Optimization Research and Application of Stereoscopic Warehouse Scheduling Strategies

Yaofa Li, Lijie Ma

***Abstract*— The problem of large cargo volume and low I/O scheduling efficiency is common in Stereoscopic warehouses. In order to improve the efficiency of warehouse access, on the basis of adopting the two-end single-aisle double stacker model, and taking the running time of the stacker as the evaluation criterion, we establish a double stacker model in which the single-command task and the double-command task coexist and reasonably merge the inbound task and the outbound task into a single process, so as to reduce the number of times that the stacker returns to the inbound platform and the outbound platform. The number of times the stacker returns to the inbound platform and the outbound platform is reduced. In order to solve the problem of scheduling the execution order of tasks, the ocean predator algorithm is used to optimize the solution. In order to evaluate the performance of the model and algorithm, simulation experiments are conducted based on a batch of inbound and outbound tasks, and the results show that this research can reduce the idle running time of the stacker cranes and improve the efficiency of the inbound and outbound scheduling of the warehouse.**

***Index Terms*—I/O scheduling, MPA, stacker cranes, stereoscopic warehouse.**

I. INTRODUCTION

Stereoscopic warehouse as a new warehousing mode is more and more applied to various production environments, with the continuous growth of the warehouse, in and out of the warehouse scheduling has become one of the key factors affecting the efficiency of the entire warehouse operation. Stacker as the main equipment for transporting and accessing goods, its working time accounts for two-thirds of the entire in and out of the warehouse scheduling operation time, so the reasonable optimization of the driving path of the stacker to reduce the stacker no-load running time, is to improve the overall efficiency of the intelligent stereoscopic warehouse is an effective means.

The scholars have conducted a lot of research on the optimization of the I/O scheduling of intelligent stereoscopic warehouse. In order to improve the efficiency of the automated storage and scheduling system. Zou et al. [1] applied expert systems to automated storage and retrieval systems in order to improve the efficiency of automated storage and scheduling systems. A knowledge base based on production rules is established. More reasonable storage allocation and faster job scheduling are achieved. In order to improve the efficiency of cargo scheduling. Liu et al. [2] used order labels to record all goods belonging to the same sequence and perform dynamic scheduling to ensure the integrity of the order. To solve the command block scheduling problem in automated storage and retrieval systems, Hsu et al. [3] proposed a framework that combines meta-heuristics, simulation, and optimization. This framework utilizes a hybrid approach of the whale optimization algorithm and particle swarm optimization algorithm to minimize the total running time. The above research mainly aimed at the total completion time and optimize the scheduling sequence. It reduces the storage and retrieval time of goods, and improves the efficiency of job scheduling.

Zhou et al. [4] aimed at the problems of long waiting time of customers and long running distance of stackers in automated stereoscopic warehouses in laneway. A beam search algorithm based on improved genetic algorithm is proposed to optimize the stacker operation path. Significantly reduced the average customer waiting time. Sun B et al. [5] use vectors to represent orders, assign goods to stackers according to vectors, and transform them into a complete scheduling solution of "First Come First Serve" (FCFS). Liu et al. [6] aimed to minimize the completion time of storage and retrieval tasks, with a prioritization given to retrieval tasks. A Pareto-based genetic algorithm was employed to solve the problem, resulting in an improved scheduling efficiency. The above research mainly focuses on a certain operation mode of the stacker, without comprehensive consideration of the scheduling combination. For example, the appropriate combination of the I/O tasks will reduce the number of times the stacker returns to the I/O workstations, and shorten the no-load operation time of the stacker.

Ma et al. [7] proposed a multi-objective biogeographer algorithm based on the characteristics of AS/RS I/O scheduling to optimize the operation model of the stacker. Geng et al. [8] proposed a two-end double stacker model with improved anti-collision strategy. Effectively solve the large automatic warehouse double stacker I/O scheduling problem. In the above research, it is assumed that the speed of the stacker is constant in the horizontal and vertical directions. In fact, the stacker moves in the horizontal and vertical directions with uniform acceleration and deceleration. The real operation situation is helpful to the correctness of the model solution.

According to the above research on the access scheduling of intelligent stereoscopic warehouses, although good progress has been made, there are still places that need to be improved, such as large warehouses only for the access scheduling combination of efforts and did not consider the

Manuscript received March 17, 2024

Yaofa Li, School of Computer Science and Technology, Tiangong University, Tianjin, China
Lijie Ma, School of Computer Science and Technology, Tiangong University, Tianjin, China

optimal division of the two stacker cranes operating area will also be effective in shortening the no-load running time of the stacker cranes; most of the research assumed that the speed of the stacker cranes in the horizontal and vertical direction is constant, but the actual cranes in the horizontal and vertical direction do uniform acceleration and deceleration motion. In this paper, based on the model of two-end type double stacker cranes in a stereoscopic warehouse, and taking the running time of the stacker cranes as the objective, the improved marine predator algorithm is applied to optimize the solution to minimize the no-load running time of the stacker cranes and improve the efficiency of the warehouse in/out scheduling.

II. DOUBLE STACKER SCHEDULING MODEL

A. *Description of the Double Stacker Mode*

The single stacker cranes layout model is often used in small and micro-enterprises, which are characterized by fewer inbound and outbound tasks, smaller warehouse capacity, etc. The single stacker cranes basically satisfy the scheduling needs of a small number of outbound/inbound tasks; in the production and living environment of medium and large enterprises, the outbound/inbound scheduling tasks of stereoscopic warehouses are suddenly increased, and the stacker cranes will perform a large number of inbound and outbound scheduling tasks, and it will take a large amount of time to deal with the picking tasks only by the single stacker cranes, which makes the productivity unsatisfactory. In order to optimize the warehouse scheduling efficiency, shorten the time of outbound/inbound scheduling tasks, and improve the efficiency of accessing goods, the dual stacker cranes model has become the preferred layout for medium and large enterprises.

The double stacker cranes layout of the stereoscopic warehouse sets out/in dock at both ends of the shelves, with the out dock and in dock located on both sides of the aisle, and the external transmission system connecting the left and right out/in dock; the aisle is equipped with two stacker cranes, each of which picks and sends to the out/in dock for its own work area, which reasonably divides the work area of the stacker cranes and greatly shortens the in/out dock scheduling time, allowing the two stacker cranes to pick and send to the out/in dock. This rationally divides the work area of the stacker cranes and greatly shortens the dispatching time of the in/out tasks, enabling both stacker cranes to perform the out/in tasks in the most efficient way, thus improving the production efficiency of the factory.

The bottleneck of the warehouse access scheduling system is how to reasonably combine the access tasks into one task process, reduce the number of times the stacker cranes return to the access table, and shorten the idle running time of the stacker cranes, so as to improve the overall efficiency of the warehouse access system. When the system receives a batch of inbound and outbound tasks, there are two main ways for the stacker cranes to perform access [9]. If the system now receives an inbound and outbound task.

1) Execute the inbound and outbound tasks in the order of the order, the stacker cranes execute the inbound instruction first, and return to the I/O station empty after completing the inbound task from the inbound station, and then execute the outbound instruction, and arrive at the location of the goods to be outbound empty to complete the outbound task. The stacker can only execute one instruction each time, which is called Single instruction cycle job (SC).

2) The stacker cranes combine the inbound and outbound tasks, picking goods from the inbound station and delivering them to the designated warehouse position, then executing the outbound task instructions immediately afterward, and arriving at the outbound position to pick goods with full load and returning to the outbound station. The stacker executes two tasks at a time, which is called Double instruction cycle job (DC).

Based on this, this paper aims to reduce the number of times the stacker cranes return to the inbound and outbound dock, combine the inbound and outbound tasks in two by two, and establish a model with the goal of shortening the idle running time of the stacker cranes, so as to enable the stacker cranes to complete the inbound and outbound tasks more efficiently, and improve the efficiency of the warehouse inbound and outbound scheduling.

B. *Relevant Assumption Parameters*

In order to make the research more practical, the parameters of the double stacker layout are set as follows:

1) The shelves on both sides of the roadway are fixed with V column and H layer, with a total of $V \times H$ warehouse locations. Let the coordinates of the i layer and j column be [i, j]. The coordinates of the left-hand area stacker I/O workstation IO_{left} is [0,0], and the coordinates of the right-hand area stacker I/O workstation IO_{right} is [$V+1$, 0].The two stackers have the same model and parameters, and both are capable of performing I/O tasks.

2) Each stacker can only load one item at one time, and the storage locations on both sides of the aisle have the same size and can only hold one item.

3) The maximum horizontal speed of the stacker when no-load is $V_{\max_x}$, and the maximum vertical speed is $V_{\max_y}$. The acceleration of the stacker is a. The startup and braking time of the stacker are ignored.

Based on the given parameters, the time and distance required for the stacker to reach its maximum horizontal speed when no-load can be determined: $T_{\max_x} = \frac{V_{\max_x}}{a}$, $S_{\max_x} = \frac{V_{\max_x}^2}{2a}$.The time and distance when the vertical velocity reaches its maximum: $T_{\max_y} = \frac{V_{\max_y}}{a}$, $S_{\max_y} = \frac{V_{\max_y}^2}{2a}$.

C. *Model Building*

Regarding the aforementioned issues, taking the left stacker as an example, assuming there are n storage points and m retrieval points.

$$\begin{cases} dc = \min(n,m) \\ task = \max(n,m) \\ sc = task - dc \end{cases} \quad (1)$$

Among them, the stacker executes dc double instruction cycle job tasks, and sc single instruction cycle job tasks. The no-load operation of the stacker is as follows:

1) When the stacker performs SC tasks, it either returns no-load after completing the storage operation or arrives empty at the retrieval point from the I/O workstation.

$$T_x = \begin{cases} 2\times T_{\max_x} + \dfrac{S_x - 2\times S_{\max_x}}{V_{\max_x}} & S_x > 2\times S_{\max_x} \\ 2\times\sqrt{\dfrac{S_x}{a}} & S_x \le 2\times S_{\max_x} \end{cases} \quad (2)$$

$$t_{(x_a, x_b)} = T_x + T_y \quad (3)$$

In formula (2), T_x is the time when the stacker moves in the horizontal direction. $T_{\max_x}$ and $S_{\max_x}$ are the time and distance when the horizontal velocity of the stacker reaches the maximum under the condition of no-load. S_x and S_y refer to the distance between the horizontal and vertical coordinate points. $V_{\max_x}$ and $V_{\max_y}$ are the maximum speed of the stacker in the horizontal and vertical directions. And a is the acceleration. $S_x > 2\times S_{\max_x}$ represents that the stacker can reach a uniform speed motion state in the horizontal direction. And $S_x \le 2\times S_{\max_x}$ represents that the stacker can not reach a uniform speed state in the horizontal direction, that is, the stacker first uniformly accelerates and then uniformly decreases; Same thing for T_y. $t_{(x_a,\ x_b)}$ represents the time it takes for the stacker to move from point a to point b.

$$T_{SC_i} = \begin{cases} t_{(x_{pos},\ x_{IO_{left}})} & if\ \ stacker \ne IO_{left} \\ t_{(x_{IO_{left}},\ x_{pos})} & if\ \ stacker = IO_{left} \end{cases} \quad (4)$$

Where T_{SC_i} represents the time taken by the stacker to perform the i-th SC task. x_{pos} represents the coordinate point of the storage and retrieval goods. $x_{IO_{left}}$ represents the position of the I/O workstation. $t_{(x_{pos},\ x_{IO_{left}})}$ represents the time taken by the stacker to return empty from the storage location to the I/O workstations. At this time, the stacker is at the storage point. $t_{(x_{IO_{left}},\ x_{pos})}$ represents the time taken by the stacker to depart no-load from the I/O workstations and reach the retrieval point.

2) The stacker performs DC tasks, during which it proceeds to the retrieval point after completing the storage task and then empties to execute the retrieval task.

$$T_{DC_j} = t_{(x_I,\ x_O)} \quad (5)$$

In Eq. (5), T_{DC_j} means that the stacker executes the j-th DC task. Where $t_{(x_I,\ x_O)}$ represents the time taken for the stacker to travel no-load from the storage point x_I to the retrieval point x_O after completing the storage task. The specific calculation method is the same as in Eq. (3).

3) When two stackers are running at the same time, the final task completion time is the maximum time of the two stackers.

$$T_{left} = \sum_{i=1}^{sc} T_{SC_i} + \sum_{j=1}^{dc} T_{DC_j} \quad (6)$$

In Formula (6), T_{left} represents the time spent by the left stacker in the operating area, including executing sc SC tasks and dc DC tasks. The time spent by the right stacker can be calculated as T_{right} similarly, so the total running time of the double stacker is:

$$T = \max(T_{left}, T_{right}) \quad (7)$$

III. OPTIMIZATION OF I/O SCHEDULING PATH

Metaheuristic algorithm is usually used to solve a batch of I/O tasks. Hammouri et al. [10] expressed the problem of a salesman visiting multiple cities as a traveling salesman problem. Access to multiple cities use dragonfly algorithm of the shortest path and obtained a good solution. Muthukumaran et al. [11] proposed and implemented a Dragonfly-Cuckoo hybrid search algorithm for solving the collision-free path problem in agricultural landscapes. This algorithm was used to generate optimal sequential paths for spray applications in greenhouse environments. Zan et al. [12] introduced the whale optimization algorithm combined with computer perception technology to realize the path planning of the robot. Han et al.[13] the whale optimization algorithm to help ships find low energy consumption in large-scale complicated Marine environment and safe routes. It solves the problem that it is difficult to plan a reasonable route for ships affected by complex Marine meteorological environment. Although dragonfly algorithm and whale algorithm have shown superiority in the face of TSP, their performance has decreased in the face of relatively complex path planning problems. In the local search stage of the algorithm, it is easy to fall into local optimum and difficult to jump out. In view of these shortcomings, based on the model established in this paper, Marine Predators Algorithm (MPA) which is easy to jump out of local optimum and has strong search ability is adopted [14].

A. Marine Predators Algorithm

The algorithm is divided into three stages according to the number of iterations. Each stage computes the step size and moves the prey position using a different stochastic strategy. If the position is in the position of the predator prey, predators will move to the location of prey. The MPA has only two main core concepts: predator and prey. In each stage, only the prey will randomly move. The predator moves to the superior prey after the prey is finished. That is to say their prey is the current position, the predator is the optimal location history.

1) Stage 1: This stage occurs in the first one-third of the total number of iterations and is referred to as the exploration stage. At this time, the predator moves faster than the prey, and mainly performs global search and captures the prey. The overall movement behavior is characterized as Brownian

motion.

$$\begin{cases} step_i = RB \otimes (Elite_i - RB \otimes Prey_i) \\ Prey_i = Prey_i + P \times R \otimes step_i \end{cases} \quad iter < \frac{Max_{iter}}{3} \tag{8}$$

In formula (8), $step_i$ is the moving step length. RB is a random vector based on normal distribution of Brownian motion. R is a random vector uniformly distributed in [0,1]. P is a constant and set to 0.5. n is the number of populations. $iter$ is the current iteration number, and Max_{iter} is the maximum iteration number.

2) Stage 2: This stage occurs between one-third and two-thirds of the total number of iterations and is referred to as the exploration-exploitation transition stage. During this stage, the predators and prey have similar speeds. The prey primarily focuses on exploitation and uses a Levy strategy for random movement. The predators, on the other hand, continue to explore using Brownian motion. Both exploration and exploitation are crucial during this stage.

$$\begin{cases} step_i = RL \otimes (Elite_i - RL \otimes Prey_i) & i \le \frac{n}{2} \\ Prey_i = Prey_i + P \times R \otimes step_i & \\ step_i = RB \otimes (Elite_i - RB \otimes Prey_i) & i > \frac{n}{2} \\ Prey_i = Prey_i + P \times CF \otimes step_i & \end{cases} \tag{9}$$

$$\frac{Max_{iter}}{3} < iter < \frac{2 \times Max_{iter}}{3}$$

$$CF = \left(1 - \frac{iter}{Max_{iter}}\right)^{2\frac{iter}{Max_{iter}}} \tag{10}$$

In Eq. (9), RL represents a random vector based on the *Levy* distribution. Multiplying RL with $Prey_i$ simulates the *Levy* motion of the prey. RB multiplied by $Elite_i$ simulates the Brownian motion of the predator. CF denotes the adaptive parameter.

3) Stage 3: This stage occurs in the last one-third of the total number of iterations and is the final stage of the algorithm. At this stage, the speed of the prey is faster than that of the predator. The predator adopts a *Levy* walk strategy for fast movement.

$$\begin{cases} step_i = RL \otimes (RL \otimes Elite_i - Prey_i) \\ Prey_i = Elite_i + P \times CF \otimes step_i \end{cases} \quad iter > \frac{2 \times Max_{iter}}{3} \tag{11}$$

The multiplication of RL and $Elite_i$ in Eq. (11) simulates the *Levy* walking motion of the predator.

4) Eddy formation and FADs effect: The efficiency of the proposed algorithm is largely determined by the predatory ability of the predator. And the Marine environment has a great impact on the ability to hunt. Eddies and fish aggregation devices (FADs), for example, can alter the behavior of predators.

$$\begin{cases} Prey_i + CF[X_{\min} + r \otimes (X_{\max} - X_{\min})] \otimes U \;\; if \;\; r \le FADs \\ Prey_i + [FADs(1-r) + r](Prey_{r1} - Prey_{r2}) \;\; if \;\; r > FADs \end{cases} \tag{12}$$

In Eq. (12), $FADs$ is the probability of influencing the optimization process and is set to 0.2. That is, the predator will roam around the prey in 80% of the search space, and the remaining 20% will roam in different areas to improve the global search. To enhance the convergence ability of the algorithm. U is a binary vector array containing 0 and 1. If the random solution is less than *FADs*, *U* is set to 0. If the random solution is greater than FADs, U is set to 1. *r1* and *r2* are random indices of the prey matrix. When $r > FADs$, the predator will make longer jumps in different dimensions. When $r \le FADs$, the predator will randomly move within the current foraging space.

IV. Experimental Simulation and Analysis

A. Setting Model Parameters

In order to validate the applicability of the improved double stacker cranes model proposed in this paper and the effectiveness of MPA. In this paper, a real warehouse of a company's goods storage and retrieval operation is used as an example for validation. The stereoscopic warehouse adopts a single aisle double rack layout, and the position of each cargo space on the rack is fixed and identical. There are I/O workstations and stacker cranes of the same type and size at both ends of the aisle. The stacker cranes perform storage and retrieval operations in their respective work areas. The specific parameters are shown Table 1.

Table 1: Simulation parameter information table of stereoscopic warehouse

Name	Parameters
Acceleration a	0.5 m/s^2
Maximum x-axis speed V	3 m/s
Length of cargo position *l*	2 m
Height of cargo position *h*	1 m
Number of shelf layers *H*	15 *floors*
Number of shelf rows *V*	70 *columns*

Table 2: Coordinates of storage and retrieval tasks

Storage tasks		Retrieval tasks	
Num	Coordinate point	Num	Coordinate point
1	(53,6)	21	(56,9)
2	(45,8)	22	(39,3)
3	(15,11)	23	(20,14)
4	(65,13)	24	(14,13)
5	(25,15)	25	(24,6)
6	(45,10)	26	(60,1)
7	(52,12)	27	(34,12)
8	(43,5)	28	(33,2)
9	(43,1)	29	(5,12)
10	(50,14)	30	(31,14)
11	(65,10)	31	(52,4)
12	(60,15)	32	(50,2)
13	(10,15)	33	(8,6)
14	(32,10)	34	(27,5)
15	(38,9)	35	(42,15)
16	(33,7)	36	(26,11)
17	(26,13)		
18	(15,8)		
19	(36,5)		
20	(25,3)		

B. Experimental Comparison

To demonstrate the effectiveness of the proposed improvement to the MPA in this paper. Based on the simulation parameters in Table 1 and the warehouse I/O tasks in Table 2, validation is conducted. DA, WOA, MPA and improved MPA are used to optimize the above I/O tasks. The initial parameters of MPA are set, the initial population size is $n = 50$, the maximum number of iterations is $Max_{iter} = 300$,The experimental simulation is performed on MatLabR2021b.The initialization parameters of the algorithm are shown in Table 3.

Table 3: Initialization parameter Settings

Algorithm	Main parameter Settings
DA	Separation weight $t = 0.1$, Alignment weight $a = 0.1$, Cohesion weight $c = 0.7$
WOA	Spiral shape coefficient $b = 1$, Position update threshold $p = 0.5$.
MPA	$FADs = 0.2$, $P = 0.5$

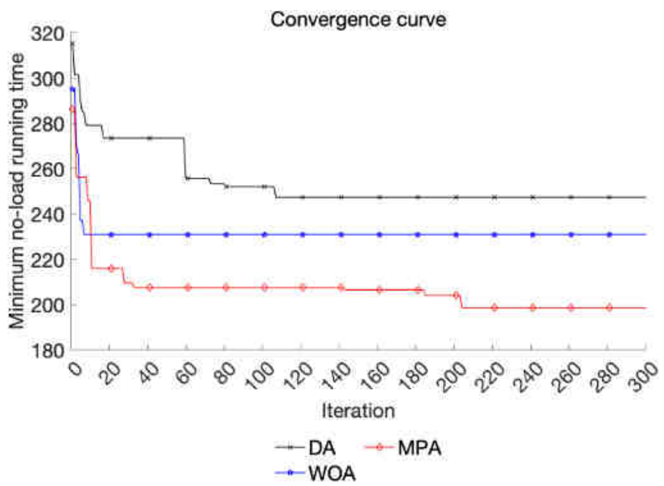

Figure 1 Convergence curve of the algorithm

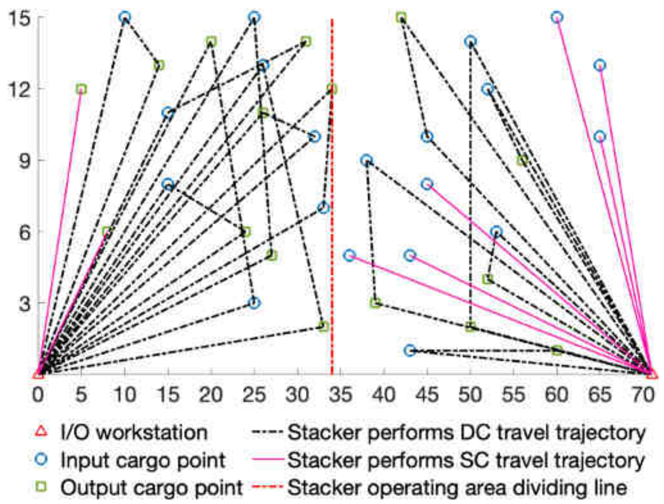

Figure 2 Trajectory of the stacker

Table 4: Comparison of stacker operation effect

Evaluation metrics	DA	WOA	MPA
iter=100	252.201s	231.037s	207.576s
iter=200	247.479s	231.037s	204.206s
iter=300	247.479s	231.037s	198.667s

As can be seen from Fig. 1, the unloaded running time of the stacker cranes decreases with the increase of the iteration number, although the convergence effect of DA and WOA is stronger than that of MPA, the quality of the MPA solution is better, and as can be seen from Table 4, DA and WOA are easy to fall into the local optimum during iteration, and the quality of the resulting solutions varies, MPA uses a lot of measures to jump out of the local optimum in the second and the third stages of iteration to avoid falling into the local optimum, and the results of jumping out of local optimum are significantly better than that of DA and WOA. In order to avoid falling into local optimality, MPA adopts many measures to jump out of local optimality in the second and third stages of iteration, and it can be seen from the images that the effect of jumping out of local optimality is significantly better than that of DA and WOA, and the unloaded running time of the stacker cranes is reduced by 48.812 s and 32.37 s, respectively.

V. Conclusions

Aiming at the phenomenon that large-scale stereoscopic warehouses have many inbound and outbound goods and low scheduling efficiency of stacker cranes, a dual stacker cranes model has been established by taking the no-load time of two stacker cranes as the evaluation criterion. The stacker cranes of this model perform acceleration and deceleration motions in the horizontal and vertical directions, which solves the defect of assuming uniform speed motion of stacker cranes in the horizontal and vertical directions. On the basis of the model, MPA is optimized to solve the problem. Comparative experiments show that the algorithm has good optimization searching effect and high solution quality, which helps to solve the in/out scheduling problem.

References

[1] Zou S, Hu M. Optimization for the Storage Management and Job Scheduling Based on Expert System[C]//8th International Conference on Management and Computer Science (ICMCS 2018). Atlantis Press, 2018: 15-19.
[2] Liu J, Liew S Y, Ooi B Y, et al. Dynamic order-based scheduling algorithms for automated retrieval system in smart warehouses[J]. IEEE Access, 2021, 9: 158340-158352.
[3] Hsu H P, Wang C N. Hybridizing whale optimization algorithm with particle swarm optimization for scheduling a dual-command storage/retrieval machine[J]. IEEE Access, 2023, 11: 21264-21282.
[4] Zhou L, Dang J, Zhang H, et al. Path optimization of stereoscopic garage stacker based on IGA-beam search[C]//2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC). IEEE, 2019: 119-124.
[5] Sun B, Zhang X, Qiao H, et al. Multi-type resources collaborative scheduling in automated warehouse with fuzzy processing time[J]. Journal of intelligent & fuzzy systems, 2020, 39(1): 899-910.
[6] Liu S. Research on scheduling policy of automated warehouse system[C]//Proceedings of the 2nd International Conference on Computer Science and Application Engineering. 2018: 1-5.
[7] Ma H, Su S, Simon D, et al. Ensemble multi-objective biogeography-based optimization with application to automated warehouse scheduling[J]. Engineering Applications of Artificial Intelligence, 2015, 44: 79-90.
[8] Geng S, Wang L, Li D, et al. Research on scheduling strategy for automated storage and retrieval system[J]. CAAI Transactions on Intelligence Technology, 2022, 7(3): 522-536.
[9] CAI A, XUE X, GUO S, et al. Stacker Scheduling of Layouts for Automatic Warehouses with Double Ended[J]. China Mechanical Engineering, 2019, 30(06): 735.

[10] Hammouri A I, Samra E T A, Al-Betar M A, et al. A dragonfly algorithm for solving traveling salesman problem[C]//2018 8th IEEE international conference on control system, computing and engineering (ICCSCE). IEEE, 2018: 136-141.
[11] Muthukumaran S, Ganesan M, Dhanasekar J, et al. Path planning optimization for agricultural spraying robots using hybrid
[12] Zan J. Research on robot path perception and optimization technology based on whale optimization algorithm[J]. Journal of Computational and Cognitive Engineering, 2022, 1(4): 201-208.
[13] Han Q, Yang X, Song H, et al. Whale optimization algorithm for ship path optimization in large-scale complex marine environment[J]. IEEE Access, 2020, 8: 57168-57179.
[14] Faramarzi A, Heidarinejad M, Mirjalili S, et al. Marine Predators Algorithm: A nature-inspired metaheuristic[J]. Expert systems with applications, 2020, 152: 113377.
[15] Kaur G, Arora S. Chaotic whale optimization algorithm[J]. Journal of Computational Design and Engineering, 2018, 5(3): 275-284.
[16] Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm[J]. Knowledge-based systems, 2015, 89: 228-249.
[17] Tuba E, Dolicanin E, Tuba M. Chaotic brain storm optimization algorithm[C]//Intelligent Data Engineering and Automated Learning–IDEAL 2017: 18th International Conference, Guilin, China, October 30–November 1, 2017, Proceedings 18. Springer International Publishing, 2017: 551-559.
[18] Mirjalili S, Mirjalili S. Genetic algorithm[J]. Evolutionary Algorithms and Neural Networks: Theory and Applications, 2019: 43-55.