Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 9
AIDML 2025

Application of Ant Colony Algorithm and Genetic Algorithm in Multi-Objective Dynamic Equalization Scheduling of AGVs

Lu Liang [1], Mingyuan Liu [2,*] and Ziyi Xiong [2]

[1] School of Information Engineering, Lanzhou Vocational Technical College, Lanzhou, China

[2] Innovation Academy for Precision Measurement Science and Technology, CAS, Wuhan, China

* Corresponding Author Email: lmyszq1017@163.com

Abstract. In this paper, an optimization method based on heuristic genetic algorithm is proposed for AGV multi-objective dynamic balanced scheduling. Firstly, the warehouse environment is constructed by raster modeling method, and the AGV path planning is transformed into a node traversal situation in a two-dimensional coordinate system, and the spatial relationship of multiple elements is defined. Then, a single-objective constraint model is established to minimize the longest path AGV picking time as the objective, and the objective function is constructed by combining the Manhattan distance and task processing time. Then the ant colony algorithm is introduced to solve the path planning, and the optimization efficiency of the algorithm is improved by dynamically adjusting the parameters. Further, a multi-objective scheduling model is constructed, taking into account the three indexes of total path length, task load balance and maximum time consumption, and a genetic algorithm is used to solve the problem. The task allocation strategy is optimized through a series of operations, and the fitness function is designed to achieve multi-objective transformation. This research provides theoretical support and practical reference for the efficient scheduling of unmanned warehousing system.

Keywords: AGV scheduling; path planning; ant colony algorithm; genetic algorithm.

1. Introduction

In this paper, we propose a heuristic scheduling framework that integrates ant colony algorithm [1] and genetic algorithm [2]. Firstly, the warehouse space model is constructed by raster modeling method, and the shelves, workstations and obstacles in the complex environment are transformed into computable coordinate nodes. On this basis, a single-objective model is constructed with the objective of minimizing the picking time of the longest-path AGV [3], and an ant colony algorithm with dynamic parameter adjustment is used to solve the path planning [4]. It is further extended to the multi-objective optimization scenario [5] and establishes a comprehensive evaluation system including total path length, task load balance and maximum time consumption, and achieves task allocation optimization through the arrangement coding of genetic algorithm and adaptive mutation strategy. The experimental results show that this method significantly reduces the total path length and running time of the system while ensuring the efficient coordination of AGVs and provides a new solution for multi-objective dynamic scheduling. This study provides theoretical support for the intelligent upgrading of unmanned warehousing system and has important reference value for improving the level of logistics automation [6].

2. Model Preparation

The system assigns the task of moving the shelves to be picked to each mobile robot trolley, which can automatically travel along the nearest line by walking in the established space, steering, and then transporting the shelves to the operation platform, placing them in the activity area after reaching the target location, and then queuing up for manual operation. When the manual operation is completed, the robot cart will return to the initial area or continue to complete the next scheduling task.

(1) Data Preprocessing

The data in this paper is visualized as the following figure, the final warehouse information graph is obtained as Fig. 1.

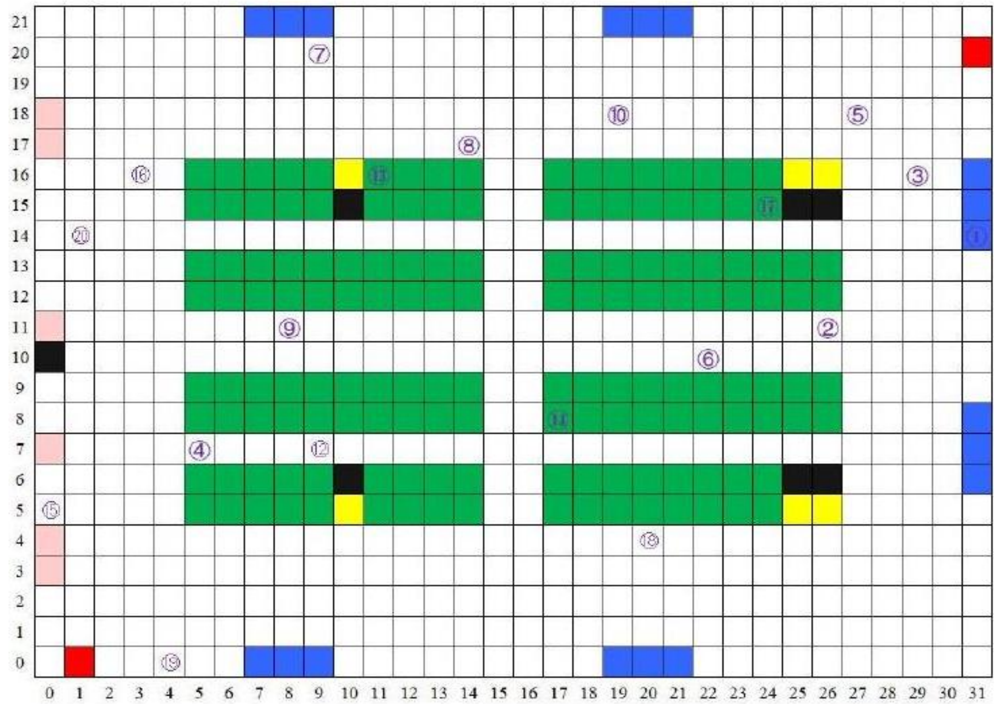

Figure 1. Preprocessed AGV information

(2) Define the elements in the map as follows

Set of all points: $A = \{0,1,2,\cdots,n\}$; Robot: $K = \{K_1, K_2, \cdots, K_n\}$; Path node: $Road = \{R_1, R_2, \cdots, R_{n_1}\}$; Reserve node: $N = \{1,2,3,\cdots,n\}$; Retain node: $Retain = \{R_1, R_2, \cdots, R_{n_3}\}$; Column node: $Column = \{C_1, C_2, \cdots, C_{n_4}\}$; Picking station node: $Pick = \{P_1, P_2, \cdots, P_{n_3}\}$; Replenishment node: $Complement = \{L_1, L_2, \cdots, L_{n_6}\}$; Empty Pallet Recovery node: $Empty = \{E_1, E_2\}$.

(3) Provision for AGV travel planning scenarios

The raster modeling method is chosen here to model the AGV picking system in the warehouse. For the different elements in the map, the route of the traveling area of the vehicle can be regarded as a two-dimensional plane, i.e., O. Establish a right-angle coordinate system for the plane area O, take the lower left corner as the coordinate origin 0, establish the X-axis horizontally, establish the Y-axis vertically, and divide the plane into a grid with the required width D of the AGV as the unit. The coordinates of the lower left corner are defined as $(0,0)$ and each grid has a defined coordinate (x, y) in the coordinate system. Place the warehouse into the coordinate system to get the location information of each AGV and node. In this case, the column nodes cannot be classified as ranges of passageways, and the other rasters are free rasters, which can be used for passage or goods placement.

3. Single Target Constraint

The AGV needs to visit all target shelves, and each target shelf is visited at least once. The distance traveled by the AGV between any two shelves is the shortest folded distance, known as the Manhattan distance. Continue to define the following information: (X_i, Y_i) denotes the position coordinates where shelf i is located; (AGV_X_k, AGV_Y_k) denotes the position coordinates where the AGV is located at k; S_i denotes the processing time required for shelf i to ship the goods; ${S_i}^g$ denotes the processing time required for shelf i to be moved from the storage node to the picking station. S_i^s denotes the time for shelf i to pick at the picking station, i.e. t_0; S_i^r denotes the time required to move shelf i from the picking station node back to the storage node by taking the shortest linear path;

P_i denotes the number of priorities for which shelf i has been picked, with smaller values indicating higher priorities; $T_{ij}{}^{k}$ denotes the shortest path required for machine k to move from shelf i to shelf j. V_k denotes the running speed of machine k. Among them, the decision variables are set as follows: $X_{ij}{}^{k}$ denotes the service relationship between machine k and shelf, if machine k delivers shelf i to shelf j after delivering shelf i, the marking takes the value of 1, otherwise it is 0; Z_i^k denotes the service relationship between machine k and shelf i, if shelf i is delivered by machine k, the marking takes the value of 1, otherwise it is 0; $U_{ij}{}^{k}$ denotes the flow variable related to the access of shelf by machine k, which is used to eliminate the sub-loop here; A_i denotes the time when shelf i starts to move; and F_k denotes the time when machine k completes all the transportation tasks and returns to the docking point.

In solving the AGV scheduling of unmanned warehouses, common solution scheduling performance indicators include the following: the shortest total distance of AVG operation, the shortest time of AGV transporting goods, the shortest time of AGV invalidly transporting goods, the largest utilization rate of AGV, the balanced distribution of AVG tasks, the smallest number of AGVs, and other cases. In order to ensure that the overall picking operation is completed quickly, this paper constructs a model to pick the AGV with the longest picking path, and minimizes the picking time as the goal, and constructs the objective function as follows:

$$\mathrm{Min}Z = \max\left[\sum_{i\in A} \sum_{j\in A, j\neq i} X_{ij}{}^{k} T_{ij}^{k} + \sum_{i\in N} z_i^k S_i\right] \tag{1}$$

4. Ant Colony Algorithm

In this section, the ant colony algorithm is used to solve the above single objective optimization model. When solving the path planning of AGVs, the ants are required to find the optimal path to reach the target point from the initial position. In this process, each ant represents each AGV, and the path taken by each ant is one of the feasible paths of the AGVs, and the optimal paths are retained, and the result is the optimal path required by the algorithm.

Through the above description, it can be seen that the ant colony needs to have a set of its own ideas in the selection of node paths according to the pheromone concentration, i.e., calculating the transfer probability of the surrounding free nodes and taking the larger of them as the next mobile node, which has a specialized transfer probability formula as follows:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}(t)]^{\beta}}{\sum_{j\in A_k} [\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}(t)]^{\beta}} & j \in A_k \\ 0 & \text{else} \end{cases} \tag{2}$$

Where, $p_{ij}^k(t)$ denotes the probability that there are k ants transferring from free node i to free node j at time t, $\tau_{ij}(t)$ denotes the pheromone concentration between free node i and free node j at time t, A_k denotes the set of free nodes around the kth ant, α denotes pheromone-inspired factor, and β denotes the expectation-inspired factor. The value range of both is usually in the interval (0,1), and $\eta_{ij}(t)$ denotes the heuristic function of ant k from free node i to free node j at time t. The specific expression is as follows:

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \tag{3}$$

The ants k keep on pathfinding according to the above probability formula, and after all the ants arrive at the specified location, all the visited road segments will undergo pheromone updating. The formula for pheromone update is shown below:

$$\begin{cases} \tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t,t+1) \\ \Delta\tau_{ij}(t,t+1) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t,t+1) \end{cases} \tag{4}$$

Where ρ is the pheromone volatilization coefficient, in order to prevent infinite increase of pheromone concentration, it is stipulated here that $\rho \in (0,1)$, m is the number of ants, $\Delta\tau_{ij}$ denotes the amount of pheromone concentration left behind by the ants arriving at the free node j from the free node i, and $\Delta\tau_{ij}^{k}$ denotes the amount of change of pheromone concentration of ant k from the free node.

$$\Delta\tau_{i}^{jk} = \begin{cases} \frac{Q}{L_k} & \text{Ant k reaches node j from node i} \\ 0 & \text{else} \end{cases} \tag{5}$$

Where Q is the pheromone intensity, which represents the concentration of information released by the ants during the pathfinding process, and L_k represents the length of the path taken by the ants during the pathfinding.

According to the above analysis, it can be learned that the pheromone heuristic factor α, the expected value heuristic factor β and the pheromone intensity coefficient Q are all very important parameters, which usually take the value of a fixed constant. However, in practice, at different stages, the parameters of the algorithm are required to be different, i.e., the parameters need to be adjusted in a dynamic way.

4.1. Dynamic Adjustment of Parameters

When using the ACO algorithm for path planning, the parameter α needs to be increased and then decreased, but the opposite is true for β. Therefore, the dynamic parameter adaptive heuristic factors are organized as follows:

$$\alpha(Nc) = \begin{cases} A + \sqrt{\frac{Nc}{Nc_{\max}}} \times C & Nc \le \frac{Nc_{\max}}{2} \\ B - \sqrt{\frac{Nc}{Nc_{\max}}} \times C & Nc > \frac{Nc_{\max}}{2} \end{cases}$$
$$\beta(Nc) = \begin{cases} D - \sqrt{\frac{Nc}{Nc_{\max}}} \times F & Nc \le \frac{Nc_{\max}}{2} \\ E - \sqrt{\frac{Nc}{Nc_{\max}}} \times F & Nc > \frac{Nc_{\max}}{2} \end{cases} \tag{6}$$

Where A, B, C, D, E, F is the tuning parameter and all are positive; Nc is the current iteration number and $Nc_{\max}$ is the maximum iteration number.

The intensity Q of this pheromone is adaptively processed with the following formula:

$$Q(Nc+1) = Q(Nc) + \lambda \times \frac{(L_B - L_b)}{L_B} \tag{7}$$

Where, λ is the adjustment parameter, L_B denotes the length of the optimal path among all previous paths, and L_b denotes the length of the optimal path found in this generation.

4.2. Improvement of Transfer Method

The study adopts the guideline function of the path with the following expression:

$$f = \frac{1}{d_{jG}} \tag{8}$$

Where d_{jG} is the distance between the next optional node j and the goal point G.

After the above optimization process, the new heuristic function is obtained as follows:

$$\eta' = \frac{1}{\omega_j \cdot (\varphi \cdot d_{ij} + (1-\varphi) \cdot d_{jG})} \tag{9}$$

Where ω_j is the superiority factor of the movable node.

$$\omega_j = \begin{cases} 1 & The\ number\ of\ free\ nodes\ around\ j\ is\ less\ than\ 5 \\ 0.5 & else \end{cases} \tag{10}$$

5. Exploring the Optimal Strategies for Integrated AGV Scheduling

5.1. Establishment of Multi-Objective AGV Scheduling Model

According to the above analysis, it is known that the total distance moved by the AGV in unloaded state is an important condition for task assignment, for R_i, its task is $H_i = [H_i{}^0, H_i{}^1, \cdots, H_i{}^n](n < N).R_i$ departs from the starting docking point, $H_i{}^0$, and then returns to the original docking point. , and then return to the original docking point, the total distance moved in the task chain is given by Eq:

$$Di = \sum_{j=0}^{n} \sum_{k=0}^{n} c_i^{jk} x_i^{jk}\ i = 1,2,\cdots,M; j \neq k$$
$$x_i^{jk} = \begin{cases} 1 \\ 0 \end{cases} \tag{11}$$

Where D_i denotes the total distance moved by R_i in H_i and c_i^{jk} denotes the distance cost of R_i from H_i^j to H_i^k. The value of $x_j{}^{jk}$ is taken as 1 to indicate that R_i moves from task node j to task node k, otherwise its value is taken as 0. After all, N handling tasks are completed, the total distance of each AGV in the task chain is:

$$D = \sum_{i=1}^{M} D_i \tag{12}$$

The synergy between AGVs is a prerequisite for the efficient operation of a multi-AGV system. In order to avoid the phenomenon of uneven distribution, the task load balance degree of AGVs is used as an optimization index as follows:

$$\mu = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(\sum_{j=1}^{N} y_i^j - \frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N} y_i^j\right)^2}$$
$$y_i^j = \begin{cases} 1 \\ 0 \end{cases} \tag{13}$$

Where μ denotes the load balancing degree of AGVs, the smaller its value, the more similar the number of tasks assigned by AGVs. After all the AGVs have completed their respective tasks, the length of time that the AGVs work in the task chain is calculated, and the maximum working time is taken as the standard of time consumption, and the multi-objective model is constructed as follows:

$$T_i = \frac{D_i}{v_i} + \sum_{j=1}^{N} y_i^j \times \delta \; i = 1,2,\cdots,M$$
$$T = \max[T_1, T_2, \cdots, T_M] \tag{14}$$

Due to the multi-AGV task assignment model established above; in order to facilitate the calculation, it is necessary to transform the multi-objective model into a single-objective problem for solving first. The evaluation function for multi-AGV task assignment is established as:

$$F = q_1 \times D + q_2 \times \mu + q_3 \times T \tag{15}$$

Where each coefficient takes a positive value.

5.2. Genetic Algorithm

After a series of comparative analysis, this paper decides to use genetic algorithm to solve the multi-AGV task allocation problem.

(1) Alignment coding

There are $\mathrm{M+N}$ traversal nodes in the model constructed in this section. Since each AGV starts from the initial stopping point and the positional order of its visit has already been determined, the initial positions of M AGVs are not encoded, and the remaining N task nodes are encoded in two segments, i.e., the chromosome is divided into two segments for processing.

(2) Initialized population

Randomly generate the initialized population for calculation, when the system receives the message, according to the initial position of the AGV, each handling task is randomly combined and arranged, and then get the initial solution set of each AGV.

(3) Adaptation evaluation function

The designed adaptation evaluation function is the inverse of the objective function, i.e:

$$f = \frac{1}{q_1 \times D + q_2 \times \mu + q_3 \times T} \tag{16}$$

(4) Selection of the operator

The algorithm used is the most commonly used roulette method, that is, for the selection of individuals based on their fitness values, retaining the individual with the larger fitness value. The selection probability formula is as follows:

$$p_i = \frac{f_i}{\sum_{j=1}^{z} f_i} \tag{17}$$

Where p_i denotes the probability that the ith individual is selected, f_i denotes the size of fitness, and z denotes the size of the population.

(5) Chromosome crossover

Matching crossover method is used in this section.

(6) Chromosome mutation

Reducing the mutation rate, the improved method of calculating the mutation rate is shown below:

$$p_m(k) = \frac{1}{1+\xi \cdot e^{\frac{k}{K}}} \tag{18}$$

Where ξ is the mutation factor, k is the current iteration number, and K is the maximum iteration number. Reversal mutation is performed for the first segment, and basic position mutation is performed for the second segment.

(7) End of iteration

Here the threshold is set, if the number of iterations does not reach the iteration threshold, the algorithm continues, if the maximum preset value has been reached, the iteration stops and outputs the solution.

5.3. Multi-Objective Result Analysis

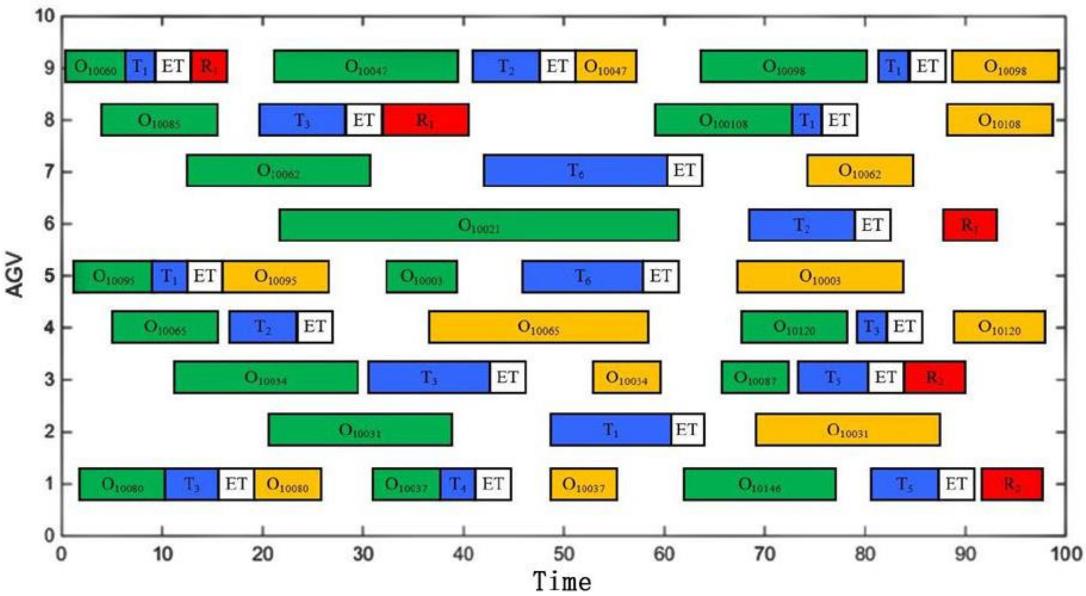

Figure 2. Scheduling Gantt chart

In Fig. 2 above, the scheduling Gantt chart for ten orders is given. Among them, the green square indicates the time taken by the robot to fetch the corresponding pallet on the shelf, which is named with the letter O + pallet ID; the blue square indicates the time taken by the robot to deliver the goods to the picking stations, and the six picking stations are represented as $T_1, T_2, T_3, T_4, T_5, T_6$; ET denotes the fixed dwell time for arriving at the workstation and waiting for the picking robot to pack and ship the goods t_0; the yellow square denotes the time used by the robot to carry out the task of returning the remaining goods on the pallet to the warehouse; the red square denotes the time used by the robot to carry out the task of retrieving the empty pallet; the length of the square denotes the length of the time used to complete this task. time; the length of the square represents the length of time used to complete this task, and the remaining positions indicate that the robot made a stop without a task during this period.

In the end, using genetic algorithm to solve the multi-objective planning, the total path to complete 675 orders are 18305 and the time taken is 22162 seconds. By comparing the multi-objective constraint optimization with the single-objective constraint, it is clear that the multi-objective constraints are able to achieve the minimization of the total path traveled by all the handling robots while each of them is as busy as possible.

The model comparison results are shown in table 1.

Table 1. Model Comparison Results

	Total path	Time taken
Single-objective constraint	21864	23765
Multi-objective constraints	18305	22162

6. Conclusion

In this paper, a heuristic optimization framework integrating ant colony algorithm and genetic algorithm is proposed for AGV multi-objective dynamic scheduling in unmanned warehouse system. By constructing a rasterized warehouse model, the AGV path planning is transformed into a node traversal situation based on the Manhattan distance, a single-objective model is established with the objective of minimizing the picking time of the longest path, and the parameters of the ant colony algorithm are dynamically adjusted to improve the path search efficiency. Further, a multi-objective optimization model is constructed to consider the total driving distance, task balance and maximum time consumption, and a two-stage coding strategy of genetic algorithm is adopted to achieve task allocation optimization. The study breaks through the limitation of traditional single-objective scheduling through multi-objective cooperative optimization, and significantly improves the system operation efficiency while ensuring the load balance of AGVs. In practical application, the model can dynamically adjust the parameters according to the warehouse layout to adapt to different order sizes and operation scenarios, providing a feasible solution for the intelligent upgrade of unmanned warehouses. This study provides a new theoretical perspective for AGV scheduling and has important reference value for the development of logistics automation technology.

References

[1] Deng Dongdong, Xu Jianmin, Meng Han, et al. Mobile robot path planning based on fusion of ant colony algorithm and artificial potential field method[J/OL]. Journal of Instrumentation, 1 - 16 [2025 - 03 - 15]. https://doi.org/10.19650/j.cnki.cjsi.J2413095.

[2] CAO Mukun, WANG Weijie, DAVID. Research on logistics distribution center layout model based on SLP and genetic algorithm[J/OL]. Industrial Engineering and Management, 1 - 22 [2025 - 03 - 15]. http://kns.cnki.net/kcms/detail/31.1738.t.20250227.1109.002.html.

[3] SUN Zhuo, QI Ziyang. Collision avoidance path planning for AGVs in automated warehouses considering vehicle and task matching correlation [J/OL]. Computer Application Research, 1 - 11 [2025 - 03 - 15]. https://doi.org/10.19734/j.issn.1001-3695.2024.10.0359.

[4] LIU Jianhui, WANG Qiong. Optimization method of emergency material distribution vehicle scheduling for multiple distribution centers based on multi-objective ant colony algorithm [J/OL]. Journal of Jilin University (Engineering Edition), 1 - 7 [2025 - 03 - 15]. https://doi.org/10.13229/j.cnki.jdxbgxb.20231452.

[5] An Yuanyuan, Ma Xiaoning. Improved Genetic Algorithm and Multi-objective Optimization Model for Flight Path Planning [J]. Computer Engineering and Science, 2024, 46 (09): 1660 - 1666.

[6] YE Xuqian, WANG Dan, LIU Yiyang, et al. Unmanned warehouse scheduling optimization based on digital twin and deep reinforcement learning [J]. Logistics Technology, 2023, 42 (09): 97 - 100+121.