Journal of Computers Vol. 33 No. 6, December 2022, pp. 143-154
doi: 10.53106/199115992022123306012

Research on Optimization Strategy of Stacker Scheduling in Intelligent Storage System Based on Intelligent Improved Algorithm

Shang-Cai Chi, Ya-Ping Li*

School of Automation Engineering, Tangshan Polytechnic College, Tangshan City 063600, Hebei Province, China
{csc390720896, lyp331811082}@163.com

Received 1 November 2022; Revised 23 November 2022; Accepted 24 November 2022

Abstract. With the vigorous development of e-commerce, higher requirements are put forward for the storage capacity and operation efficiency of the warehousing system to ensure the performance of the entire supply chain in the business process. The factors that affect the operation efficiency and storage capacity of the warehousing system mainly include: storage space planning, shelf design, goods access strategy, stacker scheduling strategy, goods picking efficiency, etc. The main research object of this paper is the optimization of the scheduling strategy of the stacker in the storage system, which can solve the problem of the storage space detention caused by the irregular storage and the empty running time in the scheduling process of the stacker, so as to improve the operating efficiency of the storage system.

Keywords: improved algorithm, stacker, scheduling strategy

1 Introduction

In e-commerce activities, the supply chain system of goods can be divided into production, warehousing, logistics, loading and unloading, use and other links. Although logistics does not directly participate in production and processing, it accounts for half of the total production cost. Therefore, improving the efficiency of commodity supply chain is an important means to reduce production costs and improve commercial competitiveness.

The warehousing system is an important part of the commodity supply chain and a key node for commodity storage and transportation. With the development of e-commerce business, higher requirements have been put forward for the storage capacity and operational efficiency of the warehousing system. The research on warehousing system in China started late. In addition to having a certain technical level in mechanical stability, there are still some problems such as low execution efficiency, insufficient intelligence level, and unreasonable use of warehouse space. Therefore, experts and scholars have done a lot of research on the above problems. Aiming at the problem of scheduling strategy of stackers, single instruction operation or single compound instruction operation is mainly adopted, which solves the problem of time attribute of commodity storage simply from the perspective of control, but does not effectively reduce the empty running time of stackers. The research work of this paper on the current scheduling strategy and storage optimization of stackers mainly focuses on the following points:

1. According to the current mainstream warehousing system, the ideal storage location model and time model function are established to establish the basis for the subsequent improvement algorithm.

2. For the optimization problem of the scheduling strategy of the stacker, this paper takes the minimum operation time of the stacker as the objective function, adopts the improved Grey Wolf algorithm, introduces the idea of multi group and Wolf King's reverse guidance, and combines the discrete particle swarm optimization algorithm in the learning mode to accelerate the iteration speed in the later period of the algorithm, and find the overall minimum operation time and no-load operation time of the stacker.

3. The idea of random recombination threshold is introduced to mark the fitness of Wolf King, so as to avoid the algorithm falling into the local optimal solution in the calculation process.

The structure of this paper is as follows: Chapter 2 discusses the development of intelligent warehousing technology and the research results of relevant scholars. The third chapter discusses the implementation process of the stacker scheduling strategy optimization using the improved Grey Wolf algorithm. The fourth chapter discusses the process of using the improved ant colony algorithm to achieve the optimal solution of storage location planning. The fifth chapter is the conclusion, which describes the shortcomings of this study and further research plans.

* Corresponding Author

2 Related Work

Hamzaoui established a continuous model for a new two-way flow rack AS/RS system, and solved it after discretization. The simulation results show that this discrete modeling method can effectively evaluate the system throughput and other indicators [1]. Silva takes the storage location of goods as a constraint condition when modeling, and proposes a variable neighborhood search algorithm, which further improves the operation efficiency [2], Kaiwen Liu et al. considered the problem of time cut-off when establishing the model, established a model with the minimum energy consumption as the goal, and used the gray wolf optimization algorithm to obtain the minimum energy consumption order sequence [3]. Xiangnan Zhan adopts improved particle swarm optimization algorithm to effectively reduce the delivery time of goods [4]. On this basis, Fandi Meng considered the situation of multiple orders, and adopted an improved genetic algorithm incorporating simulated annealing algorithm to solve the problem, which effectively solved the problem of multiple order allocation [5].

In the research process of the model, some researchers consider how to select allocation goals. G. Gavrilov summarized the zoning standards previously studied, and set the level of cargo turnover [6]. The storage mathematical model designed by Anjiang Cai et al. includes the parameters of different regions [7]. Yang Zhao and others put forward the concept of dynamic adjustment, which breaks through the previous method of given strategy and adds the idea of variable parameters [8]. The second school thinks about problems from the perspective of optimization algorithms. On the basis of a given model or no more changes, it focuses on the design of algorithms. At first, it proposed an accurate algorithm, but this algorithm can not deal with large-scale problems. At this time, the heuristic algorithm began to play a role and can provide a relatively reasonable method in a short time [9]. However, in a large number of studies, traditional intelligent algorithms such as genetic algorithm, particle swarm optimization algorithm and ant colony algorithm are mainly used [10]. Yuling Jiao and others also modified the traditional genetic algorithm to deal with the problem in a way of multi population optimization [11]. Metahri and Hachemi established the expected picking time model of the stacker under the positioning storage strategy in the free all flow back AS/RS system, and verified the validity and accuracy of the model through simulation [12]. Wu studied the compact automatic garage through the method of queuing network [13].

3 Research on the Optimization Strategy of Stacker Scheduling

This section mainly studies the scheduling strategy of the stacker, and takes the running time of the stacker as the optimization goal to reduce the no-load running time of the system.

3.1 Description of Operation Time

The goods in the stereo warehouse are always delivered and received in the form of batch orders, so the optimal target of the stereo warehouse is the minimum completion time of batch orders. If the final completion time can be reduced by reasonably arranging the order of goods in and out, that is, different goods picking paths, or different combinations of different instructions in the batch order, the operation efficiency of the automated stereoscopic library is optimized. In the actual production process, we generally divide the system into single instruction operation and compound instruction operation according to operation instructions. The operation diagram of the stacker to complete different instructions is shown in Fig. 1.

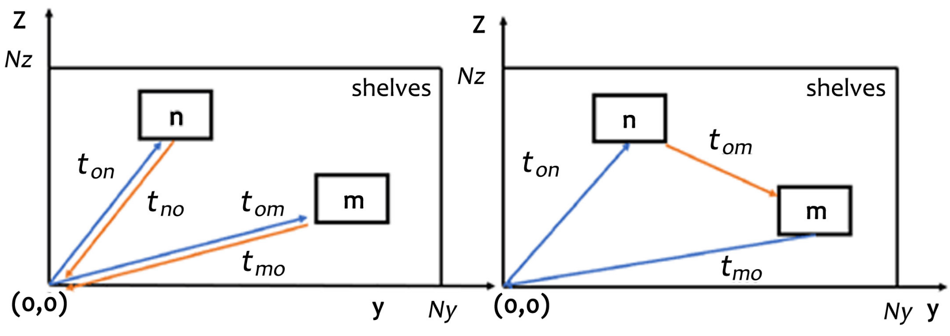


Fig. 1. Stacker operation diagram

According to the schematic diagram of the stacker, during single instruction operation, the stacker completes a single access task, that is, the next outbound (inbound) operation can only be executed after the completion of an inbound (outbound) operation. Therefore, the time required for the stacker to complete an outbound (inbound) operation is shown in Formula 1:

$$t_s = t_{in1} + t_{out1} + t_{in2} + t_{out2}\,. \tag{1}$$

In the formula, t_{in1} is the location time of the stacker from the load in and out of the warehouse to the goods in the warehouse, t_{out1} is the time when the stacker returns to the position in and out of the warehouse with no load from the position in the warehouse, t_{in2} is the location time when the stacker runs with no load to the position in and out of the warehouse, and t_{out2} is the time when the stacker runs with load from the position in and out of the warehouse to the position in and out of the warehouse.

During compound instruction operation, the system automatically matches the issue and receipt of goods in the order. The stacker first transports the goods from the goods receipt port to the designated location, then runs to the corresponding location of the issued goods, and transports the goods to the entrance and exit. At this time, the time required for completing an issue and receipt operation is shown in Formula 2:

$$t_d = t_{in} + t_{io} + t_{out}\,. \tag{2}$$

Where, t_{io} is the time when the storage location of receipt goods moves to the storage location of issue goods.

3.2 Establishment of Operation Model of Stacker

The warehouse studied in this paper is a separated shelf stereoscopic warehouse. First, an abstract three view model of this type of warehouse is established, and it is expressed in the form of three-dimensional coordinates. The warehouse abstract diagram is shown in Fig. 2.

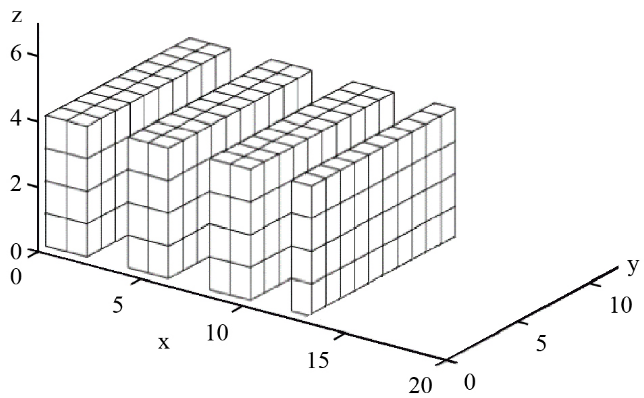


Fig. 2. Abstract drawing of stereoscopic warehouse

Then, according to the actual operation, the side parameters that cannot be measured will be avoided, and the following assumptions are made:

(1) Ignoring the influence of friction, the stacker only runs at a constant speed except for acceleration and deceleration;

(2) The fork inventory and picking of the stacker is a necessary time, and after the whole system is designed, this time is fixed and is not affected by the scheduling strategy, so this time is not included in the strategy optimization problem;

(3) The task is executed according to the serial mechanism, that is, according to the time sequence;

(4) The parameters of each storage location in the shelf are consistent;

Therefore, according to the above assumptions, in the stacker scheduling problem, this paper adopts the composite access mode, and assumes that there is a task $H=\{N,M\}$ to be accessed. In order to obtain the shortest task queue for H, the inventory task set N is expressed as:

$$N=\{n_1,n_2,\cdots,n_i\}. \tag{3}$$

Where, i indicates that there are i goods to be stored in total, and the collection task set M is expressed as:

$$M=\{m_1,m_2,\cdots m_j\}. \tag{4}$$

Among them, j indicates that there are j goods to be stored. The time function of any set of access paths can be expressed as:

$$t(n,m)_i=\left[\left(\frac{|x_n-x_m|}{v_x}\right)^2+\left(\frac{|y_n-y_m|}{v_y}\right)^2\right]_i^{1/2}\cdot L . \tag{5}$$

The symbols in Formula 5 are described as follows:

(n,m) is the task queue, and the corner mark indicates the number with inventory task sequence as counting sequence;

The speed of the stacker at axis x is v_x;

The speed of the stacker at axis y is v_y;

(x_n,y_n) represents the location coordinate of the warehouse to be stocked;

(x_m,y_m) represents the location coordinate of the warehouse to be stocked;

L is the unit shelf length;

The objective function of the shortest time consuming task queue is:

$$f_1=\min\sum_{i=1}^{c}t\{n,m\}_i . \tag{6}$$

c is the minimum pairing integer, and the task queue is expressed as follows:

$$n_1\to m_{h1}\to n_2\to m_{h2}\cdots n_c\to m_{hc}. \tag{7}$$

Among them, picking task m_{hi} is a reset order. Since the number of access tasks is often unequal, the remaining tasks will be accessed in a single mode. The time of single access mode is shown as follows:

$$f_{\text{single}}=t(r)=\sum_{g=1}^{a}\left[\left(\frac{|x_r-0|}{v_x}\right)^2+\left(\frac{|y_r-0|}{v_y}\right)^2\right]^{1/2}\cdot L . \tag{8}$$

The time is integrated into the sum of the time of the composite job time model and the single job model:

$$f = \min \sum_{k=1}^{c} \sum_{n=1}^{i} \sum_{m=1}^{j} t(n,m)_i \cdot x_{nm}^k + f_{\sin gle} . \quad \textbf{(9)}$$

x_{nm}^k is the decision variable. The constraints are established as follows:

$$\sum_{n=1}^{i} \sum_{k=1}^{c} x_{nm}^k = \begin{cases} 1 & m = k, k = 1, 2, 3 \cdots c \\ 0 & other \end{cases} . \quad \textbf{(10)}$$

$$\sum_{n=1}^{i} x_{np}^k - \sum_{m=1}^{j} x_{mp}^k = 0 . \quad \textbf{(11)}$$

In Formula 10, $m = k$ means that after rearrangement, the sequence corner of picking task m_{hi} is consistent with the matching number, that is, when the i pair of picking tasks is m_{hi}, the calculation is valid, and the rest is invalid. Formula 9 is an optimization function for finding the minimum time. In the process of calculation, different queues, Formula 7, will solve the failure result. Constraint 10 is that in each solution process, only the time consumption in the independent variable queue is calculated. Constraint 11 ensures that each calculation is carried out on the same stacker. To sum up, the time optimization model of stacker scheduling is:

$$f = \min \sum_{k=1}^{c} \sum_{n=1}^{i} \sum_{m=1}^{j} t(n,m)_i \cdot x_{nm}^k + t(r) . \quad \textbf{(12)}$$

$t(r)$ can be regarded as a constant value, and the optimization goal is mainly focused on the first item of Formula 12, that is, the composite operation model, which combines the discrete traveling salesman idea. In this paper, the gray wolf algorithm is improved to find the optimal solution.

3.3 Optimized and Improved Grey Wolf Optimizer

The original Grey Wolf Algorithm [14] has some defects, and the main problems affecting the optimization results in this paper are the following two aspects: first, the premature of the algorithm, that is, there will be a local optimal solution, on the other hand, the convergence speed of the algorithm will slow down in the late iteration. Aiming at the above defects, the algorithm is improved as follows to find the optimal scheduling strategy, that is, the minimum no-load operation time of the stacker:

(1) Multiple species compete for the position of Wolf King

The inbound and outbound task sequence of the stereoscopic warehouse is a high latitude natural number variable, and the high latitude function is more likely to have a local optimal solution under the test of the standard test function, and the introduction of multi group thinking can improve the appearance of the local optimal solution. In the initial stage of the algorithm, the entire population is divided into m populations, each population independently searches for prey according to the standard algorithm, and exchanges information after each search. The way of communication is to let the first wolf α of each group compete to produce Wolf King I_α:

$$I_a = \min \{ I_1, I_2, \cdots I_m \}, m \in N+, 1 < m < n . \quad \textbf{(13)}$$

(2) Wolf King's reverse guidance

In the formula, I_i is the first wolf α in each wolf group, wolf king I_α is the wolf with the best fitness in each population, m is the number of the population, and n is the individual synthesis of all wolf groups, which means that the formula is meaningful only when there are multiple populations. The core part of multi population is the communication between populations. The communication between populations often directly affects the effect of the algorithm. The communication between populations adopts the reverse guidance method.

(3) Random recombination threshold

In order to prevent the algorithm from falling into local optimization, this paper sets a random recombination threshold R to mark the fitness value of the elected Wolf King. If the Wolf King changes, the mark is cleared and the counting starts again. If the Wolf King has not changed, and the cumulative mark is greater than or equal to R, the population with the worst fitness is randomly reorganized. The expression of R is:

$$R = \gamma G_1 . \quad \textbf{(14)}$$

γ is the proportional coefficient of open interval $(0,1)$, and G_i is the total evolutionary algebra. The improved algorithm only competes for Wolf King by comparing the values of Wolf α without destroying other wolves, thus ensuring the diversity of the population; At the same time, Wolf King replaces the worst individual of each population to ensure the optimization speed of the algorithm. Finally, the algorithm prevents the algorithm from falling into local optimum by setting a threshold. The flow chart of the improved algorithm is shown in Fig. 3.

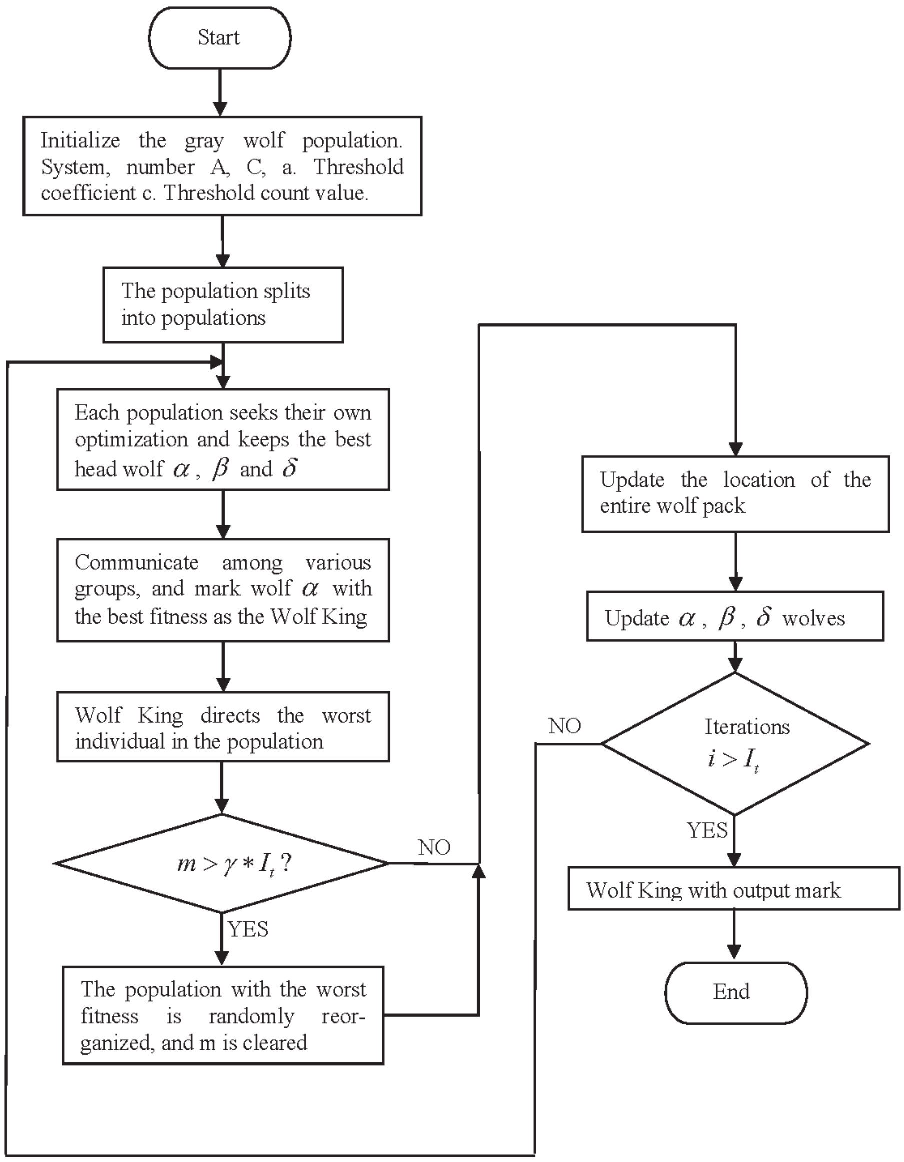
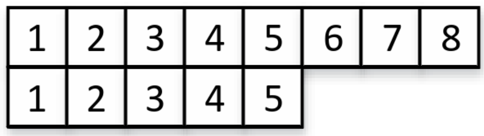


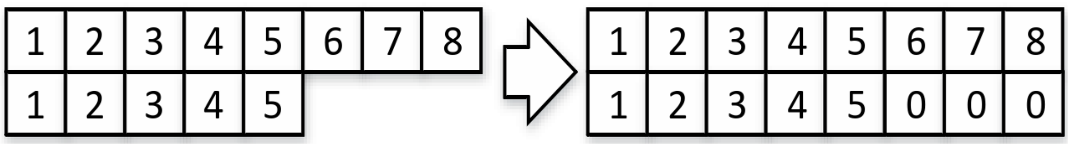
Fig. 3. Algorithm schematic diagram

3.4 Coding and Patching of Algorithms

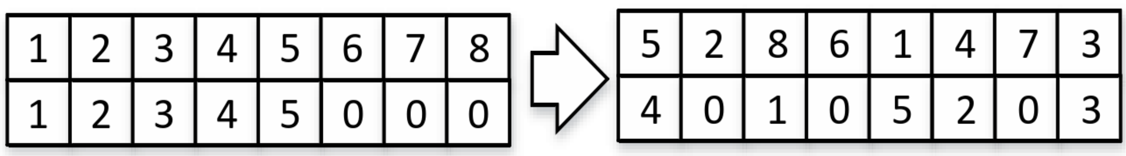
Set λ_1 and λ_0 to represent the issue/receipt quantity of a batch of orders, which is recorded as $[\lambda_1, \ \lambda_0]$. As shown in Fig. 4(a), the first segment is the receipt order serial number of the batch order, and the second segment is the delivery order serial number of the batch order, both expressed in natural numbers. Each serial number randomly generates a storage location (x_n, y_n, z_n). Because the quantity of the issue task and the receipt task does not match, the subsequent algorithm cycle cannot continue. The list with the number 0 indicates that there is no issue/receipt task. Fig. 4(b) shows a complete individual code. After the individual code is patched, it can be further optimized through the algorithm's perturbation strategy. Fig. 4(c) is a diagram after the disturbance. The total completion time of the current batch of orders can be obtained by bringing the position of the current individual into the time energy consumption formula.



(a) Algorithm individual



(b) Algorithm repair



(c) Perturbation of algorithm

Fig. 4. Algorithm individual coding and perturbation

Calculate each individual $X_1 = [2,4,1,3]$, $X_2 = [1,2,4,3]$, $X_3 = [1,3,4,2]$, according to the algorithm update principle $X = (X_1 + X_2 + X_3)/3 = [1.33, 3.00, 3.00, 2.67]$, because there are decimals, the algorithm iteration process will stagnate, so the result after correcting the illegal operator is $X = [1,4,2,3]$.

4 Verification and Simulation of Algorithms

4.1 Testing of Algorithm Effectiveness

In this paper, five typical standard test functions are selected to test the effect of the algorithm improvement. The expressions and basic properties of different types of functions are shown in Table 1.

Table 1. Basic properties of test function

Function	Latitude	Value range	Minimum value
$f_1 = \sum_{i=1}^{D-1}\left[100(x_{i-1} - x_i^2)^2 + (x_i - 1)^2\right]$	30	[− 30, 30]	0
$f_2 = -20e^{\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right)} - e^{\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right)} + 20 + e$	30	[32, 32]	0
$f_3 = -\frac{1}{1.94}\left[2.58 + \sum_{i=1}^{4}\alpha_i e^{\left(-\sum_{j=1}^{6} A_{ij}\left(x_j - P_{ij}\right)^2\right)}\right]$	6	[0, 1]	3.32237
$f_4 = \left(\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}\left(x_i - a_{ij}\right)^6}\right)^{-1}$	2	[− 65.536, 65.536]	1
$f_1\ f_5 = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	[− 5, 5]	0.0003

In the analysis of algorithm parameters, the number of population of the gray wolf algorithm is specified as 60. The Improved Gray Wolf Algorithm uses four populations, with 15 individuals per population. The value of the algorithm evolution algebra is 500. The parameters $c_1 = c_2 = 2$ and $\omega = 0.9$ of the particle swarm algorithm. In order to reduce the accidental factors of the algorithm, the average, variance and standard deviation of the calculation results after 30 times of validation algorithm are taken as the experimental results. The calculation results are shown in Table 2.

Table 2. Test the parameters of the function

Function	Algorithm	Average value	Variance	Standard deviation
f_1	**Improved Grey Wolf Optimizer**	**24.475**	**0.173**	**0.397**
	Grey Wolf Optimizer	27.127	0.614	0.811
	Particle Swarm Optimization	34.156	799.410	27.635
f_2	**Improved Grey Wolf Optimizer**	**8.37x10^{-15}**	**1.79x10^{-30}**	**1.27x10^{-15}**
	Grey Wolf Optimizer	1.91x10^{-14}	9.24x10^{-30}	3.21x10^{-15}
	Particle Swarm Optimization	0.087	0.091	0.310
f_3	**Improved Grey Wolf Optimizer**	**-3.323**	**3.51x10^{-11}**	**1.62x10^{-3}**
	Grey Wolf Optimizer	-3.421	4.99x10^{-3}	6.34x10^{-3}
	Particle Swarm Optimization	-3.430	3.71x10^{-3}	0.0612
f_4	**Improved Grey Wolf Optimizer**	**1.023**	**0.013**	**0.0963**
	Grey Wolf Optimizer	2.001	6.234	2.4536
	Particle Swarm Optimization	1.932	1.332	1.2321
f_5	**Improved Grey Wolf Optimizer**	**3.14x10^{-4}**	**3.37x10^{-15}**	**5.53x10^{-8}**
	Grey Wolf Optimizer	1.98 x10^{-3}	3.26x10^{-5}	5.72x10^{-3}
	Particle Swarm Optimization	1.43x10^{-3}	1.49x10^{-5}	3.64x10^{-3}

It can be seen from Table 2 that Grey Wolf Optimizer performs better than Particle Swarm Optimization in f_1 and f_2, as well as Particle Swarm Optimization in functions f_3 to f_5. Improved Grey Wolf Optimizer combines Particle Swarm Optimization with Grey Wolf Optimizer, which effectively improves the premature problem and population diversity of the algorithm. Therefore, in the five functions of the above test, whether it is multi peak or single peak, whether it is high dimension or low dimension operator, the experimental results, whether mean or variance, are significantly higher than Grey Wolf Optimizer and Particle Swarm Optimization.

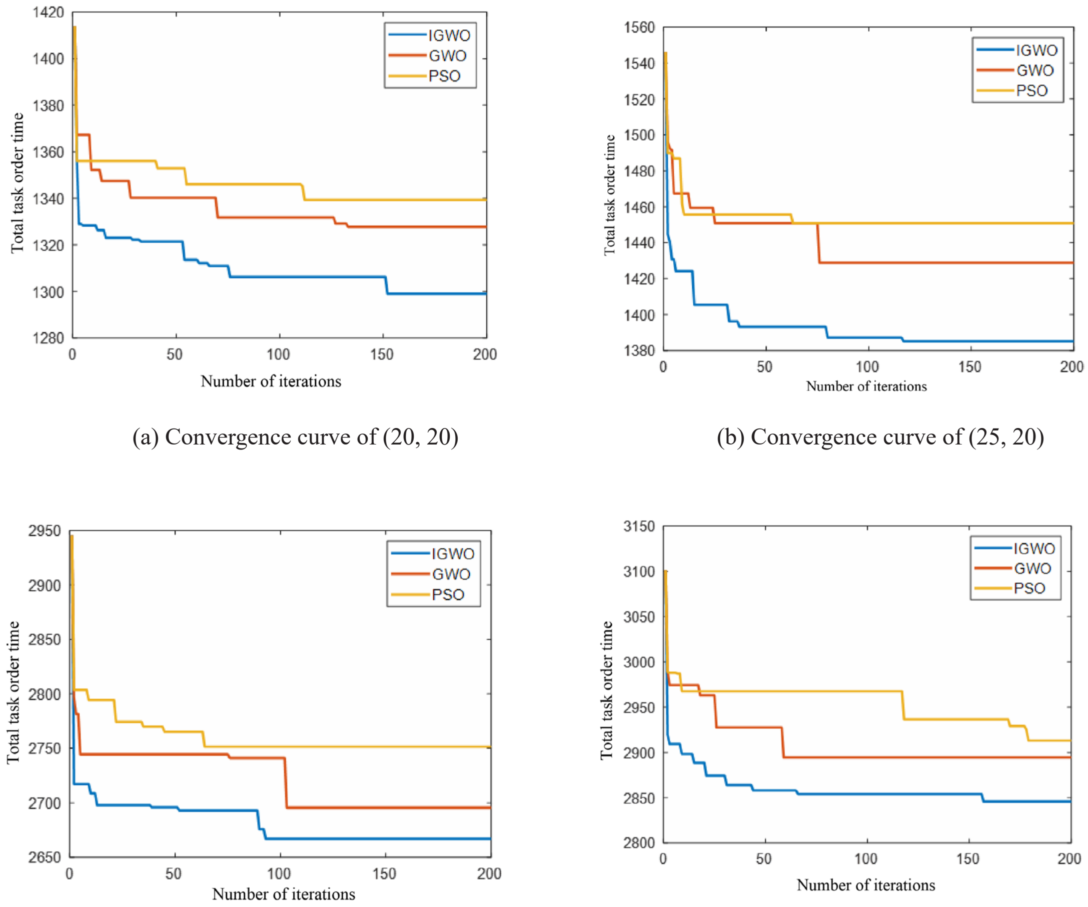
4.2 Operation Time of Stacker Under Given Order Quantity

The area of the pallet is $0.54m^2$, the height of the pallet is $0.6m$. It is composed of the height of the pallet and the height of the cargo. The roadway depth is $50m$. Export the location attributes of batch orders in the warehouse application process, that is, the number of lines and layers of goods on the shelf, as shown in Table 3:

Table 3. 30 calculation results

Warehousing task						Issue task					
Order number	Layer	Row	**Order number**	Layer	Row	**Order number**	Layer	Row	**Order number**	Layer	Row
1	9	23	**26**	5	13	**1**	5	11	**26**	2	47
2	6	47	**27**	4	15	**2**	10	6	**27**	2	32
3	8	36	**28**	4	33	**3**	5	4	**28**	4	40
4	4	9	**29**	9	4	**4**	3	2	**29**	1	14
5	9	15	**30**	6	9	**5**	9	33	**30**	2	27
6	9	48	**31**	4	10	**6**	4	43	**31**	3	3
7	8	11	**32**	3	44	**7**	6	30	**32**	7	2
8	3	24	**33**	3	3	**8**	1	38	**33**	3	4
9	3	7	**34**	10	27	**9**	3	10	**34**	2	19
10	2	24	**35**	9	15	**10**	8	30	**35**	4	40
11	6	26	**36**	8	5	**11**	5	48	**36**	4	43
12	4	27	**37**	9	44	**12**	2	20	**37**	9	27
13	8	12	**38**	4	29	**13**	8	16	**38**	5	6
14	7	8	**39**	9	18	**14**	5	14	**39**	7	4
15	6	31	**40**	9	40	**15**	7	37	**40**	7	5
16	7	29	**41**	2	18	**16**	7	18	**41**	6	49
17	3	9	**42**	4	19	**17**	2	45	**42**	9	24
18	8	4	**43**	4	49	**18**	8	14	**43**	10	38
19	7	25	**44**	7	30	**19**	4	44	**44**	3	19
20	5	47	**45**	8	33	**20**	9	12	**45**	2	6
21	4	9	**46**	3	40	**21**	8	36	**46**	5	50
22	3	21	**47**	6	48	**22**	4	11	**47**	4	22
23	8	25	**48**	10	32	**23**	3	48	**48**	7	14
24	4	16	**49**	9	48	**24**	10	42	**49**	6	34
25	10	48	**50**	4	9	**25**	4	16	**50**	4	30

Figure 5(a) to Fig. 5(d) shows the optimization curve of job completion time under different tasks. It can be seen from the figure that the Particle Swarm Optimization converges faster in solving the model, but it is easy to fall into the local optimum in the later stage of operation. Grey Wolf Optimizer is better than Particle Swarm Optimization in dealing with local optimal solution, but its convergence speed is poor. The Improved Grey Wolf Optimizer makes the algorithm converge fast enough in this model and is not easy to fall into local optimum through multi group communication and random recombination threshold.



(a) Convergence curve of (20, 20)

(b) Convergence curve of (25, 20)

(c) Convergence curve of (45, 45)

(d) Convergence curve of (45, 30)

Fig. 5. Algorithm convergence curve

4.3 No-Load Optimization Simulation of Stacker

Using the data of inbound and outbound goods in Table 3, select 15 ordered inbound goods and 10 outbound goods in batches. The calculation formula of optimization rate is as follows:

$$k = \left(1 - \frac{t_{opt}}{t_{ori}}\right) \times 100\% . \quad \textbf{(15)}$$

The optimized data of no-load operation of different algorithms under mixed operation are shown in Table 4. Through further optimization and improvement of mixed operation by different algorithms, the results show that the empty running time of the stacker can be reduced by optimizing the order of picking tasks, thus optimizing the overall running time of the system.

Table 4. No load operation optimization data of different algorithms under mixed operation

Dispatching scheme	No load time	Optimization rate
Mixed operation sequence	278.43	0
Improved Grey Wolf Optimizer	198.67	29.25%
Grey Wolf Optimizer	209.23	24.43%
Particle Swarm Optimization	207.32	25.41%

Based on the above optimization results, it is now assumed that $\{i, j\}$ is a group of matching tasks in the batch order, i is the receipt task, and j is the issue task. If i or j is 0, it is considered that there is no corresponding task. According to the Improved Grey Wolf Optimizer, the operation sequence can be adjusted as follows:

$$\begin{aligned}&\{3,8\}\rightarrow\{10,0\}\rightarrow\{12,10\}\rightarrow\{13,0\}\rightarrow\{9,2\}\\&\rightarrow\{6,5\}\rightarrow\{15,7\}\rightarrow\{4,3\}\rightarrow\{8,0\}\rightarrow\{14,0\}\\&\rightarrow\{1,1\}\rightarrow\{11,9\}\rightarrow\{2,6\}\rightarrow\{5,0\}\rightarrow\{7,4\}\end{aligned}\quad. \tag{16}$$

After optimization, it can be calculated that the total completion time of the operation is 737.23 seconds, the no-load operation time is 199.33 seconds, and the efficiency is increased by 29.25%. Compare different algorithms, take 30 no-load running time experiments, and record them in Table 5.

Table 5. 30 no-load operation time records

Order number	**Improved Grey Wolf Optimizer**	Grey Wolf Optimizer	Particle Swarm Optimization	Order Number	**Improved Grey Wolf Optimizer**	Grey Wolf Optimizer	Particle Swarm Optimization
1	205.18	**198.60**	204.72	16	**197.01**	205.78	206.91
2	**192.65**	208.62	207.40	17	**196.37**	201.62	208.92
3	**201.15**	206.83	203.40	18	**204.73**	208.12	207.51
4	**197.95**	203.21	206.07	19	**198.06**	207.41	211.76
5	**201.71**	203.40	207.71	20	**194.88**	201.40	197.19
6	**199.22**	202.31	211.88	21	**194.26**	209.82	206.62
7	204.51	211.70	**203.87**	22	205.65	**201.26**	207.59
8	204.71	**203.09**	208.09	23	**200.31**	209.30	203.21
9	**199.53**	208.33	205.64	24	**196.86**	202.37	206.32
10	**201.05**	202.60	205.07	25	**198.89**	210.47	208.36
11	**200.12**	211.27	201.65	26	202.34	199.15	**195.63**
12	**196.58**	204.94	209.74	27	**193.48**	210.45	210.76
13	202.95	207.50	**199.00**	28	193.51	**191.13**	196.60
14	**198.22**	203.63	207.42	29	**202.72**	208.55	203.05
15	203.95	**200.12**	211.08	30	**194.52**	198.31	202.32

The above experimental results show that the improved gray wolf optimization algorithm has the best average value of the calculation results under 30 times of calculation, and the shortest time of occurrence takes up many times. At the same time, the improved algorithm is more stable, and the variance is only 15.66.

5 Conclusion

In this chapter, according to the operation characteristics of the stacker, the time optimization model of the mixed operation is established to solve the relatively optimal task sequence of the inbound and outbound batch orders. In the solution, the initial population is divided into multiple populations for the diversity problem of the initial population. The communication between populations is integrated into the learning mode of the particle swarm optimization algorithm. The strategy of threshold recombination is designed for the precocity problem of the algorithm.

The simulation results show that the improvement of the algorithm is effective. Under the test of the standard test function, the global optimal value is better and the data stability is higher. In the model, it also has faster convergence speed and better global optimal value. Finally, after the second optimization of batch order mixed operation, the empty time of the stacker is effectively reduced, and the processing capacity of batch orders is improved.

In the future, the research goal will focus on other aspects of intelligent warehousing, such as the optimization of inbound and outbound storage locations. The goal is to optimize the whole intelligent warehousing system in multiple directions, so as to propose an optimal intelligent operation system.

6 Acknowledgement

This work is supported by University Level Projects: Study on Signal Timing of Asymmetric Traffic Intersections Based on Fuzzy Logic (YJKT201905).

References

[1] M.-A. Hamzaoui, S. Zaki, Cycle time models for the bidirectional flow-rack AS/RS, FME Transactions 8(1)(2020) 211-226.

[2] A. Silva, L.-C. Coelho, M. Darvish, J. Renaud, Integrating storage location and order picking problems in warehouse planning, Transportation Research Part E: Logistics and Transportation Review 140(2020) 1-20.

[3] K.-W. Liu, Z.-C. Cao, Energy-optimized task scheduling of automated warehouse based on improved grey wolf optimizer, Computer Integrated Manufacturing Systems 26(2)(2020) 376-383.

[4] X.-N. Zhan, L.-Y. Xu, X.-F. Ling, C. Chen, Schdeuling optimization of multi-deep four-way shuttle warehousing system, Computer Integrated Manufacturing Systems 28(8)(2020) 2496-2507.

[5] F.-D. Meng, L. Wang, L. Liu, K.-Y. Zhang, X. Wang, Research on the optimization of the scheduling strategy of the double station shuttle, Manufacturing Automation 44(7)(2021) 21-23.

[6] G. Gavrilov, E. Vlahu-Gjorgievska, V. Trajkovik, Healthcare data warehouse system supporting cross-border interoperability, Health informatics journal 26(2)(2020) 1321-1332.

[7] A.-J. Cai, Y. Cai, S.-H. Guo, C. Geng, Ensemble multi-objective genetic algorithm with application to automated warehouse scheduling, Machinery Design & Manufacture (5)(2019) 95-98.

[8] Y. Zhao, Research on application of automated storage and retrieval system slotting optimization based on improved genetic algorithm, Logistics Engineering and Management 42(2)(2020) 53-56.

[9] Z. Xu, S. Li, X.-J. Hu, Y.-H. Feng, M. Chen, Y.-Y. Fu, Optimal design of wind and solar hybrid power system based on particle swarm optimization, Journal of Zhejiang University of Technology 46(6)(2018) 650-655.

[10]S.-L. Zhang, H.-M. Ma, Y.-Y. Song, Design proposal of automated high-rise warehouse for hazardous wastes, Environmental Sanitation Engineering 30(3)(2019) 90-94.

[11]Y.-L. Jiao, P. Zhang, G.-D. Tian, X.-C. Xing, L.-H. Zou, Slotting optimization of automated warehouse based on multi-population GA, Journal of Jilin University (Engineering and Technology Edition) 48(5)(2018) 1398-1404.

[12]D. Metahri, K. Hachemi. Retrieval–travel-time model for free-fall-flow-rack automated storage and retrieval system, Journal of Industrial Engineering International 14(4)(2018) 807-820.

[13]G. Wu, B. Zou, X. Xu, Shuttle-based operating policies for multiple-lift compact automated parking systems based on queuing networks, Cluster Computing (2018).

[14]S. Mirjalili, M.-M. Seyed, L. Andrew, Grey wolf optimizer, Advances in Engineering Software 69(2014) 46-61.