}{2} \le E_{\max} \tag{13}$$

IV. Algorithm Design

Prioritizing different types of operations is essential in addressing the re-entrant hybrid flow shop scheduling problem with dual-gripper. This initial phase should be followed by the determination of an optimal processing sequence and the selection of appropriate machines for each task. The decision-making for production plans also necessitates an integration with the scheduling decisions of the dual-gripper robot to ascertain the optimal transportation route and speed control strategy. The dynamic nature of this scheduling system, influenced by stochastic elements such as quality inspection rework and dual-gripper conflicts, complicates the identification of effective scheduling rules that can adapt to the fluctuating conditions within the workshop. Therefore, this study proposes an improved Q-learning algorithm combined with Deep Deterministic Policy Gradients (DDPG) to determine the most appropriate scheduling rules, thereby optimizing scheduling decisions within the workshop.

In the improved Q-learning algorithm, a weighted approach is employed to mitigate the complexity of the state space, which includes multiple agents and stochastic elements, thereby streamlining the action space. Gradient-based optimization algorithms are applied to continuously update the weights, thereby iteratively refining the Q-function. To enhance the convergence speed of the algorithm, heuristic rules and greedy strategies are combined to dynamically search for scheduling policies. To effectively address the continuous action space associated with dual-gripper control, which includes both path and speed, the DDPG algorithm is integrated with the improved Q-learning algorithm. The DDPG enhances learning effectiveness in the micro-level control tasks of this scheduling problem through target networks and experience replay mechanisms. In the scheduling system established in this study, jobs and dual-grippers, as two types of agents, interact with the dynamically changing workshop environment to obtain the optimal state-action pairs, thereby selecting the most suitable scheduling scheme.

A. Detailed description of Q-DDPG

The proposed integration of an improved Q-learning algorithm with the Deep Deterministic Policy Gradient (DDPG) method, referred to as Q-DDPG, encompasses four steps: initialization, global search, local search, and interactive update. The framework of the algorithm is illustrated in Fig. 2 In the first step, Q-learning and DDPG jointly perform the overall configuration, parameter initialization, and Q-table initialization to obtain the initial solution, including job sequence, machine assignment, dual-gripper robot assignment, and transport speed. In the subsequent iterations, the algorithm executes the scheduling strategy based on the hybrid Q-learning and DDPG approach, with global search driven by Q-learning and local search performed by DDPG. During the initialization process, not only are the basic parameters and tables initialized, but also a comprehensive analysis of historical production data is conducted. This analysis of historical data is helpful in setting more reasonable initial values for parameters such as the job processing times, machine efficiency, and average speed of the dual-gripper robots. When the historical data indicates that a specific type of job typically takes longer to process on certain machines, this information can be utilized to adjust the initial job-machine assignment probabilities in the Q-table.

In the second step, a global search is conducted by observing the shop floor state, including job progress, machine availability, and dual-gripper robot status. Based on a combination of ε-greedy strategy and heuristic rules, actions are selected to determine job sequencing, machine assignment, and task allocation for dual-gripper robots. During this phase, Q-learning performs the macroscopic scheduling by reducing the state space complexity through weighted feature representation and applying heuristic rules to conduct an efficient global search. The global search first divides the job agents into two subgroups based on task types: normal process jobs and urgent rework jobs, then Q-learning is applied to search for the optimal scheduling solution for each subgroup. The ε-greedy strategy is further refined during the global search process. Instead of having a fixed ε value throughout the entire search process, it is dynamically adjusted according to the number of iterations and the current performance of the scheduling scheme. During the early stages of the search, a relatively large ε value is set to encourage exploration. This implies a higher probability of randomly selecting actions rather than always choosing the known optimal action based on the Q-table. As the number of iterations increases and the solution starts to converge, ε gradually decreases, with more focus placed on exploiting the learned optimal actions. Furthermore, the heuristic rules utilized in the global search are tailored to be specific to both jobs and machines. Jobs with stringent delivery time requirements are prioritized during the scheduling process. Additionally, machines that demonstrate greater efficiency in processing particular types of tasks are more likely to be assigned jobs corresponding to those task types.

The third step involves a local search performed by DDPG after global action selection. The dual-gripper robot agents are classified into three subgroups according to load conditions: idle, single-gripper full load, and dual-gripper full load, to avoid inefficiency and path conflicts in the transfer process. Then, deep policy gradient methods are applied to adjust the paths and transport speeds of the dual-grippers selected by Q-learning, generating continuous control actions at the microscopic level. The Critic network is used to evaluate the effectiveness of the control actions, while experience replay and target networks are employed to stabilize the training process. The Critic network evaluates the effectiveness of control actions not only based on the task completion time but also takes into account the energy

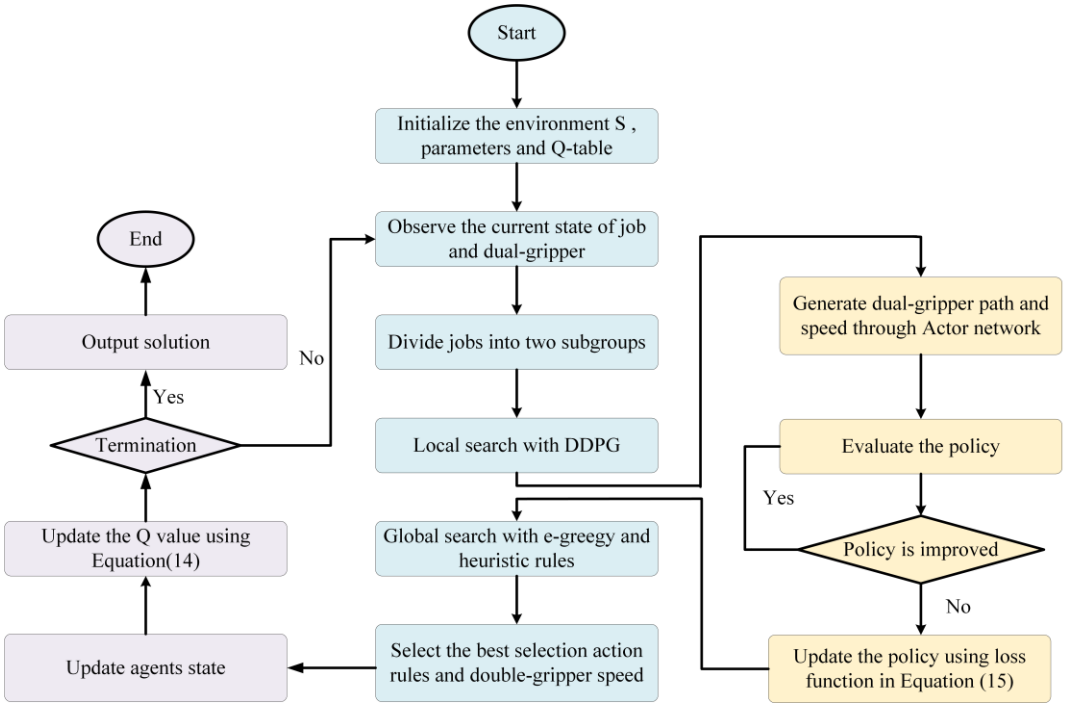


Fig. 2. The flowchart of Q-DDPG for RHFSJSP

consumption and wear of the dual-gripper robots. Such multi-criteria evaluation helps to generate more sustainable and cost-effective control actions.

In the final step, an interactive update occurs, where the macroscopic and microscopic subgroups are recombined to form a new population for the next iteration. The process is repeated until the termination condition is satisfied, and the optimal job sequencing and dual-gripper control strategy are outputted. During the interactive update process, a fitness function is also introduced to measure the quality of the new population formed after the recombination of macroscopic and microscopic subgroups. The fitness function takes into account factors such as the overall makespan of the production process, the utilization rate of machines, and the idle time of dual-gripper robots. Based on the fitness values, a tournament selection method is employed to select the next generation population. This ensures that the most promising solutions have a higher chance of being passed on to the next iteration, accelerating the convergence of the algorithm towards the optimal scheduling solution.

This problem not only requires the evaluation of discrete decisions but also involves the optimization of continuous work, hence the Q-value calculation has been optimized. By using a combination of greedy strategies and heuristic rules to calculate the target value, when the optimal action value function is found, the policy will become the optimal policy. Although it is impossible to solve the global problem directly in the continuous action space alone using DDPG, we can compromise by combining DDPG with Q-learning to obtain the Q-values while avoiding falling into local optima.

$$Q(s_t,a_t)\leftarrow Q(s_t,a_t)+\alpha[r_{t+1}+\gamma(Q(s_{t+1},\mu(s_{t+1};\theta^{\mu}))+\max_{a'\in A(s_{t+1})}Q(s_{t+1},a'))-Q(s_t,a_t)] \quad (14)$$

In Equation (14), α denotes the learning rate. A higher learning rate results in a greater impact of immediate return value and future reward value on the current Q value, and the more the agent can see the future results. However, the convergence speed will be slower. Additionally, γ represents the discount factor. The closer γ is to 0, the greater the decision-maker's preference for the immediate return value. Conversely, the closer is it to 1, the stronger the decision-maker's preference for the future reward value.

B. *Definition of details*

The primary focus of employing reinforcement learning to solve the re-entrant flow shop scheduling problem lies in effectively translating the problem into a reinforcement learning framework and use the corresponding algorithms to obtain optimized policy outcomes. This study constructs and refines the set of states, actions and feedback functions in the environment through the application of the Q-learning algorithm. In this algorithm, an agent employs a behavioral policy to select an action and updates the Q table to learn the target policy. A greedy strategy is implemented, wherein an action is chosen with a probability of 1 based on its action value, while an alternative action is selected randomly according to the behavioral policy. The target strategy employed in this problem adheres to the greedy strategy, which prioritizes the selection of the action with the highest value. The agent chooses the action and receives the reward r_t as Equation(14) shows, where α is a learning rate and $\alpha\in$ (0,1). When α is close to 1, the Q table is updated rapidly, while when α is close to 0, the Q table is not updated. When the discount factor γ is close to 1, this means that the value of future rewards is as high as the value of immediate rewards, while the value of immediate rewards is significantly higher when γ is close to 0. We will now define the characteristics of the re-entrant agents, including their states, actions and rewards in the flow shop.

Definition 1. State set

In the proposed scheduling method, each job and dual-gripper is treated as an agent, with their respective locations and statuses serving as the environmental states. Each job can be in one of the following three states: (1) The job has arrived at the assigned processing machine and is waiting to be released. (2) The job is currently being

processed on the machine, during which the dual-gripper are unable to perform grasping and releasing operations. (3) The job has been processed and is waiting to be picked up by the dual-gripper. When considering transportation speed, although the dual-gripper robots are in continuous states, there are four relatively discrete states that can be identified: (1) Both grippers are empty. (2) Both grippers are fully loaded. (3) The left gripper is fully loaded. (4) The right gripper is fully loaded. The four states are represented in matrix form as [0, 0], [1, 0], [0, 1], [1, 1]. Therefore, the workshop has a total of $12NM_{Machine}{}^{2}D_{Double\text{-}gripper}$ discrete environmental states without considering speed. Here N is the total number of jobs, $M_{Machine}$ is the number of machines, and $D_{Double\text{-}gripper}$ is the number of dual-gripper robots.

Definition 2. Action set

After observing the environmental states of the jobs and dual-gripper robots, the corresponding actions are executed: selecting the job and its corresponding machine, and selecting a dual-gripper robot and its corresponding machine. In s state, the set of available actions is denoted as A. Based on the predefined value of ε, the next action of the agent is determined. Generally, there are two types of actions: exploitation and exploration [31].

Exploitation is a strategy where the agent selects the optimal action among all known state-action pairs. This selection is guided by the principle of maximizing the action value. In a production scheduling context, this means that the agent will choose the action that, based on its current knowledge and experience, is expected to yield the best outcome. For instance, if the agent has learned that assigning a particular job to a specific machine at a given time, along with a certain dual - gripper robot for transportation, leads to shorter processing times and higher overall efficiency, it will choose this combination. In other words, when an agent chooses an action from the set of known actions, this decision-making behavior is termed exploitation. However, relying solely on exploitation may cause the agent to get stuck in local optima, missing out on potentially better solutions that could be discovered through exploration.

In balancing exploitation and exploration, the ε-greed strategy is a common approach. It introduces a small positive parameter ε (where $0< \varepsilon <1$), which represents the probability of the agent selecting a random action to explore unknown possibilities. Conversely, with a probability of 1 - ε, the agent opts for the action with the highest estimated value, thereby prioritizing exploitation [32].

The motion form of agents is defined as a set of optional actions, which in this study consists of the following seven typical heuristic rules:

(1) SPT (Shortest Processing Time): The process with the shortest processing time is preferred.

(2) LPT (Longest Processing Time): The process with the longest processing time is selected first.

(3) FCFS (First Come First Serve): The process that arrives at the job earliest is selected first.

(4) EFT (Early Finish Time): Priority is given to the process with the shortest finish time.

(5) LFT (Late Finish Time): The process with the longest finish time is selected first.

(6) MWKR (Most Work Remaining): The process that gives priority to the job with the longest remaining machining time.

(7) RP (Re-entry priority): The original re-entry process takes precedence over the re-entry workpiece that fails quality inspection and is reworked.

Based on these rules, the set of selectable actions in this study is defined as A= {a_1, a_2, a_3, a_4, a_5, a_6, a_7}.

Definition 3. Reward function

By utilizing the objective function as the reward signal, it is established that the smaller the value of the objective function, the larger the corresponding reward value This indicates that a smaller objective function value implies a better-selected action. The objective function is expressed as follows:

$$r = \frac{1}{\mu * makespan + (1-\mu) * delay} * \left\lceil \frac{2-H}{2} \right\rceil \tag{15}$$

Where H ($0 \leqslant H \leqslant 2$) represents the number of jobs gripped by the dual-gripper, and when H = 2, the job does not select any action.

Based on the above definitions, the re-entrant flow shop scheduling problem is mapped onto the Q-Learning algorithm, as described in Algorithm 1.

Algorithm 1: Q-DDPG algorithm for solving RHFSJSPD

Input: data set and parameters.
Output: Optimal job sequence and dual-gripper control strategy.
Initialize Q (s, a) for all states s and actions a
Initialize the DDPG critic network and actor network
Initialize best solution Best
For each episode:
  Initialize the state s
  Determine the objective function F
  While not finished (all artifacts):
    Initialize all action values
    Select a heuristic rule based on s state using a greedy strategy
    Execute this heuristic rule to obtain the sequence of remaining artifacts
    Get the next state s_{t+1} and the reward value r(1/F)
    Update Q (s, a) using Q-learning
    $Q(s, a) \leftarrow Q(s, a) + \alpha [r_t + \gamma (\max Q(s_{t+1}, a_{t+1})) - Q(s, a)]$
    If the action space is continuous:
      Use DDPG to update the policy for continuous actions
      Update the critic network using the sampled experiences
      Update the actor network to maximize the critic's output
    $s \leftarrow s_{t+1}$
  End while
  If F(s) < F(Best):
    Best ← s
End for
Return Best

The simulation results include the maximum completion time (Makespan) and total delay (TD). TD serves as a metric for comparing the job delay rate of the three algorithms. It can be employed to determine whether jobs are delayed during actual production, thereby helping to avoid unnecessary waste. Specifically, a higher value of TD indicates a higher job delay rate, signifying that the algorithm causes the jobs to remain on the production line for a longer period of time than necessary. The job delay rate is calculated by Equation (16) as follows:

$$TD = \frac{\sum_{i=1}^{n} \frac{p_i' - p_i}{p_i}}{n} \tag{16}$$

Where n is the total number of jobs. p_i is the sum of processing time that job i requires on each piece of equipment, and p_i is the actual time when job i reaches the output end. The evaluation criterion for the algorithm is obtained by

combining the maximum completion time and the job delay rate, with appropriate weights assigned to each. The calculation is carried out according to Equation (17) as follows:

$$P_i = 80 * \frac{2 * \bar{M} - M_i}{\bar{M}} + 20 * \frac{2 * \overline{TD} - TD}{\overline{TD}} \quad (17)$$

V. Case Studies

A. Test Instances Set

Given the absence of a standard test case for the RHFSJSPD problem, this study builds upon the Taillard benchmark example by incorporating the dual-gripper and reentrant characteristics. This study establishes three test cases, each comprising 10 instances, as detailed in Table II. All numerical values are integers within a specified range. The number of machines is the number of processing machines included in each process. The number of dual-gripper robots is the total number in the entire workshop. The processing time for each process operation is represented as an integer within the defined interval. Additionally, the distances between each adjacent machines and the distances between workstations associated with different adjacent processes, are also expressed as integers within the specified range. The number of all re-entrant situations caused by the fixed re-entrant processes and quality inspection rework following the initiation of production for each job, is likewise an integer within the defined interval. Since the loading and unloading time of dual-gripper robots constitutes less than 0.01% of the overall production cycle and is therefore considered negligible.

TABLE II
SUMMARY OF RHFSJSPD BENCHMARK INSTANCES

Scale	Small	Medium	Large
Stage number	5	10	15
Machine number	3	5	8
Processing time	6	13	20
Transition Distance	[10,15]	[10,15]	[10,15]
Reentry times	[2,4]	[3,7]	[4,12]

B. Parameters Set

In accordance with the findings presented in literature [33] and [34], the parameters for the Q-DDPG algorithm are set as: learning rate α=0.6, discount factor γ=0.6, and an exploration rate ε for the ε-greedy strategy ranging from 0.2 to 0.35. Additionally, the target network is configured to update every 100 steps, the experience replay buffer size is set to 512, and the batch size is determined to be 64.

To the best of our knowledge, no existing studies have been addressed the RHFSJSPD yet. Therefore, three algorithms developed based on the Q-learning algorithm were selected for comparative analysis (1) the Hyper-heuristic with Q-learning algorithm (HHQL) [35], (2) the improved artificial bee colony (ABC) algorithm with Q-learning (QABC) [36], (3) the multi-population memetic algorithm with Q-learning (MPMA-QL) [37]. HHQL demonstrates its ability to handle complex constraints and perform multi-objective optimization in complex problems such as multi-objective distributed blocked flow-shop scheduling. By combining the hyper-heuristic framework with Q-learning, it can explore superior scheduling solutions in various scenarios. QABC leverages the excellent global search ability of the ABC algorithm. When combined with Q-learning, it can better balance the exploration and exploitation processes, and shows certain advantages in dealing with permutation flow-shop scheduling problems involving large-scale data and complex task assignments. MPMA-QL stands out in solving combinatorial optimization problems of complex production systems, like distributed three-stage assembly hybrid flow-shop scheduling. Relying on the characteristics of multi-population and memetic algorithms, it can find high-quality solutions through information exchange among populations and local search capabilities. In order to ensure the fairness of the comparison experiment, the relevant general parameter Settings of all algorithms are the same as Q-DDPG, and their unique parameter setting schemes are derived from relevant literature. The termination criterion for each algorithm is defined as the completion of 250*n evaluations of the reward function. The implementation of these algorithms is conducted using PyCharm on a Windows 11 system, utilizing a PC equipped with a GPU, an 11th Generation Intel(R) Core processor, and 16.0 GB of RAM.

C. Comparative Analysis with Other Algorithms

In this study, the performance of all algorithms is meticulously evaluated based on solution quality and cumulative rewards. Given the multi-objective nature of the problem, Inverted Generational Distance (IGD) and Hypervolume (HV) are utilized as the primary metrics to measure solution quality. IGD gauges the average minimum distance from each point on the true Pareto front to the approximate set of non-dominated solutions identified by the algorithm. A smaller IGD value indicates that the algorithm's solutions are closer to the true Pareto front, signifying better-quality and more uniformly distributed solutions. The HV metric assesses the diversity of solutions by calculating the volume in the objective space enclosed by a set of non-dominated solutions and a reference point. A larger HV value implies that the algorithm can cover a broader range of the Pareto front, demonstrating stronger search capabilities and greater solution diversity.

To ensure the reliability and generality of the experimental results, the Q-DDPG algorithm and three comparable algorithms are executed 20 times on 30 test instances across three scales. The IGD and HV indicator results of all algorithms are presented in Table III and IV. To facilitate a more intuitive analysis of solution quality, the data in these tables are visualized as radar charts in Fig. 3. In Fig. 3 (a), each point represents the IGD value of the solution obtained by running an algorithm on an instance. Although a few small and medium-sized instances fail to attain the minimum IGD value, Q-DDPG still performs satisfactorily in most cases. Through a detailed analysis of each instance, it is discovered that in instances where Q-DDPG does not achieve the optimal IGD, there usually exist more intricate job sequences and machine configurations. For instance, in certain instances with a large number of jobs re-entering the same machine and uneven processing time distributions among jobs, Q-DDPG may require a longer time to converge to the optimal solution. However, as the problem scale expands, the superiority of

Q-DDPG becomes more evident. This is attributed to the hierarchical search architecture of the algorithm, which enables it to handle large-scale problems more effectively. It can efficiently decompose the problem and evade being trapped in local optima.

TABLE III
RESULTS OF IGD

No.	Q-DDPG	HHQL	QABC	MPMA-QL
1	0.1752	**0.1688**	0.2742	0.2752
2	0.1945	**0.1846**	0.2621	0.2951
3	**0.1862**	0.2245	0.2985	0.3365
4	**0.1562**	0.2654	0.3254	0.3950
5	**0.1236**	0.3025	0.3369	0.4420
6	**0.1502**	0.2245	0.3456	0.3589
7	**0.1026**	0.1985	0.3145	0.2987
8	0.1998	**0.1946**	0.2955	0.3549
9	**0.1753**	0.2412	0.2741	0.3654
10	0.2174	**0.2125**	0.3654	0.2298
11	**0.1562**	0.1984	0.3184	0.3485
12	**0.1526**	0.2541	0.3516	0.2564
13	**0.1456**	0.2954	0.4125	0.3258
14	0.2466	**0.2459**	0.4415	0.3184
15	**0.1523**	0.2874	0.3654	0.3654
16	**0.2056**	0.2218	0.2987	0.2597
17	0.2254	**0.2218**	0.3341	0.3122
18	**0.1812**	0.3025	0.2489	0.3511
19	**0.1845**	0.2977	0.2798	0.4599
20	**0.1842**	0.2705	0.4151	0.4181
21	**0.2874**	0.2814	0.4026	0.3952
22	**0.1741**	0.1951	0.3218	0.2476
23	**0.1954**	0.2814	0.3240	0.3618
24	**0.1523**	0.2577	0.3540	0.3659
25	**0.1982**	0.2241	0.2952	0.3059
26	**0.2001**	0.2256	0.2745	0.3482
27	**0.1743**	0.1874	0.3251	0.3654
28	**0.1985**	0.2652	0.2984	0.2892
29	**0.1455**	0.2147	0.3554	0.3358
30	**0.1301**	0.3254	0.3450	0.3628
Average	**0.1622**	0.2530	0.6358	0.3382

TABLE IV
RESULTS OF HV

No.	Q-DDPG	HHQL	QABC	MPMA-QL
1	0.7184	**0.8102**	0.6114	0.5514
2	**0.7752**	0.6122	0.6012	0.3398
3	**0.6810**	0.5023	0.4152	0.4418
4	0.6418	**0.7145**	0.4418	0.4178
5	0.6541	**0.6711**	0.3988	0.6052
6	**0.6184**	0.5820	0.4178	0.6147
7	**0.7103**	0.5541	0.5458	0.3681
8	0.7410	**0.7751**	0.6651	0.5141
9	**0.6641**	0.5514	0.4984	0.4419
10	**0.6684**	0.6101	0.4598	0.4788
11	**0.6661**	0.4451	0.3398	0.4366
12	**0.6411**	0.4985	0.5418	0.4452
13	**0.7155**	0.5029	0.6698	0.5417
14	**0.7014**	0.6647	0.7005	0.5457
15	**0.6441**	0.5995	0.5551	0.3395
16	**0.6815**	0.5841	0.5488	0.4321
17	0.6355	**0.6514**	0.5417	0.4417
18	**0.6471**	0.6112	0.5541	0.5514
19	**0.6772**	0.6174	0.6041	0.4158
20	**0.6458**	0.6011	0.5123	0.4418
21	**0.6698**	0.5980	0.5417	0.4552
22	**0.6148**	0.5471	0.5143	0.4711
23	**0.6543**	0.4988	0.4117	0.5419
24	**0.6871**	0.5941	0.4771	0.4552
25	**0.7105**	0.6126	0.5141	0.4516
26	**0.6443**	0.6025	0.5412	0.5144
27	**0.6543**	0.5654	0.4125	0.4871
28	**0.6652**	0.6201	0.4511	0.5118
29	**0.6228**	0.4551	0.6144	0.4185
30	**0.6780**	0.5881	0.6514	0.5147
Average	**0.6710**	0.6090	0.5318	0.4729

In Fig.3 (b), Q-DDPG attains the maximum HV value in 25 instances, encompassing all large-scale instances. By visualizing the distribution of solutions in the objective space, it can be observed that the solutions obtained by Q-DDPG cover a wider area than those of other algorithms and are more evenly distributed across multiple objective dimensions. This means that the set of non-dominated solutions found by Q-DDPG covers a wider area in the target space and is more evenly distributed. The solution obtained will not be excessively biased to a certain goal, but has a good performance in multiple goal dimensions.

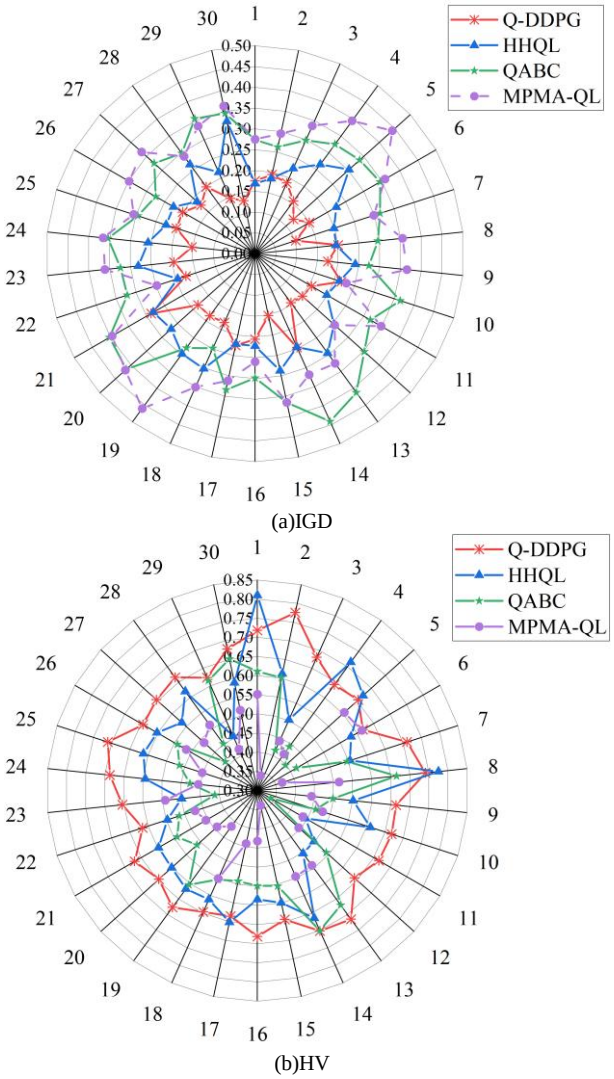


Fig. 3. Comparison of the quality of solutions obtained by the algorithms

In production scheduling research, evaluating solution quality is crucial as it affects both algorithm performance theoretically and actual production efficiency and profitability. Traditional indicators like IGD and HV are not direct enough for measuring scheduling schemes from a practical standpoint. Dual-gripper robots, key resources in automated production, directly impact material transportation efficiency and production line smoothness. Different scheduling algorithms lead to various load states of dual-gripper, significantly affecting production progress. High idle or single-gripper operation times result in resource waste and extended production cycles, while fully loaded dual-gripper enhance transportation efficiency and accelerate production. Conversely, a reasonable extension of the time when both grippers are fully loaded can enhance transportation efficiency and accelerate the production rhythm. This experiment compares the impacts of four

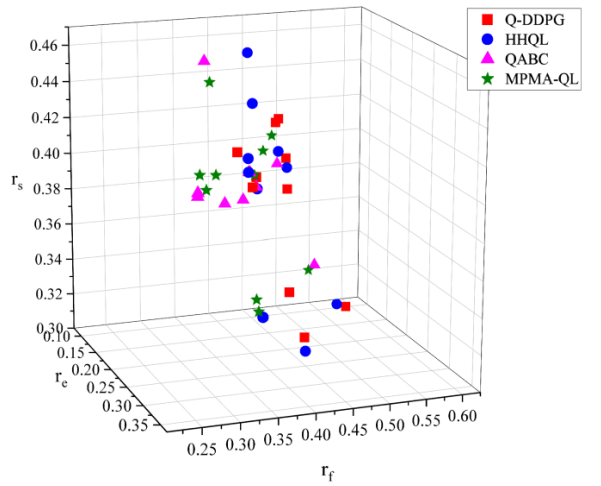
algorithms on the utilization rate of dual-gripper robots in problems of different scales. The aim is to verify whether the solutions obtained by these algorithms can achieve efficient resource allocation and smooth production processes in actual production from the perspective of key resource utilization. This approach enables a more comprehensive and in-depth evaluation of solution quality, providing a reference that is more closely aligned with the practical needs for algorithm selection in actual production scheduling.

During production, dual-gripper robots are crucial for material transportation, and their utilization rate affects the smoothness and efficiency of the production system. In terms of solution quality, an excellent scheduling scheme should ensure that dual-gripper robots are fully utilized in transportation tasks, reducing resource waste caused by idle time and increasing the proportion of fully loaded operations, thus accelerating material flow. Table V presents the duration proportion of three states: empty-loaded, single-gripper loaded, and full-loaded for dual-gripper under different algorithm across all scale instances. The minimum values of the empty-loaded proportion r_e and single-gripper loaded proportion r_s for all dual-gripper robots, as well as the maximum value of fully loaded proportion r_f, are highlighted in bold. To accurately assess the effectiveness of algorithms in dual-gripper scheduling, we calculated the proportion of the duration that all dual-gripper robots are in these states. The production process is discretized into time steps with a length of 1, and the total number of time steps is N. $s_e^d(n)$represents the state of the d-th dual-gripper at the n-th time step. When $s_e^d(n)$ takes the value of 0, the dual-gripper is in an unloaded state. When $s_e^d(n)$ takes the value of 1, it is in a single-gripper loaded state. When $s_e^d(n)$ takes the value of 2, the dual gripper is in a fully loaded state. The calculation of the idle time for each dual-gripper is shown in Equation (18). The calculation method for r_e, is shown in Equation (19). The calculation of the r_f and the r_s follows a similar approach.

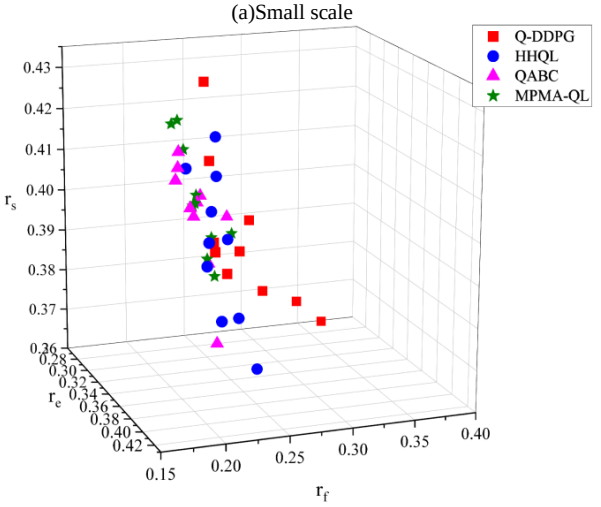
$$T_e^d = \sum_{n=1}^{N} S_e^d(n) \tag{18}$$

$$r_e = \frac{\sum_{d=1}^{D} T_e^d}{D \times C_{\max}} \tag{19}$$

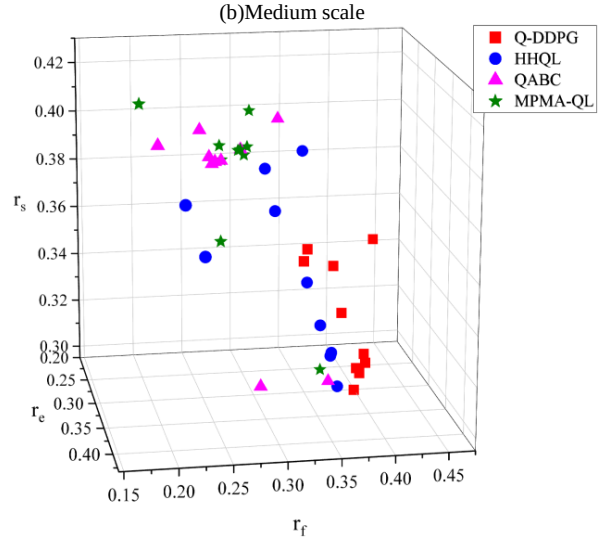
Additionally, to intuitively display the data distribution characteristics, three 3D scatter plots in Fig. 4 have been created. For each point, the smaller the mapped values on the r_e and r_s axes, and the larger the mapped value on the r_f axis, the higher the utilization rate corresponding to that point in the instance. For ease of comparison, the lower right half of the three-dimensional space is referred to as the advantage zone, and the points distributed in the advantage zone are called advantage points. Fig.4 (a) shows that when solving small-scale problems, the distribution of points for four algorithms concentrated in the advantage zone is relatively uniform. Among them, Q-DDPG and HHQL each have about 6 advantage points, and the distance between the advantage points of the two algorithms is also relatively close. By comparing the specific values in Table V, although Q-DDPG performs reasonably well in some indicators, considering the number of advantage points and the values, Q-DDPG does not show a significant advantage in dual gripper utilization compared to other algorithms in small-scale problems. This may be due to the low complexity of small-scale problems, making it difficult for the characteristics of Q-DDPG to stand out. Fig.4 (b) shows that when solving medium-scale problems, although some points of Q-DDPG are outside the advantage zone, the number of points in the advantage zone is significantly more than that of other algorithms, and the gap in r_f of the advantage points compared with other algorithms has widened. In Fig.4 (c), Q-DDPG achieved relatively optimal results for large-scale problems, significantly outperforming other algorithms in both quantity and quality.



(a)Small scale



(b)Medium scale



(c) Large scale

Fig. 4. Loading state time ratio of dual-gripper robots

The average processing time $\overline{p}_m$ and standard deviation σ_m of jobs on machines are closely related to machine utilization and solution quality in production scheduling. A shorter $\overline{p}_m$ means that machines can handle more jobs per unit of time, resulting in higher utilization rates. It also reflects efficient resource allocation in the scheduling and indicates a higher quality of the solution. A smaller σ_m suggests a more balanced workload for machines and more stable operation, ensuring the stability and continuity of production, which is a manifestation of high-quality solutions. Introducing $\overline{p}_m$ and σ_m can further demonstrate the superiority of the solution quality from the perspective of production resource utilization. The calculation method for $\overline{p}_m$ of each job on a machine is shown in Equation (20). Here, $t_{j,m}^{start}$ and $\mathrm{t}_{\mathrm{j,m}}^{\mathrm{end}}$ represent the start and end processing times of job j on machine m, respectively. The $\overline{p}_m$ is calculated as shown in Equation (21).

$$p_{j,m} = t_{j,m}^{end} - t_{j,m}^{start} \tag{20}$$

$$\overline{p}_m = \frac{\sum_{j=1}^{N} p_{j,m}}{N} \tag{21}$$

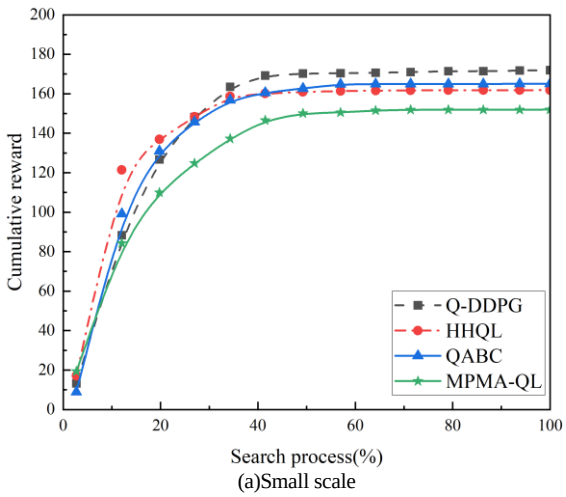
The σ_m calculated based on Equation (22) represents the dispersion of job processing times on machines. It reflects the stability of the algorithm. A smaller σ_m implies that the algorithm can stably control the processing time of jobs, reducing the potential for production plan chaos and delays caused by fluctuations in processing times.

$$\sigma_m = \sqrt{\frac{\sum_{j=1}^{N} (p_{j,m} - \overline{p}_m)^2}{N}} \tag{22}$$

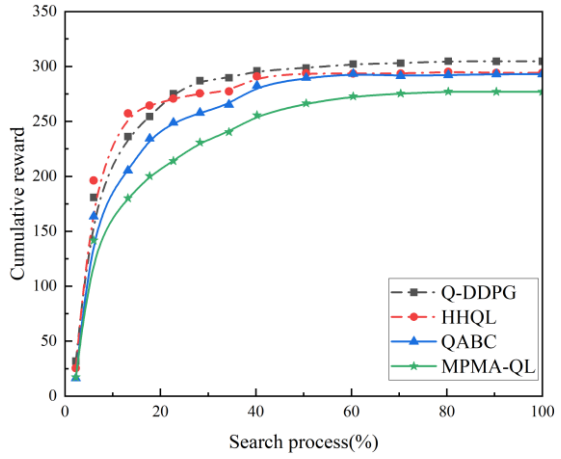
Table VI presents the average processing time and standard deviation of job processing times on some machines for the three different-scale instances of 3, 12, and 27. Some machines were randomly selected from each process. For example, instance 3 has 5 processes, so one machine was chosen from each process, and data for a total of 5 machines were recorded. During the solution of these three instances, the Q-DDPG algorithm achieved the smallest average processing time and standard deviation on all machines. This indicates that it has a clear advantage in improving machine utilization and ensuring production stability, thereby further corroborating that the scheduling plan obtained by this algorithm is superior in terms of solution quality.

In the RHFSJSPD, each decision made by the algorithm will have an impact on the subsequent production process. The cumulative reward reflects the total benefits obtained by the algorithm throughout the entire scheduling process. A higher cumulative reward indicates that the algorithm can make better decisions when scheduling jobs and allocating resources, thus achieving better overall performance in long-term operation. It comprehensively takes into account multiple factors such as production efficiency, resource utilization, and job delay, and serves as a comprehensive evaluation of the algorithm's performance. To further analyze the performance of Q-DDPG, Fig.5 presents the cumulative reward training curves of each algorithm on three different scale instances of 3, 12, and 27. In this curve, the horizontal axis represents the search process, that is, the proportion of the running time of the current algorithm in the total running time, while the vertical axis reflects the cumulative reward value at the corresponding search stage. It can be seen from Fig.5 (a) that the initial rising rate of Q-DDPG is relatively slower compared with HHQL and QABC. Delving into the underlying reasons, at the beginning of the training, HHQL and QABC tend to adopt more aggressive exploration strategies. They can quickly capture jobs with shorter processing times by virtue of their heuristic rules, thus enabling the cumulative reward to increase rapidly in a short time. For example, HHQL may preferentially select jobs with significantly shorter processing times for scheduling, and then obtain higher reward feedback in the initial stage. In contrast, Q-DDPG focuses on constructing a global search framework based on weighted feature representation and heuristic rules. At this stage, it lays more emphasis on the layout and planning of the overall search space rather than simply pursuing immediate reward growth. However, when the search process advances to around 35%, a significant change occurs. Q-DDPG successfully overtakes and finally obtains the highest cumulative reward. This crucial turning point strongly demonstrates that the combination of global and local search strategies adopted by Q-DDPG has significant advantages. It can efficiently integrate resources and optimize task allocation at a more macroscopic level, thus exhibiting a global search efficiency that surpasses other algorithms.

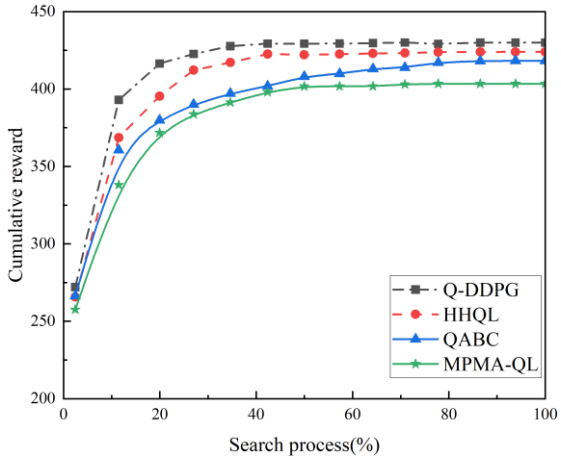
Further analysis of Fig.5 (b) and Fig.5 (c) shows that as the problem scale increases, the search efficiency advantage of Q-DDPG becomes more evident. The time point at which it surpasses the other three algorithms is significantly earlier, the overtaking process is faster, and the convergence speed is remarkably accelerated. The core reason for this is that in large-scale instances, the sharp increase in the number of jobs and machines places extremely high demands on the algorithm's task allocation and resource coordination capabilities. Thanks to its unique hierarchical design, Q-DDPG can accurately divide agents into multiple subgroups based on the different characteristics of jobs and machines, and customize appropriate search strategies for each subgroup. This refined hierarchical processing method enables Q-DDPG to handle complex task allocation and resource coordination tasks in an orderly manner. Thus, compared with other algorithms, it demonstrates higher efficiency and stability, can converge to a better solution space more quickly, and effectively guarantees the efficiency in large-scale problem. In conclusion, the performance of Q-DDPG in different instances fully demonstrates its advantages in search strategies and architecture design, providing a powerful support for solving complex scheduling problems in actual production.



(a)Small scale



(b) Medium scale



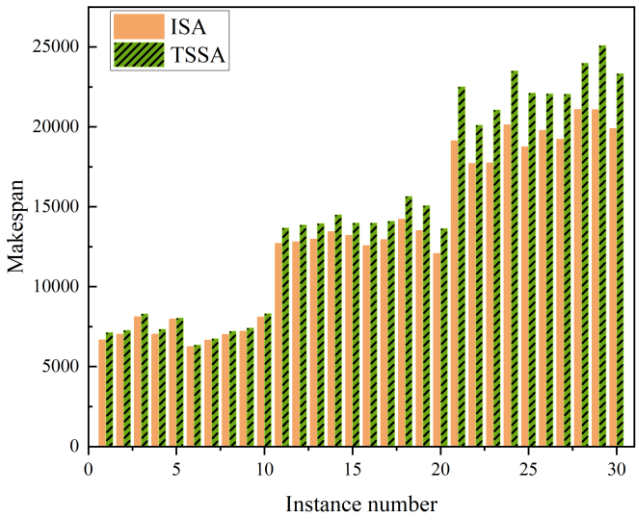
(c) Large scale

Fig. 5. Cumulative reward training curve

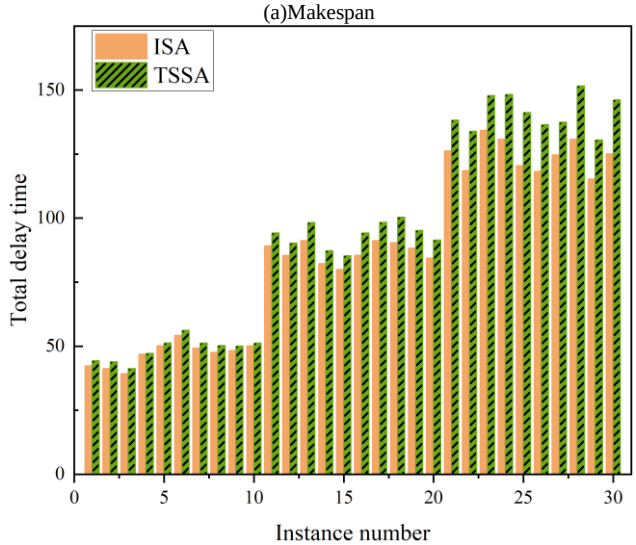
D. Comparative Analysis with Two-stage Separated Scheduling Approach

This section evaluates the efficacy of the integrated scheduling approach (ISA) through a comparison with the two-stage separated scheduling approach (TSSA). The TSSA is derived from the problem addressed by ISA. The actual production scheduling process is divided into two stages: job sequencing and dual-gripper control. In the first stage, when an order arrives, the scheduling system initially employs Q-DDPG to optimize the two objectives of minimizing the maximum completion time and total delay. Subsequently, the optimized job production sequence, job machine sequence, and other results are used as inputs for the next stage. Based on these inputs, Q-DDPG optimizes the loading scheme and speed control of dual-gripper robots to minimize the transfer time of dual-gripper robots, thereby minimizing the two objectives. The scheduling of TSSA concludes at this point. In contrast, the ISA adopts a more integrated scheduling method. When dealing with actual production scheduling, it does not separate the two stages of job sequencing and dual-gripper control. The ISA directly uses the Q-DDPG algorithm to simultaneously consider multiple objectives, such as the makespan, total tardiness, and the transfer time of the dual-gripper robot, and makes scheduling decisions within a unified framework. After jobs arrive, the ISA comprehensively analyzes factors such as the job production sequence, job-machine sequence, and the loading and speed control of the dual-gripper robot through the Q-DDPG algorithm, and optimizes the entire production scheduling process at once. This integrated scheduling method can better capture the interactions and influences among various factors and is expected to achieve better overall scheduling results, enabling better balance and optimization of multiple objectives in the same scheduling process.

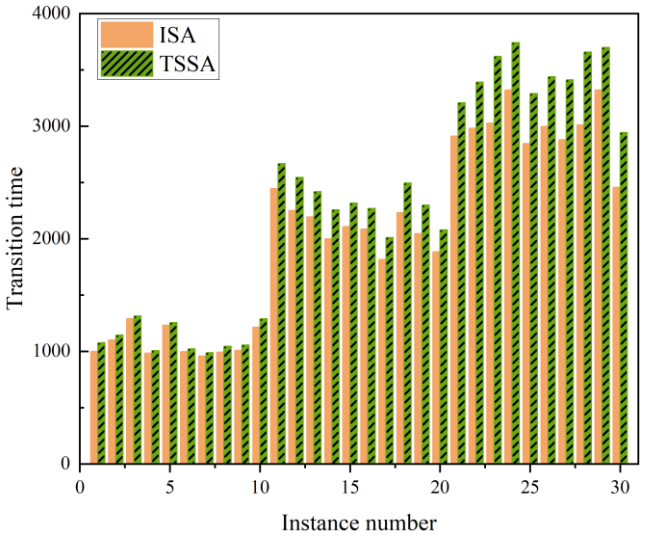
Table VII presents a detailed comparison of multiple crucial indicators, including IGD, HV, makespan, total delay time, and transition time. These indicators are obtained by running the two scheduling strategies on 30 test instances of three scales. Each instance involves 500 workpieces. The optimal values for each indicator are emphasized in bold. Without exception, the ISA consistently outperforms the TSSA in terms of IGD and HV across all instances. The ISA demonstrates an average reduction of 9.41% in makespan compared to the TSSA when addressing all instances. Furthermore, upon completion of the scheduling process, the ISA exhibits an average decrease of 16.14% in total tardiness and 10.2% in total transportation time relative to the TSSA. Fig.6 (a), (b), and (c) visually illustrate the comparative performance of these two scheduling strategies in terms of makespan, total delay time, and transition time. Notably, ISA consistently yields lower values than the TSSA across all evaluated indicators. As the scale of the problem increases, the disparity between the two strategies becomes increasingly pronounced.



(a)Makespan



(b)Total delay time



(c)Transition time

Fig. 6. Comparison of the results of various indicators of ISA and TSSA

This outcome can be attributed to the fundamental differences in their design. ISA inherently integrates job processing and dual-gripper scheduling in a unified framework. This integration allows for a more comprehensive consideration of the interactions between different aspects of the production process. For example, during the optimization process, ISA can simultaneously adjust job sequences and dual gripper operations based on the real time status of the production line. In contrast, TSSA's two-stage separation may lead to suboptimal decisions. In the job sequencing stage, decisions might be made without fully considering the subsequent impact on dual gripper control. Similarly, in the dual gripper control stage, the optimization might be limited by the fixed job sequences determined in the previous stage.

Overall, the superiority of ISA in solving the RHFSJSPD problem is clearly demonstrated, and this advantage becomes more pronounced as the problem scale enlarges. This finding provides valuable insights for the selection and improvement of scheduling strategies in practical manufacturing scenarios.

VI. Conclusions

This study focuses on the multi-objective scheduling problem in a re-entrant hybrid flow shop with dual-gripper robots, emphasizing integrated production and transportation management. The problem, complex with dynamic, discrete, and continuous features, is modeled using a mixed-integer linear programming model (RHFSJSPD) aiming to minimize makespan and total delay. To solve this problem, the innovative Q-DDPG algorithm is proposed. It combines the advantages of Q-learning and DDPG, constructing a hierarchical search architecture for RHFSJSPD. This enables coordinated macroscopic and microscopic scheduling, effectively handling high-dimensional state spaces and complex decision-making, marking a pioneering approach in similar research.

To comprehensively evaluate the Q-DDPG algorithm's performance, we conducted extensive numerical experiments. The results demonstrate its superiority over other Q-learning-based algorithms in key metrics like IGD and HV, which measure solution quality. Specifically, Q-DDPG achieved an average IGD of 0.1622 and HV of 0.6710, outperforming HHQL, QABC, and MPMA-QL. The utilization rates of machines and dual-gripper robots further demonstrate the superiority of the solution quality in practical production. For the dual-gripper robots, the average empty-loaded and single-gripper loaded ratios are lower than those of the other three algorithms, while the average full-loaded ratio reaches 37%, significantly higher than the others. Across all three scales, the algorithm achieves the smallest average processing time and standard deviation on machines. These results demonstrate its practical effectiveness beyond theoretical metrics. The algorithm also shows faster convergence and stability in large-scale scenarios, avoiding local optima. Compared to the TSSA, the ISA reduced makespan by 9.41%, total delay by 16.14%, and transportation time by 10.20% across 30 instances with 500 workpieces. This proves ISA's ability to better coordinate job processing and dual-gripper scheduling, improving production efficiency and overall system performance.

The scheduling strategy's superiority in this paper is mainly due to: (1) A hierarchical search mechanism that combines Q-learning for macro scheduling and DDPG for micro control, with jobs divided into subgroups for global search and target networks/experience replay used for local search to enhance convergence and dual-gripper control accuracy. (2) Integration of greedy strategies and heuristic rules in Q-learning, which improves search efficiency and provides diverse decision-making bases to avoid local optima. (3) A well-designed state space composed of continuous and discrete variables representing jobs and dual-gripper robots positions and states, with reduced complexity through weighted feature representation. (4) A comprehensive action space covering key scheduling operations, enabling the algorithm to make decisions in different production environments. (5) Adoption of the ISA framework for collaborative job and dual gripper processing, optimizing workflow and resource utilization to improve overall production efficiency.

This study has several limitations, which can be highlighted in three main aspects. First, in terms of the problem scope, the current research focuses on RHFSJSPD under ideal production conditions. It neglects uncertainties like workpiece size variations, and dynamic disruptions. This simplification restricts the model's ability to align with the complexities of actual manufacturing. Second, regarding algorithmic adaptability, although the proposed Q-DDPG framework shows efficiency in controlled environments, its computational requirements and convergence stability might deteriorate in scenarios with frequent disturbances or in large-scale instances that demand real-time responsiveness. Third, considering sustainability, the model does not include energy consumption metrics, which are crucial for green manufacturing initiatives.

To address these limitations, future research will explore the following directions:(1) Expanding the problem scope to incorporate disruptions, such as machine failures, dynamic order cancellations, and preventive maintenance events. (2) Developing hybrid frameworks that combine deep reinforcement learning with meta-heuristics. This aims to balance the exploration-exploitation trade-offs and enhance the robustness of solutions. (3) Integrating energy-aware

objectives into the scheduling model, like minimizing idle machine power consumption or optimizing dual-gripper motion trajectories to reduce kinetic energy waste.

By filling these gaps, future work will move RHFSJSPD solutions towards greater robustness, computational efficiency, and sustainability, ultimately supporting intelligent and eco-friendly manufacturing systems.

Table V

THE DURATION RATIO OF THREE LOADING STATES OF DUAL-GRIPPER ROBOTS

N0.	Q-DDPG			HHQL			QABC			MPMA-QL		
	r_e	r_f	r_s	r_e	r_f	r_s	r_e	r_f	r_s	r_e	r_f	r_s
1	0.2118	0.3740	0.4142	**0.2057**	**0.3925**	**0.4018**	0.3185	0.2693	0.4122	0.2718	0.3154	0.4128
2	0.2098	0.3961	**0.3941**	**0.1949**	**0.3975**	0.4076	0.3218	0.2674	0.4108	0.3017	0.2798	0.4185
3	**0.2218**	**0.4417**	**0.3365**	0.2936	0.3676	0.3388	0.2771	0.3246	0.3983	0.2918	0.3599	0.3483
4	**0.1937**	**0.4095**	0.3968	0.2017	0.4066	**0.3917**	0.2048	0.3935	0.4017	0.3011	0.2886	0.4103
5	**0.2287**	**0.4597**	0.3116	0.2381	0.4561	**0.3058**	0.2389	0.3683	0.3928	0.2971	0.3601	0.3428
6	**0.1382**	**0.4826**	**0.3792**	0.1394	0.4315	0.4291	0.2018	0.4054	0.3928	0.1958	0.4059	0.3983
7	**0.1049**	**0.4829**	**0.4122**	0.1059	0.4413	0.4528	0.2041	0.3327	0.4632	0.2104	0.3369	0.4527
8	**0.1137**	**0.4935**	0.3928	0.1201	0.4915	0.3884	0.1382	0.4677	0.3941	0.1253	0.4665	0.4082
9	**0.0953**	**0.4920**	0.4127	0.1248	0.4764	0.3988	0.2028	0.3961	0.4011	0.1573	0.4376	0.4051
10	**0.1124**	**0.5849**	**0.3027**	0.1305	0.5611	0.3084	0.1451	0.5192	0.3357	0.1629	0.5007	0.3364
11	**0.3827**	**0.2251**	**0.3922**	0.3931	0.2057	0.4012	0.4011	0.2013	0.3976	0.3865	0.1846	0.4289
12	**0.3027**	**0.3252**	**0.3721**	0.3457	0.2345	0.4198	0.4028	0.1771	0.4201	0.3947	0.1762	0.4291
13	**0.3874**	**0.2122**	**0.4004**	0.3937	0.1876	0.4187	0.4017	0.1896	0.4087	0.3912	0.1966	0.4122
14	**0.3628**	**0.2209**	0.4163	0.3789	0.2146	**0.4065**	0.4027	0.1865	0.4108	0.3987	0.1905	0.4108
15	**0.3891**	**0.2125**	**0.3984**	0.4038	0.1989	0.3973	0.4198	0.1984	0.3818	0.3956	0.1935	0.4109
16	**0.3397**	**0.2281**	**0.4322**	0.4065	0.2088	0.3847	0.3865	0.2021	0.4114	0.3898	0.1876	0.4226
17	**0.3346**	**0.2675**	0.3979	0.3851	0.2331	**0.3818**	0.3964	0.1807	0.4229	0.3987	0.2071	0.3942
18	**0.3588**	**0.2473**	**0.3939**	0.3597	0.2281	0.4122	0.4097	0.1721	0.4182	0.3884	0.2098	0.4018
19	**0.3448**	**0.2734**	**0.3818**	0.3693	0.2323	0.3984	0.3927	0.1966	0.4107	0.4011	0.2002	0.3987
20	**0.2752**	**0.3627**	**0.3621**	0.3818	0.2495	0.3687	0.3618	0.2354	0.4028	0.3621	0.2391	0.3988
21	**0.3062**	**0.3977**	**0.2961**	0.3221	0.3767	0.3012	0.3293	0.3656	0.3051	0.3327	0.3566	0.3107
22	**0.2924**	**0.3545**	**0.3531**	0.3957	0.2147	0.3896	0.3517	0.2367	0.4116	0.3772	0.2507	0.3721
23	**0.2787**	**0.4157**	**0.3056**	0.3133	0.3731	0.3136	0.3536	0.2478	0.3986	0.3427	0.2589	0.3984
24	**0.2924**	**0.4073**	**0.3003**	0.3128	0.3621	0.3251	0.3536	0.2451	0.4013	0.3387	0.2579	0.4034
25	**0.2843**	**0.3912**	**0.3245**	0.3078	0.3501	0.3421	0.3432	0.2586	0.3982	0.3217	0.2795	0.3988
26	**0.2718**	**0.3861**	**0.3421**	0.3017	0.3102	0.3881	0.3187	0.2826	0.3987	0.3176	0.2861	0.3963
27	**0.3017**	**0.3485**	**0.3498**	0.3981	0.232	0.3699	0.3497	0.2515	0.3988	0.3985	0.1728	0.4287
28	**0.2937**	**0.4036**	**0.3027**	0.3158	0.3714	0.3128	0.3988	0.2828	0.3184	0.3189	0.2824	0.3987
29	**0.2812**	**0.4165**	**0.3023**	0.3098	0.3181	0.3721	0.3977	0.1895	0.4128	0.3108	0.2907	0.3985
30	**0.2138**	**0.4434**	**0.3428**	0.2531	0.3594	0.3875	0.2656	0.3315	0.4029	0.2929	0.2968	0.4103
Average	**0.2641**	**0.3719**	**0.3640**	0.2934	0.3294	0.3772	0.3230	0.2792	0.3978	0.3191	0.2823	0.3986

Table VI

AVERAGE PROCESSING TIME AND STANDARD DEVIATION OF MACHINES

Scale	No.	$\bar{p}_m$				σ_m			
		Q-DDPG	HHQL	QABC	MPMA-QL	Q-DDPG	HHQL	QABC	MPMA-QL
Small	1	**12.2314**	12.3425	12.7093	12.7223	**1.2231**	1.2626	1.4247	1.4427
	2	**13.2842**	12.3374	12.3029	13.4434	**1.3284**	1.2621	1.3792	1.5245
	3	**10.0502**	10.0884	10.2241	10.2094	**1.0050**	1.0320	1.1461	1.1577
	4	**11.5186**	11.5568	11.6926	11.6778	**1.1519**	1.1707	1.3119	1.3359
	5	**12.6170**	12.6552	12.7910	12.7762	**1.2617**	1.2921	1.4339	1.4488
Medium	1	**12.2893**	12.3994	12.4633	12.4485	**1.2289**	1.2933	1.3971	1.4142
	2	**12.7012**	12.8113	12.8752	12.8604	**1.2701**	1.3106	1.4472	1.4584
	3	**12.6819**	12.7820	12.8559	12.8411	**1.2682**	1.3204	1.4411	1.4626
	4	**11.8864**	11.9865	12.0604	12.0456	**1.1886**	1.2262	1.3520	1.3660
	5	**10.6970**	10.7971	10.8710	10.8562	**1.0697**	1.1045	1.2186	1.2311
	6	**13.0392**	13.1393	13.2132	13.1984	**1.3039**	1.3441	1.4812	1.6287
	7	**15.0905**	15.2006	15.2645	15.2497	**1.5090**	1.5550	1.7111	1.7293
	8	**15.4095**	15.4194	15.6035	15.5887	**1.5430**	1.5774	1.7492	1.7678
	9	**12.3847**	12.4948	12.5587	12.5439	**1.2385**	1.2782	1.4078	1.4225
	10	**11.9003**	11.9192	12.1243	12.1095	**1.1950**	1.2193	1.3591	1.3974
Large	1	**10.6634**	10.7155	10.7762	10.8226	**1.0663**	1.0748	1.2080	1.2273
	2	**10.1564**	10.2085	10.2692	10.3156	**1.0156**	1.0647	1.1512	1.1698
	3	**10.1256**	10.1777	10.2384	10.2848	**1.0126**	1.0412	1.1477	1.1663
	4	**11.3253**	11.3370	11.4581	11.5045	**1.1345**	1.2731	1.2844	1.3046
	5	**12.8643**	12.8860	13.0071	13.0535	**1.2894**	1.3182	1.4581	1.4803
	6	**12.2708**	12.2825	12.4036	12.4500	**1.2291**	1.2688	1.3904	1.4118
	7	**12.8227**	12.9159	12.9355	12.9819	**1.2823**	1.3213	1.4501	1.4722
	8	**15.5091**	15.6023	15.6219	15.6683	**1.5509**	1.6008	1.7512	1.7768
	9	**12.5381**	12.6313	12.6702	12.6973	**1.2538**	1.2922	1.4203	1.4399
	10	**15.2149**	15.3081	15.3470	15.3741	**1.5215**	1.6273	1.7204	1.7434
	11	**12.8101**	12.8330	12.9722	12.9993	**1.2840**	1.3000	1.4542	1.4741
	12	**12.7275**	12.7404	12.8796	12.9067	**1.2748**	1.3033	1.4451	1.4636
	13	**14.2151**	14.3078	14.3472	14.3743	**1.4215**	1.5066	1.6083	1.6300
	14	**12.0745**	12.1672	12.0798	12.2337	**1.2074**	1.2447	1.3541	1.3873
	15	**13.6069**	13.6996	13.6122	13.7661	**1.3607**	1.4083	1.5286	1.6161

TABLE VII
COMPARISON RESULTS BETWEEN THE ISA AND TSSA

No.	IGD		HV		Improve		
	ISA	TSSA	ISA	TSSA	Makespan	Total delay rate	Transition time
1	**0.1721**	0.4112	**0.7745**	0.2284	6.84%	4.72%	7.60%
2	**0.1029**	0.4956	**0.8345**	0.2154	3.63%	6.57%	4.01%
3	**0.0913**	0.3958	**0.7719**	0.3019	2.30%	5.09%	2.03%
4	**0.1128**	0.4821	**0.8016**	0.2788	4.54%	0.81%	2.34%
5	**0.1354**	0.5146	**0.8802**	0.1981	0.81%	2.13%	1.76%
6	**0.0852**	0.6108	**0.8451**	0.2098	1.76%	3.65%	2.49%
7	**0.1225**	0.5417	**0.8474**	0.2446	1.35%	4.02%	3.24%
8	**0.1305**	0.5477	**0.8023**	0.2706	2.93%	5.64%	5.34%
9	**0.1547**	0.4981	**0.7981**	0.3981	2.60%	3.73%	4.66%
10	**0.1641**	0.4881	**0.7759**	0.2981	2.68%	2.19%	6.11%
11	**0.1145**	0.5433	**0.8854**	0.1981	7.62%	5.76%	9.04%
12	**0.1551**	0.5812	**0.8941**	0.1058	8.27%	5.51%	13.07%
13	**0.1108**	0.5788	**0.8556**	0.1455	7.50%	7.67%	10.15%
14	**0.1406**	0.5624	**0.8664**	0.1654	7.94%	6.09%	12.91%
15	**0.1351**	0.5981	**0.8154**	0.1681	5.80%	6.59%	9.83%
16	**0.1084**	0.6009	**0.8446**	0.1456	11.14%	10.30%	8.77%
17	**0.1413**	0.5881	**0.8561**	0.1145	8.83%	7.79%	10.80%
18	**0.1144**	0.5618	**0.7998**	0.2844	10.12%	10.93%	11.84%
19	**0.1325**	0.4987	**0.7184**	0.2954	11.66%	7.87%	12.36%
20	**0.1288**	0.4788	**0.8825**	0.1684	12.99%	8.40%	10.29%
21	**0.1415**	0.5124	**0.8449**	0.1554	17.63%	9.52%	10.17%
22	**0.0981**	0.6156	**0.8585**	0.1478	13.54%	12.93%	13.76%
23	**0.0788**	0.6458	**0.8458**	0.1624	18.66%	10.12%	19.60%
24	**0.1535**	0.4255	**0.8908**	0.1097	16.82%	13.30%	12.64%
25	**0.1544**	0.5144	**0.7988**	0.2981	17.88%	17.18%	15.54%
26	**0.1576**	0.5122	**0.7708**	0.3025	11.54%	15.39%	14.69%
27	**0.1473**	0.5246	**0.8544**	0.2145	14.71%	10.28%	18.47%
28	**0.1228**	0.5711	**0.8546**	0.2113	13.72%	15.90%	21.44%
29	**0.0998**	0.6811	**0.8045**	0.2450	19.11%	13.27%	11.37%
30	**0.1151**	0.5014	**0.8245**	0.1985	17.34%	16.87%	19.77%
Average	**0.1274**	0.5361	**0.8299**	0.2160	9.41%	16.14%	10.20%

REFERENCES

[1] H. H. Miyata and M. S. Nagano, "The blocking flow shop scheduling problem: A comprehensive and conceptual review," Expert Systems with Applications, vol. 137, pp130-156, 2019

[2] J. S. Neufeld, S. Schulz and U. Buscher, "A systematic review of multi-objective hybrid flow shop scheduling," European Journal of Operational Research, vol. 309, no. 1, pp1-23, 2023

[3] W. Jia, Z. Jiang and Y. Li, "Combined scheduling algorithm for re-entrant batch-processing machines in semiconductor wafer manufacturing," International Journal of Production Research, vol. 53, no. 6, pp1866-1879, 2015

[4] T. S. Yu and M. Pinedo, "Flow shops with reentry: Reversibility properties and makespan optimal schedules," European Journal of Operational Research, vol. 282, no. 2, pp478-490, 2020

[5] B. Jeong and Y. D. Kim, "Minimizing total tardiness in a two-machine re-entrant flow shop with sequence-dependent setup times," Computers & Operations Research, vol. 47, pp72-80, 2014

[6] S. Yoon and S. Jeong, "Line balancing strategy for re-entrant manufacturing," IEEE Transactions on Semiconductor Manufacturing, vol. 31, no. 1, pp42-51, 2017

[7] C. Zhang, Q. Gao, M. V. Basin, J. Lü and H. Liu, "Robust control of multi-line re-entrant manufacturing plants via stochastic continuum models," IEEE Transactions on Automation Science and Engineering, vol. 21, no. 4, pp4923-4935, 2023

[8] C. Sriskandarajah and B. Shetty, "A review of recent theoretical development in scheduling dual-gripper robotic cells," International Journal of Production Research, vol. 56, no. 1-2, pp817-847, 2018

[9] M. Hajiaghaei-Keshteli and M. Aminnayeri, "Solving the integrated scheduling of production and rail transportation problem by Keshtel algorithm," Applied Soft Computing, vol. 25, pp184-203, 2014

[10] H. Tang, J. Zhou, Y. Shao and Z. Yang, "Hybrid flow-shop scheduling problems with missing and re-entrant operations considering process scheduling and production of energy consumption," Sustainability, vol. 15, no. 10, pp7982, 2023

[11] S. Aminzadegan, M. Tamannaei and M. Fazeli, "An integrated production and transportation scheduling problem with order acceptance and resource allocation decisions," Applied Soft Computing, vol. 112, pp107770, 2021

[12] Y. Liu, J. Fan, L. Zhao, W. Shen and C. Zhang, "Integration of deep reinforcement learning and multi-agent system for dynamic scheduling of re-entrant hybrid flow shop considering worker fatigue and skill levels," Robotics and Computer-Integrated Manufacturing, vol. 84, pp102605, 2023

[13] H. J. Kim and J. H. Lee, "Scheduling of dual-gripper robotic cells with reinforcement learning," IEEE Transactions on Automation Science and Engineering, vol. 19, no. 2, pp1120-1136, 2021

[14] J. P. Huang, Q. K. Pan and Z. H. Miao, "Effective constructive heuristics and discrete bee colony optimization for distributed flow shop with setup times," Engineering Applications of Artificial Intelligence, vol. 97, pp104016, 2021

[15] C. Chamnanlor, K. Sethanan, M. Gen and C. Chien, "Embedding ant system in genetic algorithm for re-entrant hybrid flow shop scheduling problems with time window constraints," Journal of Intelligent Manufacturing, vol. 28, pp1915-1931, 2017

[16] J. Mlekusch and R. F. Hartl, "The dual-resource-constrained re-entrant flexible flow shop a constraint programming approach and a hybrid genetic algorithm," International Journal of Production Research, pp1-22, 2024

[17] J. Dong and C. Ye, "Green scheduling of distributed two-stage reentrant hybrid flow shop considering distributed energy resources and energy storage system," Computers & Industrial Engineering, vol. 169, pp108146, 2022

[18] S. M. Mousavi, I. Mahdavi, J. Rezaeian and M. Zandieh, "An efficient bi-objective algorithm to solve re-entrant hybrid flow shop scheduling with learning effect and setup times," Operational Research, vol. 18, pp123-158, 2018

[19] S. Topaloglu and G. Kilincli, "A modified shifting bottleneck heuristic for the reentrant job shop scheduling problem with makespan minimization," The International Journal of Advanced Manufacturing Technology, vol. 44, pp781-794, 2009

[20] R. Hu, X. Wu, B. Qian, J. Mao and H. Jin, "Differential Evolution Algorithm Combined with Uncertainty Handling Techniques for Stochastic Reentrant Job Shop Scheduling Problem," Complexity, vol. 2022, no. 1, pp9924163, 2022

[21] E. A. Eigbe, B. De Schutter, M. Nasri and N. Yorke-Smith, "Sequence-and time-dependent maintenance scheduling in twice re-entrant flow shops," IEEE Access, vol. 11, pp103461-103475, 2023

[22] K. C. Ying, S. W. Lin and S. Y. Wan, "Bi-objective reentrant hybrid flow shop scheduling: an iterated Pareto greedy algorithm," International Journal of Production Research, vol. 52, no. 19, pp5735-5747, 2014

[23] J. Hurink and S. Knust, "Tabu search algorithms for job-shop problems with a single transport robot," European Journal of Operational Research, vol. 162, no. 1, pp99–111, 2005
[24] J. Carlier, M. Haouari, M. Kharbeche and A. Moukrim, "An optimization-based heuristic for the robotic cell problem," European Journal of Operational Research, vol. 202, no. 3, pp636-645, 2010
[25] H. J. Paul, C. Bierwirth and H. Kopfer, "A heuristic scheduling procedure for multi-item hoist production lines," International Journal of Production Economics, vol. 105, no. 1, pp54-69, 2007
[26] A. Che, H. Hu, M. Chabrol and M. Gourgand, "A polynomial algorithm for multi-robot 2-cyclic scheduling in a no-wait robotic cell," Computers & Operations Research, vol. 38, no. 9, pp1275-1285, 2011
[27] I. G. Drobouchevitch, S. P. Sethi and C. Sriskandarajah, "Scheduling dual gripper robotic cell: One-unit cycles," European Journal of Operational Research, vol. 171, no. 2, pp598-631, 2006
[28] H. N. Geismar, L. M. A. Chan, M. Dawande and C. Sriskandarajah, "Approximations to Optimal k-Unit Cycles for Single-Gripper and Dual-Gripper Robotic Cells," Production and Operations Management, vol. 17, no. 5, pp551-563, 2008
[29] H. Gultekin, B. Coban and V. E. Akhlaghi, "Cyclic scheduling of parts and robot moves in m-machine robotic cells," Computers & Operations Research, vol.90, pp161-172, 2018
[30] M. Dawande, H. N. Geismar, M. Pinedo and C. Sriskandarajah, "Throughput optimization in dual-gripper interval robotic cells," IIE Transactions, vol. 42, no. 1, pp1-15, 2009
[31] Y. E. Hou, W. Gu, C. Wang, K. Yang and Y. Wang, "A Selection Hyper-heuristic based on Q-learning for School Bus Routing Problem," IAENG International Journal of Applied Mathematics, vol. 53, no. 4, pp817-825, 2022
[32] A. Boumaalif and O. Zytoune, "Power Control Based on Safe Q Learning for D2D Communication," IAENG International Journal of Applied Mathematics, vol. 53, no. 2, pp518-523, 2023
[33] S. Yeganeh, A. B. Sangar and S. Azizi, "A novel Q-learning-based hybrid algorithm for the optimal offloading and scheduling in mobile edge computing environments," Journal of Network and Computer Applications, vol. 214, pp103617, 2023
[34] C. L. Liu, C. C. Chang and C. J. Tseng, "Actor-critic deep reinforcement learning for solving job shop scheduling problems," IEEE Access, vol. 8, pp71752-71762, 2020
[35] F. Zhao, S. Di and L. Wang, "A hyper-heuristic with Q-learning for the multi-objective energy-efficient distributed blocking flow shop scheduling problem," IEEE Transactions on Cybernetics, vol. 53, no. 5, pp 3337-3350, 2022
[36] H. Li, K. Gao, P. Y. Duan, J. Q. Li and L. Zhang, "An improved artificial bee colony algorithm with Q-learning for solving permutation flow-shop scheduling problems," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 5, pp2684-2693, 2022
[37] Y. Jia, Q. Yan and H. Wang, "Q-learning driven multi-population memetic algorithm for distributed three-stage assembly hybrid flow shop scheduling with flexible preventive maintenance," Expert Systems with Applications, vol. 232, pp120837, 2023

Jiajia Wang received B.Eng. degree in Electrical and Intelligent in Nanjing Tech University in 2022. She is Currently pursuing a Master's degree in Mechanical Engineering at the College of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, China. Her research interest is intelligent manufacturing.

Hongming Zhou is a Professor of College of Mechanical and Electrical Engineering, Wenzhou University, China. He received a PhD degree in Industrial Engineering from the Jiangsu University, China. His research interests include manufacturing system modeling and simulation, virtual prototyping and industrial engineering, and digital design.

Jian Guo is an associate professor of College of Mechanical and Electrical Engineering, Wenzhou University, China. He received a PhD degree in Mechanical Engineering from Northwestern Polytechnical University. His research interests include production system scheduling and intelligent manufacturing systems.

Hongwei Si received a master's degree in mechanical engineering from College of Mechanical and Electrical Engineering, Wenzhou University, China. His main research interest is intelligent manufacturing.

Xu Chen is currently a senior business manager in Senken Group CO., LTD, Wenzhou, Zhejiang, China.

Minhai Zhang is currently a senior business manager, a first-level human resources manager, and a patent engineer in Senken Group CO., LTD, Wenzhou, Zhejiang, China.

Yaqi Zhang received a Bachelor's degree in Electronic Information Engineering from Qiqihar University in 2023. She is Currently pursuing a Master's degree in Mechanical Engineering at the School of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, China. Her main research interest is intelligent manufacturing.

Guixuan Zhou received a Bachelor's degree in Mechanical design, manufacture, and automation in Hubei Polytechnic University in 2022. He is Currently pursuing a Master's degree in Mechanical Engineering at the School of Mechanical and Electrical Engineering, Wenzhou University, Wenzhou, China. His main research interest is intelligent manufacturing.