Research on the Optimization of Picking Path Considering the Minimization of Goods Damage Cost

Jiaxing Lu[1], Hui Zhang[1], Zhenzhuo Zhang[1], Xiaodong Li[2*], Jia Liu[3,a]

[1]College of Automation and Electrical Engineering, Linyi University, Linyi, Shandong, China
[2]Logistics College, Linyi University, Linyi, Shandong, China
[3]Shandong Modern Logistics Technology Industry Research Institute Co., Ltd, Linyi, Shandong, China

*wlxy@lyu.edu.cn; [a]yjy20221@163.com

Abstract. Along with the vigorous development of e-commerce platforms, all kinds of goods with different values appear on various platforms, with the value of goods exhibiting a considerable gap. In this regard, the packaging damage of goods has consistently been an inevitable problem during the picking process. Moreover, customers put forward increasingly high requirements for the integrity of goods. Particularly, the cost caused by the breakage of valuables is usually hundreds of times that of items with lower prices. To address the foregoing challenges, this paper proposes a mathematical model of the multi-vehicle picking path considering the goods damage cost, leveraging an improved ant colony algorithm to solve it. In this foundation, this research further implements relevant simulation experiments through MATLAB. The research findings reveal that the vulnerability value of goods in the second half of the path within the scale of 100 is obviously higher than that in the first half, further verifying the validity of the model and algorithm. Simply put, the methodology proposed in this research improves both the picking efficiency and customer satisfaction.

Keywords: MATLAB; improved ant colony algorithm; path optimization; goods damage cost

1 INTRODUCTION

Recently, the rise of e-commerce has not only brought new impetus to the logistics industry but also imposed inevitable challenges. Relevant research indicates that the cost of picking operations reaches more than 60% of the total distribution cost [1]. As the orders increase, all kinds of goods involving diverse materials and prices appear on e-commerce platforms, with the value gap between some goods reaching several hundred times. Within this context, the breakage of some valuable and fragile goods will reduce the customer's satisfaction with e-commerce platforms while greatly increasing the picking cost. Consequently, the reasonable arrangement of picking paths based on the characteristics and price of goods is of paramount significance.

A. Haldorai et al. (eds.), *Proceedings of the 2024 3rd International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID 2024)*, Atlantis Highlights in Intelligent Systems 11,
https://doi.org/10.2991/978-94-6463-490-7_54

As a whole, scholars at home and abroad have implemented extensive research on the optimization of picking paths. From the perspective of domestic research, on the one hand, Hu [2] constructed a time satisfaction function based on a fuzzy time window and a goods integrity satisfaction function based on the damage rate of goods, taking the maximum customer satisfaction and the minimum transportation cost as the objective functions, which were solved by LINGO17.0 software. Shen [3] took the minimum sum of five costs encompassing goods damage cost as the objective function, which was solved by genetic algorithm (GA). Through the comparison between the scheme aiming at loss cost and the scheme aiming at the shortest path, Du et al. [4] verified the effect of the proposed model in improving customer satisfaction and reducing loss. Additionally, Wang et al. [5] established the TSP model based on the picking path problem of RFID in large warehouses, solving the model related to this methodology by utilizing the hybrid ant colony algorithm. From the perspective of foreign research, on the other hand, Wen [6] proposed an improved ant colony algorithm to address the picking path optimization problem of RFID-based mass storage. Sebo [7] investigated the comparison between GA and other path-related strategies such as heuristic, empirical warehouse selector, and brute-force algorithm under a given hypothesis. Key [8] demonstrated the complexity of the algorithm generated by the picking path as well as the fact that the industry failed to explore it extensively. By summarizing the problems concerning warehouse optimization, Karasek [9] focuses on the coordinated control system of warehouse operation, depicting the typical warehouse operation that depends on the technical and operational structure.

As outlined above, the majority of current researchers put a new premium on picking strategies and intelligent algorithms, especially the solution of the shortest path during the process of warehouse picking, but rarely focus on the research related to the goods damage cost during the picking process. Accordingly, the foregoing research typically lacks practical consideration of the inherent characteristics of goods. Hence, based on the actual demands, this paper proposes a mathematical model of the multi-vehicle picking path considering the goods damage cost, which is solved by an improved ant colony algorithm. Research findings indicate that the average vulnerability value of the second half of the picking path is superior to that of the first half under a certain scale.

2 MULTI-VEHICLE PICKING PATH MODEL CONSIDERING THE GOODS DAMAGE COST

2.1 Problem description and assumptions

Figure 1 illustrates the layout of the dual-partition warehouse. Assuming that there is only one goods space entry and exit point within the warehouse, the order tasks all start from the entry and exit point after being assigned to the picking vehicle. At the end of the picking task, the picking vehicle needs to come to the entry and exit point to wait for the next assigned order distribution task. Hence, the entry and exit point here is also the warehouse access point, which is referred to as the picking and deposit (P&D) point for short, from which the picking vehicle is operated according to the established picking order. In other words, the picking vehicle is required to go to the designated P&D point to perform further picking operations.

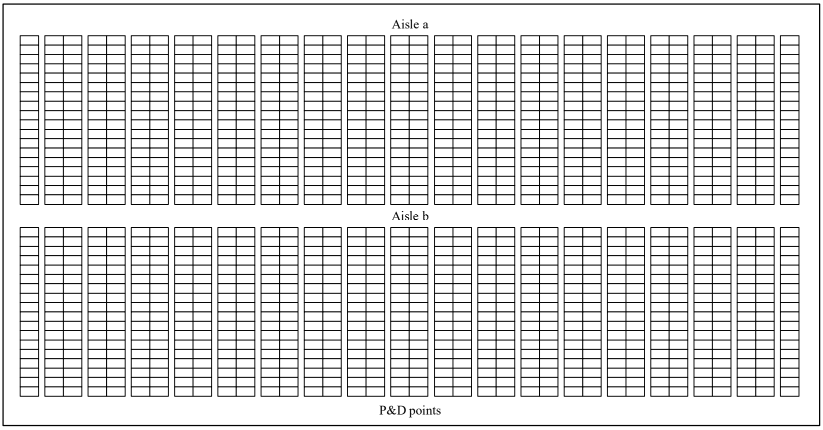

Fig. 1. Planar Graph of the Dual-partition Warehouse

2.2 Problem parameters and distance formula

Figure 1 presents the plane of the whole warehouse, in which a dual-partition warehouse with 18 laneways and 3 transverse passages is built. Among them, each laneway has 36 goods spaces, while each transverse passage has 36 goods spaces. A total of 1296 goods spaces are divided into upper and lower areas by aisle B, with the P&D point serving as the entrance and exit point of the warehouse. On the whole, the position, weight, volume, and vulnerability value of each goods space and its stored goods can be represented by an array [xn, yn, zn, wn, vn, Pn] covering 6 parameters, where n = 1, 2, 3 ... 1296. Specifically, first of all, xn represents the number of the laneway where the n-th cargo space is located and satisfies $0 \leq xn \leq 18$. Secondly, yn indicates whether the goods space is located on the left or right side of the laneway. yn =1 is valid when the n-th goods space is located on the left side of the roadway; otherwise, it is 2. Thirdly, zn indicates the number of rows where the goods space is located, ranging from 0 to 36. Fourthly, wn represents the weight of the goods picked at this point. Fifthly, vn represents the volume of the goods taken. At last, Pn stands for the vulnerability value of the goods stored in this goods space. Notably, the vulnerability value of goods is generated randomly, with a range of [0,1000]. With regard to the specific parameter setting, W represents the width of the warehouse, with the length of the goods space being represented as Wa = 1, the width of the cargo space being represented as Wb = 1, W1=2 denoting the width of the laneway, W2 = 1 denoting the width of the aisle, the starting and ending points of the warehouse being represented as P&D points, and L representing the length of the warehouse. Before solving the foregoing problems, it is imperative to calculate the distance that the person in charge of picking goods walks between the two storage points.

The distance between any two picking points within the warehouse is calculated as follows:

Case 1: In the case that i and j are located in the same picking aisle and both are located above aisle B or below aisle B, the distance at this time is given by:

$$d_{ij} = \left|z_i - z_j\right| \times W_b \tag{1}$$

Case 2: In the case that *i* and *j* are located in the same picking aisle and both are located on both sides of aisle B, the distance at this time is given by:

$$d_{ij} = \left|z_i - z_j\right| \times W_b + W_1 \tag{2}$$

Case 3: In the case that *i* and *j* are located in different picking aisles and both are located above aisle B, the distance at this time is given by:

$$d_{ij} = \begin{Bmatrix} \left|x_i - x_j\right| \times \left(W_1 + 2W_a\right) + \left(72 - z_i - z_j\right) \times W_b \\ \left|x_i - x_j\right| \times \left(W_1 + 2W_a\right) + \left(z_i + z_j - 36\right) \times W_b \end{Bmatrix} \tag{3}$$

Case 4: In the case that *i* and *j* are located in different picking aisles and both are located below aisle B, the distance at this time is given by:

$$d_{ij} = \begin{Bmatrix} \left|x_i - x_j\right| \times \left(W_1 + 2W_a\right) + \left(36 - z_i - z_j\right) \times W_b \\ \left|x_i - x_j\right| \times \left(W_1 + 2W_a\right) + \left(z_i + z_j\right) \times W_b \end{Bmatrix} \tag{4}$$

Case 5: In the case that *i* and *j* are located in different picking aisles and both are located on both sides of aisle B, the distance at this time is given by:

$$d_{ij} = \left|x_i - x_j\right| \times W_b + \left|x_i - x_j\right| \times \left(W_1 + 2W_a\right) + W_2 \tag{5}$$

2.3 Mathematical model of multi-vehicle picking path

Regarding the modeling of problems related to picking path optimization, most scholars regard the shortest path as the objective function of the model. Depending on whether the capacity of the picking vehicle can fulfill the demand of a picking order, the actual picking operation can be divided into two scenarios: one order with one vehicle and one order with multiple vehicles. In this connection, the traditional mathematical model of the multi-vehicle picking path is determined as:

$$\min Z = \sum_{m=1}^{M} \sum_{i=1}^{Nm} \sum_{j=1}^{Nm} d_{ij}^{m} x_{ij}^{m} + \sum_{m=1}^{M} \left(d_{01}^{m} + d_{n0}^{m}\right) \tag{6}$$

Given the obvious discrepancy of goods in the picking center under the e-commerce environment, a host of characteristics of goods, such as storage location, value, and material, exert a great impact on the picking cost. Within this environment, apart from the goods themselves, customer satisfaction also involves logistics to a great extent. Moreover, damaged and squeezed goods packaging, or even direct damage to goods will cause negative comments. Consequently, only taking the total picking distance as the objective function of the model cannot satisfy the demand of reducing the picking cost of the picking center in the actual situation. Simply put, it is imperative to make an in-depth analysis of the picking cost. Meanwhile, since all the vehicles used during the picking process are automated guided vehicles (AGVs), both the picking distance and the goods damage cost should be taken into account. Hence, the model constructed in this paper takes the minimum cost as the objective function based on the traditional model. By adding and changing realistic constraints, the new picking path model is determined as:

$$\lambda = C\left(\sum_{m=1}^{M}\sum_{i=1}^{N_m}\sum_{j=1}^{N_m} d_{ij}^m x_{ij}^m + \sum_{m=1}^{M}(d_{01}^m + d_{n0}^m)\right) \tag{7}$$

where C represents the distribution cost per unit distance, with λ representing the goods damage cost. As AGV acts as the picking vehicle, the carbon emission and fuel consumption cost of the vehicle are not taken into account during driving. Instead, only a certain damage cost, denoted as P_n, caused by the extrusion and collision of goods and the operation error of picking trucks during picking and walking is taken into account. In this research, α is employed to represent the risk coefficient of the goods damage, which refers to the probability of the goods damage per unit distance during picking. Notably, α is calculated by dividing the number of goods damaged per year by the total moving distance related to picking.

The goods damage cost, denoted as λ, is jointly determined by the vulnerability value of the goods within the picking vehicle P_n, the moving distance S, and the risk coefficient α. Mathematically, it is determined as:

$$\lambda = P_n \cdot S \cdot \alpha \tag{8}$$

As the total vulnerability value of the goods within the picking vehicle is increasing with the advance of picking, this paper puts forward a variable, denoted as P_k^m, for recording the total vulnerability value of the goods within the picking vehicle, which represents the total vulnerability value of the goods within the picking vehicle after the m-th vehicle has finished the k-th path, with its unit being yuan. The calculation formula of P_k^m is determined as:

$$P_{k+1}^m = \sum_{i=1}^{N_m}\sum_{j=1}^{N_m} x_{ij}^{mk} p_j + P_k^m \quad (k = 1,2,....N_m + 1) \ \ (m = 1,2,...M) \tag{9}$$

where x_{ij}^{mk} represents the decision variable. In case that the m-th vehicle is in Step k and the picker sends the goods from goods space i to goods space j, $x_{ij}^{mk}=1$ is valid; otherwise, it is 0. In addition, j in p_j represents the vulnerability value of the goods stored in the goods space.

Consequently, $P_k^m \alpha$ represents the goods damage cost per unit distance of the m-th vehicle when it finishes the k-th path, with its unit being yuan. Goods damage cost caused by the whole picking operation can be regarded as the sum of goods damage costs caused by each picking vehicle moving from one picking point to the next until all goods are picked. Goods damage cost is given by:

$$\lambda=\sum_{k=1}^{N_{m+1}}\sum_{m=1}^{M}\sum_{i=1}^{N_m}\sum_{j=1}^{N_m} d_{ij}^m x_{ij}^{mk} P_k^m \alpha+\sum_{k=1}^{N_{m+1}}\sum_{m=1}^{M}\left(d_{01}^m+d_{n0}^m\right)P_k^m \alpha \tag{10}$$

The relevant constraints are presented as follows:

$$\sum_{i=1}^{Nm} W_{mi}\le W_{\max} \tag{11}$$

$$\sum_{i=1}^{N_m} V_{mi}\le V_{\max} \tag{12}$$

$$\sum_{i=1}^{N_m}\sum_{k=1}^{N_m+1} x_{ij}^{mk}=1 \tag{13}$$

$$\sum_{j=1}^{N_m}\sum_{k=1}^{N_m+1} x_{ij}^{mk}=1 \tag{14}$$

$$\sum_{i,j\in Q}^{N} x_{ij}^{mk}\le |Q|-1 \tag{15}$$

$$x_{ij}^{mk}\in\{0,1\} \quad (k=1,2,\ldots N_m+1) \tag{16}$$

$$P_{k+1}^m=\sum_{i=1}^{Nm}\sum_{j=1}^{Nm} x_{ij}^{mk} p_j+P_k^m \quad (k=1,2,\ldots N_m+1)\ (m=1,2,\ldots M) \tag{17}$$

$$p_j\ge 0 \quad P_1=0 \tag{18}$$

As mentioned above, Formula (10) is an objective function, which indicates the damage cost for each idle picking vehicle to pick the goods at the goods points to be picked in the same order set in the shortest distance. Formula (11) depicts a load constraint, which indicates that the overall load of each picking vehicle after picking the goods at a goods point to be picked should not exceed the upper limit of the picking weight that the picking vehicle can carry. In addition, Formula (12) describes a volume constraint, which indicates that the sum of the picking volumes of each picking vehicle after picking the goods at a goods point to be picked shall not be greater than the maximum volume that the picking vehicle can bear. Formulas (13) and (14) present the constraint of the number of visits, indicating that each item can solely pass through one of the picking paths. The constraint summarized in Formula (15) indicates that the picking vehicle will not have a small cycle when picking goods. The constraint set forth in Formula (16) represents a decision variable. In other words, in the case that the m-th picking vehicle picks the items to be picked and continues to perform the next operation, the value of the variable is 1; otherwise, it is 0. The constraint depicted in Formula (17) represents the total vulnerability value of the goods taken by the m-th vehicle. In closing, the constraint outlined in Formula (18) indicates the value of p_j. Here, $P_1 = 0$ indicates that the picking vehicle fails to perform picking when it starts from the origin.

Table 1 provides a mathematical explanation of the symbols and parameters in the above formulas.

Table 1. Description of Symbols

Symbols	**Description**
M	Number of idle picking vehicles available for picking operations.
N	Total number of goods points to be picked.
N_m	Total number of all goods points to be picked by the m-th picking vehicles.
P_k^m	Total vulnerability value of the goods within the m-th picking vehicle after the k-th path.
d_{ij}^m	Picking distance of the m-th picking vehicle from the picking point i to j.
x_{ij}^{mk}	If the m-th vehicle continues to move from the goods point i to j after the k-th path, the value is 1; otherwise, it is 0.
d_{01}^m	Distance from P&D point to the first goods point to be picked for the m-th vehicle.
d_{n0}^m	Distance for the m-th vehicle to return to the P&D point from the last goods point to be picked.
W_{mi}	Increased load capacity of the m-th vehicle after picking the goods at the goods point i to be picked.
W_{max}	Maximum load borne by picking vehicles.
V_{mi}	Increased volume of the m-th vehicle after picking the goods at the goods point i to be picked.
V_{max}	Maximum volume borne by picking vehicles.
$\|Q\|$	Number of elements in a batch of order sets to be picked.

3 SOLUTION METHOD FOR MODEL

3.1 Improved ant colony algorithm

Ant colony algorithm, as a novel heuristic intelligent algorithm, is widely applied to solve the problems related to the planning of warehouse picking paths. In the context of solving the shortest path, it can not only derive more optimized results but also exhibit fewer iterations and shorter picking time compared with GA, simulated annealing algorithm, and tabu search algorithm [10]. Nevertheless, it still inevitably exposes various shortcomings such as the tendency to seek local optimum, slow convergence speed, and poor diversity of solutions. Hence, this paper improves it from the perspective of the initial pheromone and pheromone updating mode, thus accelerating the optimization speed of the algorithm in the initial stage and improving the search efficiency.

1. Improvement of Initial Pheromones

Traditional ant colony algorithm randomly selects paths due to the same pheromone of each path in the early stage, thereby leading to the emergence of potential invalid paths. To improve the searching ability of the classical ant colony algorithm [11] at the initial moment, the distance between the total pheromone and each demand point and distribution center is taken as the pheromone distribution matrix. The improved mathematical formula of the initial pheromone is presented as:

$$\tau_{ij}(0) = \frac{Q}{d_{1i} + d_{1j}} \tag{19}$$

where Q represents the total amount of pheromones, whereas d_{1i} represents the distance from the demand point i to the distribution center, thus increasing the probability of the superior path being selected as well as the optimization ability in the initial stage of the algorithm. Thus, a new node transition probability is given by:

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{allowd_k} \tau_{ij}^{k}(t)\eta_{ij}^{k}(t)} & j \in allowd_k \\ 0 & otherwise \end{cases} \tag{20}$$

2. Strategy for Updating Global Pheromones

The traditional ant colony algorithm optimization process is the performance of improving pheromone concentration. In this connection, the pheromone updating strategy plays a key role in the optimization search of the ant colony algorithm. The traditional ant colony algorithm, however, solely leverages the overall information of ant paths to calculate the pheromone concentration, without distinguishing between the current optimal path and the poor path. Therefore, it fails to change the distribution of pheromones quickly, showcasing poor convergence speed and solution quality. The improved mathematical expression of the pheromone update is as follows:

$$\tau_{ij}(t+1) = (1-p)\tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^{*} \tag{21}$$

where the expressions of $\Delta\tau_{ij}$, $\Delta\tau_{ij}^{kk}$, and $\Delta\tau_{ij}^{*}$ are presented in (2-4):

$$\begin{cases} \Delta\tau_{ij} = \sum_{kk=1}^{sizepop} \Delta\tau_{ij}^{kk} \\ \Delta\tau_{ij}^{kk} = \begin{cases} \dfrac{Q}{l_{kk}}, l_{kk} \le l_{ave}, \text{and the edge (i, j) is the optimal path at present.} \\ 0,\ l_{kk} > l_{ave} \end{cases} \\ \Delta\tau_{ij}^{*} = \begin{cases} \dfrac{Q}{l_{best}}, \text{the edge (i, j) is the optimal path at present.} \\ 0, \text{the edge (i, j) is not the optimal path at present.} \end{cases} \end{cases} \tag{22}$$

where $\tau_{ij}(t)$ represents the pheromone concentration between demand points i and i in the t-th iteration; p represents the pheromone volatilization factor of the path after each iteration; $\Delta\tau_{ij}$ represents the total amount of pheromone changes at demand points i and j in each iteration; $\Delta\tau_{ij}^{kk}$ represents the contribution value of kk-th ant to the pheromone changes of demand points i and j in this iteration; $\Delta\tau_{ij}^{*}$ represents additional compensation for the optimal path obtained at present, further improving the pheromone on this path; l_{kk} represents the total length of the path selected by the kk-th ant; l_{best} represents the total length of the optimal path currently obtained, and; $sizepop$ denotes the total number of ants. Feedback on diverse pheromones [12] of the optimal path and the non-optimal path is helpful for the ant colony to quickly search for the global optimal path.

The overall algorithm flow is as follows:

Step 1: Initialize parameters and set the maximum number of iterations Max. In this step, Formula (2-1) is used to generate the initial pheromone between each storage point, and Nc (iteration times) is initialized to 0, thus determining the parameter value of each function.

Step 2: Create a tabu list and let all ants (i.e., picking vehicles) start from the P&D point and select the next demand point under the premise of satisfying the constraints, which is added to the tabu list.

Step 3: In the case that the delivery vehicle fails to meet the demand of the next customer point[13], the delivery vehicle returns to the delivery center, with the tabu list being updated. This process should be repeated until all customer points are added to the tabu list.

Step 4: Upon completion of the cycle by all ants, pheromones are updated according to Formula (2-3). Simultaneously, the feasible solution obtained by the current iteration is calculated and compared with the feasible solution obtained before to record the optimal solution.

Step 5: In the case of Nc = Nc+1 and Nc < Ma, Steps 2, 3, and 4 are executed; otherwise, the algorithm iteration should be ended to output the optimal solution[14].

4 SIMULATION EXPERIMENT AND ANALYSIS

4.1 Algorithm validity

Taking the dual-partition warehouse shown in Figure 1 as the simulation object, this research employed the improved ant colony algorithm to solve the mathematical model of path optimization considering the goods damage cost, with MATLAB software being utilized to perform the simulation. Picking vehicles are automatic AGVs, with a maximum load of 100 and a maximum volume of 10. The number of ant colony races is 50, and the random generation range of the vulnerable value of goods in this paper is [0, 1000]. The significance of the risk coefficient of cargo damage is the probability of cargo damage per unit distance when picking. According to the average level of the warehouse picking truck, the average cargo damage rate is about 1 / 10000, which is calculated to be about 0.00001. On the premise of making the comprehensive function equal to the objective function, this research implemented a simulation comparison and obtained the results as shown in Figure 2, revealing that the average solution and optimal solution of the improved ant colony algorithm outperform those of the traditional ant colony algorithm. By randomly selecting four picking scales of 10, 30, 50, and 100, this research calculated them 30 times respectively. The results demonstrate that the minimum value obtained by this algorithm has little error with the real optimal value in the case of a small picking scale (i.e., $n \leq 50$). In the case of a large picking scale (i.e., $n \geq 100$), however, the algorithm presents a large calculation error.

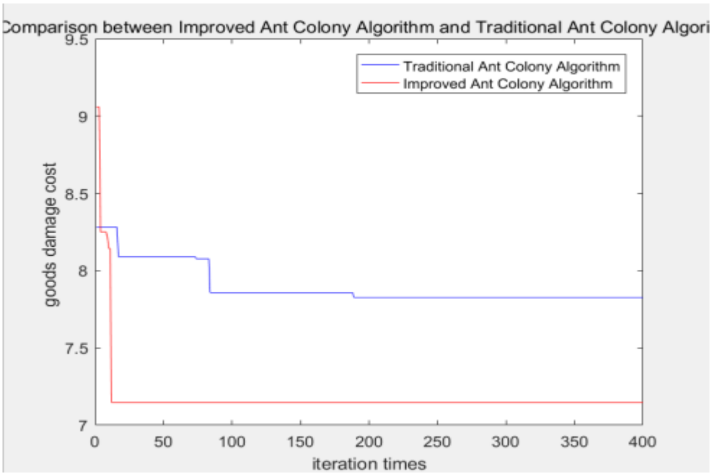

Fig. 2. Comparison between Improved Ant Colony Algorithm and Traditional Ant Colony Algorithm

A 30-scale example is selected, and the maximum number of iterations is changed to 100,200,300, and 400, respectively, and multiple runs are compared to obtain Table 2. It is found that the traditional ant colony algorithm converges to the optimal solution after the number of iterations reaches 200, while the improved ant colony algorithm has converged to the optimal solution after the number of iterations is 100.

Table 2. Cargo damage cost comparison table

algorithm comparison iteration times	**Cargo damage cost under the scale of 100**	**Cargo damage cost under the scale of 200**	**Cargo damage cost under the scale of 300**	**Cargo damage cost under the scale of 400**
Traditional Ant Colony Algorithm	7.86	7.83	7.83	7.83
Improved Ant Colony Algorithm	7.15	7.15	7.15	7.15

5 MODEL VALIDITY

In the case of picking scales of 10, 30, and 50, the calculation considering the risk of goods damage was conducted 30 times, with the goods damage cost λ of the objective function serving as the evaluation standard of the fitness function. As depicted in Figure 3, under the picking scale of 10, the average vulnerability value of goods picked by AGV in the first half of the path is 322.21, whereas that in the second half of the path is 505.82. As depicted in Figure 4, under the picking scale of 30, the average vulnerability value of AGV goods in the first half of Path 1 is 312.45, while that in the second half is 571.55. Likewise, its average vulnerability value in the first half of Path 2 is 488.75, while that in the second half is 572.11. As shown in Figure 5, under the picking scale of 50, the average vulnerability value of AGV goods in the first half of Path 1 is 443, while that in the second half is 576.89. Similarly, the average vulnerability values of its goods in the first half and the second half of Path 2 are 438.44 and 495.77, respectively, while those in the first half and the second half of Path 3 are 571 and 675.83, respectively. At last, under the picking scale of 100, the average vulnerability value of goods in the second half of three of the five paths is better than that in the first half. Thus, on a certain scale, goods with low vulnerability values will be arranged at the front end of the path, while goods with high vulnerability values will be arranged at the back end of the path, indicating that the improved model reduces the moving distance of goods with high vulnerability value by calculating the vulnerability value of goods as well as the picking distance. Furthermore, it effectively reduces the risk of goods damage during picking. Consequently, the output results demonstrate that the proposed model is effective in reducing the goods damage cost.

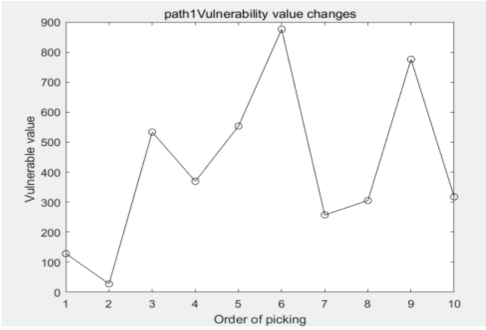

Fig. 3. Change in the Vulnerability Value under the Picking Scale of 10

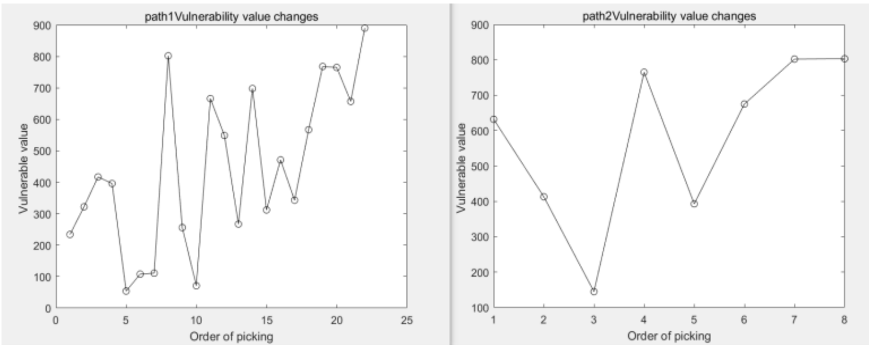

Fig. 4. Change in the Vulnerability Value under the Picking Scale of 30

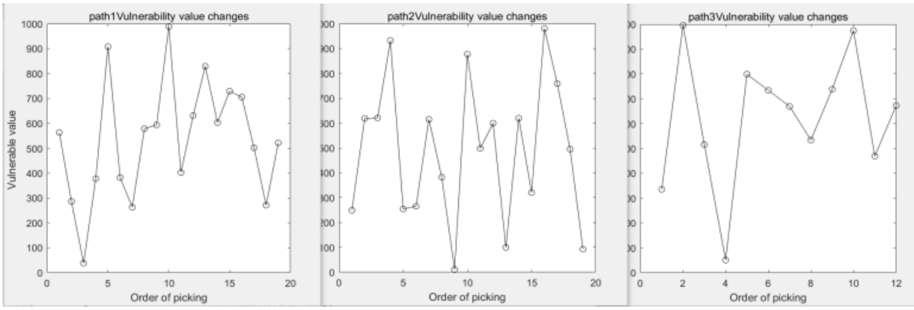

Fig. 5. Change in the Vulnerability Value under the Picking Scale of 50

6 CONCLUSION

Taken together, given the current situation characterized by high goods damage cost and low customer satisfaction during the process of picking in the logistics industry, and the tendency of most scholars within the field of warehouse picking for the mathematical model with the shortest time and distance, this paper proposes a mathematical model of multi-vehicle picking path considering the goods damage cost, employing an improved ant colony algorithm to solve such problems. Through the simulation experiment by using MATLAB, this research further concludes that during the process of picking goods on a certain scale, picking vehicles will arrange more high-value and easily damaged goods at the back end of the picking path, thus verifying the validity of the model and algorithm proposed. Simply put, this research provides a novel idea for the optimization of the picking path. In future research, multiple factors, such as the weight of goods, labor cost, and the wear cost of picking vehicles, will be taken into account to implement comprehensive research that is more in line with the actual demands.

REFERENCES

1. Liu J., Xiong F. and Chen J., Optimization of picking path in dual-partition warehouse based on ant colony algorithm. High-tech communication 27(01), 72-80 (2017).
2. Hu Z., Jia Y. And Li B., Research on vehicle routing optimization considering customer satisfaction. Industrial Engineering 22(01), 100-107 (2019).
3. Shen L., Li C., Gan Y. and Zhao G., Optimization of fresh product distribution routes considering cargo damage and carbon emissions. Journal of Shanghai Maritime University 42(01), 44-49+70 (2021).
4. Du C. and Li Y., Research on cold chain distribution routing problem based on customer satisfaction and minimum loss. Industrial Engineering and Management 25(06), 163-171 (2020).
5. Wang Y., Wen P. and Li L., Research on large warehouse picking path optimization algorithm. Computer simulation 30(05), 337-340 (2013).
6. Wen P., Wang Y. and Li L., Research on Improved Ant Colony Algorithm for the Optimization of the Large Storage Picking Path Based on RFID. Applied Mechanics and Materials, 239-240(239-240) (2012).
7. Sebo J and Busa Jr J, Comparison of advanced methods for picking path optimization: Case study of dual-zone warehouse. METHODOLOGY 3, 14-17 (2020).
8. Key R and Dasgupta A, Warehouse pick path optimization algorithm analysis//Proceedings of the International Conference on Foundations of Computer Science (FCS). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 63 (2015).
9. Karásek J, An overview of warehouse optimization. International journal of advances in telecommunications, electrotechnics, signals and systems 2(3), 111-117 (2013).
10. Chen X., Dai Y. and Chen J., Research on picking path planning in warehouse system. Automation technology and application 40(01), 85-89 (2021).
11. Wen P., Wang Y. and Li L., Research on Improved Ant Colony Algorithm for the Optimization of the Large Storage Picking Path Based on RFID. Applied Mechanics and Materials, 239-240(239-240) (2012).
12. Fan B., Optimization Model of Warehouse Picking Path Based on Simulated Annealing Algorithm//The International Conference on Artificial Intelligence and Logistics Engineering. Springer, Cham, 555-564 (2022).
13. Shen X, Yi H and Wang J, Optimization of picking in the warehouse//Journal of Physics: Conference Series. IOP Publishing 1861(1), 012100 (2021).
14. Liu H, Wang F and Zhao J, Performance Analysis of Picking Path Strategies in Chevron Layout Warehouse. Mathematics 10(3), 395 (2022).