

Intelligent Loading of Scattered Cargoes Based on Improved Ant Colony Optimization

Zhisong Lin^{1*}, Xiu Chen²

¹ Dept. of Military logistics, Army Logistics University, Chongqing 401331, China

² Disong Network Technology Co., Ltd., Chongqing 401331, China

Corresponding Author Email: 463716447@qq.com

<https://doi.org/10.18280/ria.330206>

Received: 6 January 2019

Accepted: 20 March 2019

Keywords:

wall-based ant colony optimization (WBACO), scattered cargoes, volume utilization, expectation function, heuristic factors

ABSTRACT

This paper improves the ant colony optimization (ACO) to optimize the scattered cargo loading problem. Firstly, the concept of scattered cargoes was defined clearly, and a mathematical model was established to maximize the volume utilization under multiple constraints of scattered cargoes. Next, the wall-based loading strategy was put forward to rationalize the spatial arrangement and stabilize the loaded cargoes. After that, the ACO's expectation function was modified to ensure the consistency between cargo selection and the said strategy. In addition, a pheromone heuristic factor and an expected heuristic factor, both of which are dynamically adjustable, were set up to enhance the global search ability of the proposed algorithm, wall-based ACO (WBACO). Finally, three experiments were conducted respectively on classical weakly heterogeneous data, actual production data with weak heterogeneity, and classical strongly heterogeneous data, to verify the performance of our algorithm. In Experiment 1, the WBACO achieved an objective function value 2.6 % higher than the B&R algorithm and 3.1 % higher than the CBGAT. In Experiment 2, the WBACO led the space-based ACO by 6.82 % in average volume utilization and 3.35 % in optimal volume utilization. In Experiment 3, the result of the WBACO was 0.91 % smaller than the B&R algorithm on wtpack7_51, and 6.97 % greater than the latter on wtpack7_74. The experimental results show that the WBACO lays theoretical and practical bases for intelligent loading of scattered cargoes.

1. INTRODUCTION

With the growing attention in intelligent technology, intelligent algorithms have been widely adopted to solve the cargo loading problems. Under current strategies, the cargoes are usually loaded in the patterns of walls, layers [1], towers [2] or blocks [3]. The loading process can be improved through combinatorial optimization [4, 5], mathematical programming [6] and artificial intelligence [7, 8]. Facing heavy computing load and fuzzy convergence conditions, the first two optimization approaches have difficulty in outputting ideal solutions.

This problem can be solved excellently by intelligent algorithms. The ant colony optimization (ACO) is an intelligent algorithm with relatively strong robustness. Capable of distributed computing and positive feedbacks, the ACO has been adopted by many to solve the cargo loading problem [9-11]. However, the ACO may fall into the local optimum trap under excessively strong positive feedbacks, if the parameters are not configured properly. Li [12] proposed three ways to adjust the pheromone amount, aiming to prevent the ACO from early convergence to the suboptimal solution. Yang et al. [13] designed a reward and penalty mechanism to process the trail pheromone after each iteration, and then further optimized the parameters by particle swarm optimization (PSO). Pei [14] developed a dynamic adaptive pheromone heuristic factor and an expected heuristic factor to avoid the local optimum trap.

The boom of express delivery industry has highlighted the importance of loading scattered cargoes. Compared with bulk

cargo, scattered cargoes are extremely complex to load into trucks. In scattered cargo loading, many types of cargoes, each with a small quantity, need to be loaded into the same truck. The various cargoes must be placed in suitable directions and layers. Otherwise, the truck may lose balance, the cargoes may become unstable, and the cargoes at the bottom may be crushed.

Considering the features of scattered cargoes, this paper sets out the principle for wall-type cargo loading, and designs the wall-based ACO (WBACO) to solve the scattered cargo loading problem. The key lies in the spatial arrangement and stability of scattered cargo.

The remainder of this paper is organized as follows: Section 2 defines the concept of scattered cargoes, and sets up a scattered cargo loading model for the optimal volume utilization; Section 3 puts forward the cargo loading strategy, improves the ACO algorithm, and explains how to solve the established model with the improved algorithm; Section 4 verifies the proposed algorithm through experiments and comparative analysis; Section 5 puts forward the research conclusions.

2. PROBLEM DESCRIPTION AND MODELLING

2.1 Problem description

Scattered cargoes are those that have not yet been containerized, and often fall into more than two types. The cargoes in none of the types are large enough in volume or

quantity to fill up a truck. In other words, a truck can carry multiple types of scattered cargoes simultaneously.

On e-commerce festivals like the Double Eleventh in China, a cross-city truck may carry tens of thousands of cargoes, which belong to several to several thousand categories. As a result, the optimization of scattered cargo loading has become a research hotspot. This paper mainly attempts to optimize the loading of scattered cargoes in a single truck for one-to-one distribution (from one start point to one destination).

Our problem can be described as follows: The distribution center has a batch of scattered cargoes. The parameters of each type of cargoes are known in advanced. All cargoes need to be delivered to the same destination. This batch of cargoes should be loaded into a truck with known carrying capacity, in a manner that the center of gravity of the loaded truck falls within the safe area, the cargoes have stable support and bear a rational amount of load, and the carrying capacity of the truck is utilized to the maximum.

2.2 Hypotheses

The following hypotheses were put forward for our problem:

Hypothesis 1: No cargo needs to be loaded into a separate truck.

Hypothesis 2: The truck body and cargoes are all rectangular in shape and nondeformable.

Hypothesis 3: The total size of a type of cargoes does not exceed the truck body in any direction.

Hypothesis 4: The center of gravity of each cargo lies close to its geometric center.

Hypothesis 5: The cargo must be placed in a direction orthogonal to the truck body.

Hypothesis 6: All cargoes are of the same urgency.

Hypothesis 7: All cargoes can withstand a certain load.

2.3 Symbols

To facilitate modelling, a coordinate system (Figure 1) was set up for the truck body, taking the lower left vertex near the cab as the origin and the cab direction as the negative direction of the X-axis. Each cargo was placed into the truck body, such that the right edge, upper edge and front edge point to the positive direction of the Y-, Z- and X- axes, respectively. The remaining space was then divided into the right space, the upper space and the front space.

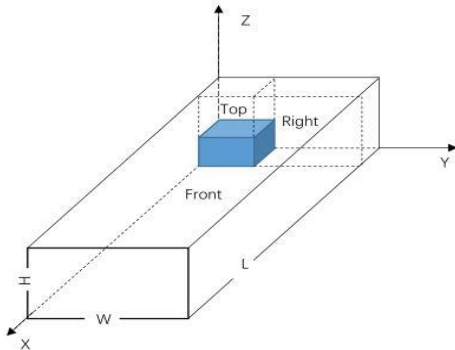


Figure 1. Sketch map of the truck body coordinate system

The symbols used in our research are introduced as follows: M is the number of ants; N is the total number of cargoes; n is the total number of cargo types; $N_{max}(i)$ is the total number of type i cargoes; $N_z(i)$ is the total number of type i cargoes being

loaded; l_i , w_i , h_i and g_i are the length, width, height and weight of type i cargoes, respectively; L , W and H are the length, width and height of the truck body, respectively; $[c1x, c2x]$, $[c1y, c2y]$ and $[0, cz]$ are the allowable intervals for the center of gravity of the truck body on the X-, Y- and Z- axes, respectively; (x_i, y_i, z_i) are the coordinates of the i -th cargo in the truck body; (rx_i, ry_i, rz_i) are the coordinates of the upper right front vertex of the i -th cargo; (lx_i, ly_i, lz_i) are the coordinates of the lower right rear vertex of the i -th cargo; V and G are the maximum volume and maximum carrying capacity of the truck body, respectively; S is the loading information matrix containing the serial number of cargo type, coordinates, directions and spaces of the loaded cargoes (the matrix elements are inputs to the intelligent loading algorithm); Nc_max is the number of maximum iterations; U_i is a decision variable (if the i -th cargo has been loaded, $U_i=1$; otherwise, $U_i=0$).

2.4 Objective function

In this paper, the utilization of the truck's carrying capacity is described as the volume utilization of the truck body (Z). Thus, the objective function can be established as:

$$\max Z = \frac{\sum_{i=1}^N l_i w_i h_i U_i}{V} \quad (1)$$

2.5 Constraints

(1) Quantity constraint. The number of cargoes in each type being loaded should be limited within the following range:

$$N_z(i) \leq N_{max}(i) \quad (2)$$

where, $i=1, 2, \dots, n$.

(2) Volume constraint. The total volume of all cargoes being loaded should not exceed the volume of the truck body:

$$\sum_{i=1}^N l_i w_i h_i U_i \leq V \quad (3)$$

(3) Weight constraint. The total weight of all cargoes being loaded should not exceed the carrying capacity of the truck:

$$\sum_{i=1}^N g_i U_i \leq G \quad (4)$$

(4) Direction constraint. According to Hypothesis 5, each cargo can be placed in several of the following six directions. In the following expression, $l_i//W$ means that the long side of the cargo is parallel to the wide side of the truck body.

$$\left\{ \begin{array}{ll} l_i // W \ \& \ w_i // L \ \& \ h_i // H & \text{direction 1} \\ l_i // W \ \& \ w_i // H \ \& \ h_i // L & \text{direction 2} \\ l_i // L \ \& \ w_i // W \ \& \ h_i // H & \text{direction 3} \\ l_i // L \ \& \ w_i // H \ \& \ h_i // W & \text{direction 4} \\ l_i // H \ \& \ w_i // L \ \& \ h_i // W & \text{direction 5} \\ l_i // H \ \& \ w_i // W \ \& \ h_i // L & \text{direction 6} \end{array} \right. \quad (5)$$

(5) Center of gravity constraint. The center of gravity of the truck body loaded with cargoes should satisfy the following conditions in each direction of the three axes in the coordinate system:

$$\left\{ \begin{array}{l} c1x \leq \frac{\sum_{i=1}^N g_i U_i x_i}{\sum_{i=1}^N g_i U_i} \leq c2x \\ c1y \leq \frac{\sum_{i=1}^N g_i U_i y_i}{\sum_{i=1}^N g_i U_i} \leq c2y \\ 0 \leq \frac{\sum_{i=1}^N g_i U_i z_i}{\sum_{i=1}^N g_i U_i} \leq cz \end{array} \right. \quad (6) \quad \left\{ \begin{array}{l} lz_i \geq rz_j \\ dz(j) - dz(i) \leq Q \end{array} \right. \quad (10)$$

(6) Cargo stability constraint. A cargo is considered stable if and only if (1) its bottom is fully supported by adjacent cargoes or the truck body, and (2) at least one side is supported by adjacent cargoes or the truck body.

In the vertical direction, the following conditions should be satisfied if the i -th cargo is placed above the j -th cargo:

$$\left\{ \begin{array}{l} lz_i = rz_j \\ lx_i \geq lx_j \\ rx_i \leq rx_j \\ ly_i \geq ly_j \\ ry_i \leq ry_j \end{array} \right. \quad (7)$$

In the horizontal direction, the following conditions should be satisfied if the i -th cargo is placed next to the j -th cargo:

$$\left\{ \begin{array}{l} lx_i = rx_j \\ lz_j \leq lz_i \leq rz_j \\ (ly_j \leq ly_i \leq ry_j) \text{ or } (ly_i \leq ly_j \leq rx_i) \end{array} \right. \quad \text{or} \quad \left\{ \begin{array}{l} ly_i = ry_j \\ rz_j \geq lz_i \geq lz_j \\ (lx_i \leq lx_j \leq rx_i) \text{ or } (lx_j \leq lx_i \leq rx_j) \end{array} \right. \quad (8)$$

The cargo stability constraint is satisfied only if both (7) and (8) are valid. This constraint should be modified if the i -th cargo is placed directly above the bottom or next to the walls of the truck body, because the cargo should not exceed the size of the truck body in any direction. The modified conditions can be expressed as:

$$\left\{ \begin{array}{l} lx_i \geq 0 \\ ly_i \geq 0 \\ lz_i \geq 0 \end{array} \right. \text{ and } \left\{ \begin{array}{l} 0 < rx_i \leq L \\ 0 < ry_i \leq W \\ 0 < rz_i \leq H \end{array} \right. \quad (9)$$

(7) Cargo load constraint. The bearing capacity of each cargo was described by the bottom level $dz(i)$. The value of this index is a positive integer $\{1, 2, \dots, n_z\}$. The cargo with a

high bottom layer is relatively suitable to be placed on the bottom layer, i.e. the cargo has a strong bearing capacity. The following conditions should be satisfied to ensure that the cargoes with small bottom levels are placed on the bottom layer:

where, Q is the difference between two cargoes in bottom level.

3. PROBLEM SOLVING

3.1 The ACO

The ACO was proposed by Dorigo et al. [15] in the 1990s, mimicking the foraging behavior of ants. During the search for food, each ant prefers to choose the trail with dense pheromone. Over time, the pheromone trail starts to evaporate, thus reducing its attractive strength. However, the residual pheromone, coupled with the new pheromones released by ants, will continue to affect other ants. Under this positive feedback mechanism, the ant colony manages to continuously optimize the trail. The basic flow of solving the cargo loading problem with the ACO is illustrated in Figure 2 below.

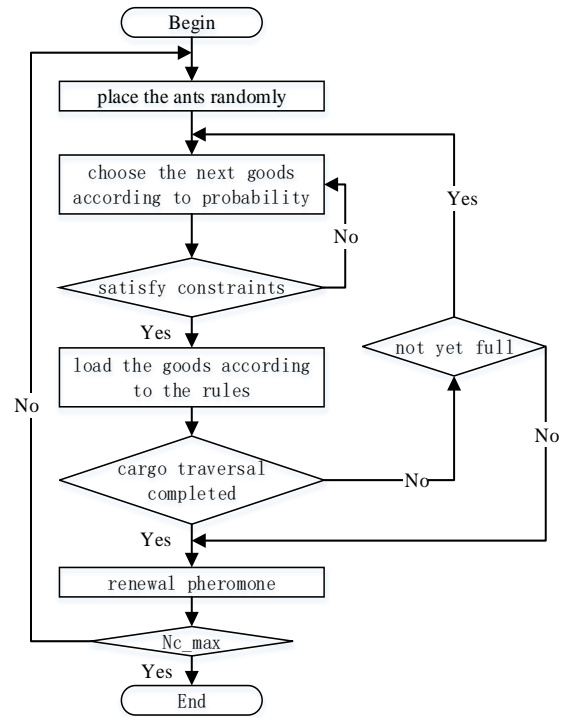


Figure 2. The basic flow of solving the cargo loading problem with the ACO

In Figure 2, the probability that an ant selects the next cargo can be expressed as:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}(t)]^\beta}{\sum_{s \in allow_k} [\tau_{is}(t)]^\alpha \times [\eta_{is}(t)]^\beta}, & j \in allow_k \\ 0, & j \notin allow_k \end{cases} \quad (11)$$

where, $allow_k$ is the set of cargoes to be selected by the k -th ant; $\eta_{ij}(t)$ is the expectation function about how willing the ant is to climb from the i -th cargo to the j -th cargo; $\tau_{ij}(t)$ is the pheromone density of trail ij at time t ; α and β are the pheromone heuristic factor and the expected heuristic factor, respectively. The continued evaporation or accumulation of pheromone can be described by the iterative formulas below:

$$\begin{cases} \tau_{ij}(t+1) = \rho \times \tau_{ij}(t) + \Delta \tau_{ij}, 0 < \rho < 1 \\ \Delta \tau_{ij}(t, t+1) = \sum_{k=1}^M \Delta \tau_{ij}^k(t, t+1) \\ \Delta \tau_{ij}^k(t, t+1) = Q_{\tau} \times Z(Nc, k) \end{cases} \quad (12)$$

where, ρ is the residual pheromone coefficient; $\Delta \tau_{ij}(t, t+1)$ is the sum of pheromones released by all the ants passing through trail ij ; $\Delta \tau_{ij}^k(t, t+1)$ is the pheromone amount newly released by the k -th ant on trail ij ; Z is the volume utilization of the truck body; Nc is the number of iterations; Q_{τ} is the pheromone enhancement coefficient (the value of Q_{τ} is selected empirically).

3.2 Wall-based loading strategy

The wall-based loading strategy is very efficient in loading cargoes. Once a cargo is loaded, the vessel will be divided into a wall space and a non-wall space. The former will be loaded first. When the wall space is fully loaded, a new wall will be built in the non-wall space. This process is repeated until no newer wall can be built in the vessel.

Under the wall-based loading strategy, the space in the truck body can be allocated in advance before loading scattered cargoes, eliminating the frequent space gaps in scattered cargo loading. In the light of the features of scattered cargoes, the following loading principles were prepared:

(1) In the coordinate system of the truck body (Subsection 2.3), the first cargo loaded into the truck body is taken as the reference cargo of the first wall, and the remaining space of the wall is divided into three rectangular subspaces. The next cargo will be placed in one of the subspaces. Once a new cargo is loaded, the remaining space of the wall will be further divided into three subspaces.

(2) In the same wall, the three subspaces should be loaded in the following sequence: the right subspace, the front subspace and the upper subspace. Each wall should be built along the positive direction of the X -axis. In this way, the X -axis dimension of the reference cargo is the thickness of the wall. The horizontal layer within the wall should be filled with cargoes of low bottom levels, before the next higher layer is loaded.

(3) The same type of cargoes should be placed together within the wall. Putting similar cargoes together helps to reduce the use of fillers, lower economic losses and curb environmental pollution. This principle makes it easy to bind up the cargoes and reduce the workload.

(4) Large and heavy cargoes should be placed between the walls. To make better use of space, the relatively large cargo should be selected as the reference cargo of the new wall. The relatively heavy cargo should be placed first near the cab, such that the center of gravity of the truck body is kept close to the middle of the truck.

(5) The reference cargo should be placed with the longest edge along the X -axis direction.

(6) The shortest side of the new space should be maximized. Priority should be given to the cargo that maximizes the shortest side of the new space, such that the new space can accommodate more types of cargoes.

(7) The remaining subspaces should be merged. Let $V_r=[l_r, w_r, h_r, l_{x_r}, l_{y_r}, l_{z_r}]$ be a remaining subspace, where l_r , w_r and h_r are the length, width and height of the subspace, respectively; l_{x_r} , l_{y_r} and l_{z_r} are the X -, Y - and Z - axis coordinates of the lower left rear vertex of the subspace, respectively.

For two subspaces adjacent along the Y -axis, the widths of the two subspaces should be merged as follows if the lower left rear vertices of the two subspaces have the same X - and Z -coordinates.

Let $pp=[1,3,4,5,6]$ be the indexed array.

$$\text{if } l_{z_{r1}} = l_{z_{r2}} \ \& \ l_{x_{r1}} = l_{x_{r2}} \ \& \ (l_{y_{r1}} + w_{r1}) = l_{y_{r2}} \quad (13)$$

$$\text{then } V_{r1}(2) = V_{r1}(2) + V_{r2}(2) \quad (14)$$

$$V_{r1}(pp) = \min(V_{r1}(pp), V_{r2}(pp)) \quad (15)$$

For two subspaces adjacent along the X -axis, the widths of the two subspaces should be merged as follows if the lower left rear vertices of the two subspaces have the same Y - and Z -coordinates.

Let $pp=[2,3,4,5,6]$ be the indexed array.

$$\text{if } l_{z_{r1}} = l_{z_{r2}} \ \& \ l_{y_{r1}} = l_{y_{r2}} \ \& \ (l_{x_{r1}} + l_{r1}) = l_{x_{r2}} \quad (16)$$

$$\text{then } V_{r1}(1) = V_{r1}(1) + V_{r2}(1) \quad (17)$$

$$V_{r1}(pp) = \min(V_{r1}(pp), V_{r2}(pp)) \quad (18)$$

3.3 Adjustment of ACO parameters

(1) Adjustment of the expectation function. The loading strategy expects the next type of cargoes to be loaded within the wall to have similar volume, lighter weight and higher bottom level than the current type of cargoes. After a new wall is built, it is expected to place large, heavy cargoes with low bottom level at the bottom layer of the wall. In addition, priority should be given to the cargo that maximizes the shortest side of the new space. In the light of the above, two expectation functions were established, namely, the inter-wall expectation function $\eta_{ij1}(t)$ (to select the reference cargo of the new wall) and the intra-wall expectation function $\eta_{ij2}(t)$ (to select the next type of cargoes for the current wall):

$$\eta_{ij1}(t) = \frac{v_j \times g_j \times \min_size_j}{dz(j)^2} \quad (19)$$

$$\eta_{ij2}(t) = \frac{g_j \times \min_size_j}{|v_i - v_j| \times dz(j)}$$

where, i is the current type of cargoes; j is the next type of cargoes; v_j and g_j are the volume and weight of a cargo, respectively; \min_size_j is the minimum dimension among the length, width and height of the cargo.

(2) Adjustment of pheromone heuristic factor and expected heuristic factor. In the traditional ACO, there is no limit on the increment of pheromone after each iteration. However, However, the algorithm may fall into the local optimum trap under excessively strong positive feedbacks. To solve the problem, the pheromone heuristic factor α and expected

heuristic factor β should be adjusted to change their impacts on the probability function [14]. In this paper, the timing to adjust the two factors is refined to make the process control more accurate. Specifically, the iterative process was evenly divided into three phases, each of which was further split into three equal parts. Then, the number of iterations in each part can be expressed as $N_{c_max}/9$. In this way, the α and β of each part were calculated and recorded in Table 1 below.

Table 1. The values of pheromone heuristic factor and expected heuristic factor

Stage	1			2			3		
	1	2	3	1	2	3	1	2	3
α	1	2	4	6	8	9	4	2	1
β	9	8	6	4	2	1	6	8	9

3.4 Flow of the WBACO

Step 1: Initialize the algorithm.

Step 2: Place the ants randomly on different types of cargoes, and look up the α and β values from Table 1.

Step 3: Compute the probability function by $\eta_{ij}(t)$, select the first cargo for the new wall, and place the cargo with the longest edge along the X-axis direction.

Step 4: If the constraints permit, select a cargo of the same type with the current cargo, and load the cargo in the same manner. Repeat this step until the wall is filled up with this type of cargoes in the X-axis direction or this type of cargoes are all loaded.

Step 5: If not all cargoes of the current type has been loaded, merge the subspaces. Then, load this type of cargoes in the new remaining space along a different direction. Continue with the loading process until the wall is filled up with this type of cargoes in the new direction or this type of cargoes is all loaded.

Step 6: if all cargoes of the current type have been loaded or if the wall is filled up with this type of cargoes in any direction, merge the subspaces again. If no remaining cargo can be loaded into the new remaining space, go to Step 7; Otherwise, compute the probability function by $\eta_{ij}(t)$, select the next type of cargoes to be loaded into the wall, and then repeat Steps 4~6.

Step 7: If the remaining size of the truck body in the positive direction of the X-axis is greater than the shortest edge of any remaining cargo, repeat Steps 3-7; Otherwise, terminate the loading process of the current ant, and go to Step 8.

Step 8: Evaluate the loading plan of the ant against the center of gravity constraint and the cargo load constraint. If both constraints are satisfied, compute the objective function and update the information of the optimal plan for the current iteration; Otherwise, denote the target function value of the current iteration as zero. Repeat Steps 3~8 until all ants complete the loading task.

Step 9: Update the information of the optimal plan and update the pheromone amount. Judge if the maximum number of iterations has been reached. If yes, terminate the algorithm and output the optimal plan; Otherwise, repeat Steps 2-9.

4. EXPERIMENTAL VERIFICATION

Three sets of data with the features of scattered cargoes were selected. The first dataset is the classical weakly heterogeneous data LN11 and LN02 from Loh and Nee [16].

The second dataset is the loading orders randomly extracted from a bike manufacturer. The third dataset is the classical strongly heterogeneous data wtpack7_41, wtpack7_51 and wtpack7_74 from Bischoff and Ratcliff [17]. The three datasets were used separately to verify the performance of our algorithm in solving weakly heterogeneous problems, practical problems, and strongly heterogeneous algorithms. (The classical data can be found in OR-Library: <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/>.)

In Experiment 1, our algorithm is compared with two classical algorithms: One is the heuristic algorithm proposed by Bischoff and Ratcliff [17] (the B&R algorithm) to enhance the loading stability and uniformity of cargo distribution, and the other is a version of the GA denoted by CBGAT algorithm proposed by Gehring and Bortfeldt [2]. The latter is an improved genetic algorithm (GA) based on the tower loading strategy. In Experiment 2, our algorithm is contrasted with the ACO based on the three-space segmentation (space-based ACO), which was developed by Du et al. [18]. In Experiment 3, our algorithm is compared with the B&R algorithm, because the dataset was created by Bischoff and Ratcliff.

The three experiments were conducted in Matlab R2015a. The CPU model is Intel Core i7-3520 2.9GHz. The parameters were initialized as follows: $M=1.5 \times n$, $c1x=0.1 \times L$, $c2x=0.75 \times L$, $c1y=0.25 \times W$, $c2y=0.75 \times W$, $c2z=0.5 \times H$, $\rho=0.6$ and $Q_{\tau}=10$. It is assumed that the truck equals the truck body in volume, and has a sufficiently large volume-weight.

4.1 Experiment 1

The data LN11 contains 6 types of cargoes, and LN02 contains 8 types of cargoes. The weight and bearing capacity of the cargoes were not given in the original dataset. It is assumed that all cargoes are of the same weight and has the same bearing capacity.

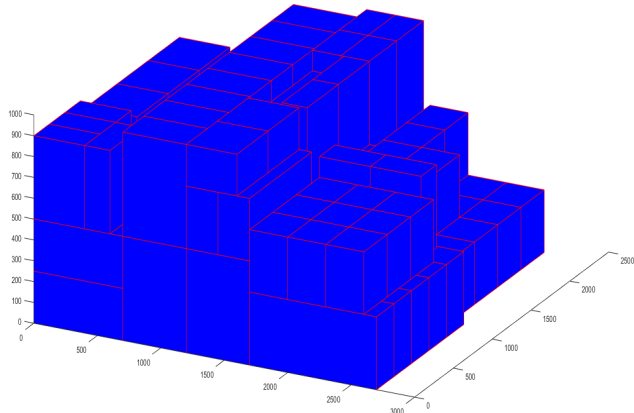
Table 2. Results of Experiment 1

Algorithm	LN11		LN02	
	Left out	Volume utilization(%)	Left out	Volume utilization(%)
B&R	0	62.2	35	90.0
CBGAT	0	62.2	39	89.5
WBACO	0	62.2	40	92.6

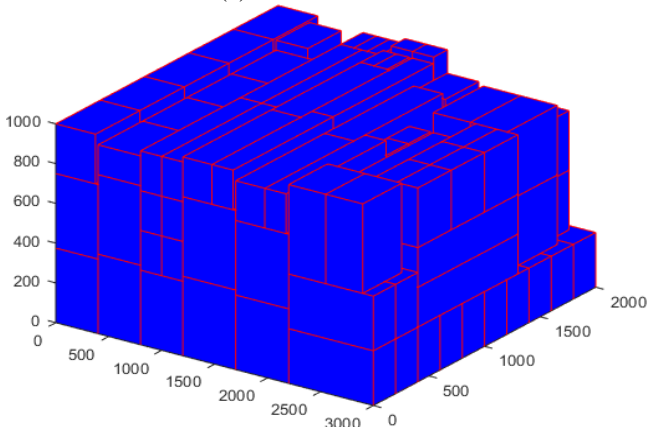
As shown in Table 2, all cargoes in LN11 were loaded into the truck body, indicating that the three algorithms have the same volume utilization. Thus, the proposed algorithm WBACO was proved valid.

Since the total volume of the cargoes in LN02 is 1.11 times the volume of the truck body, it is inevitable that some cargoes cannot be loaded into the truck body. The number of unloaded cargoes in each algorithm is shown in the “Left out” column of Table 2. The proposed algorithm WBACO loaded slightly fewer cargoes than the two contrastive algorithms. This is because our algorithm aims to maximize the volume utilization and gives priority to cargoes with large volume and long edges. As shown in the “Volume utilization” column of Table 2, the WBACO achieved an objective function value of 92.6 %, 2.6 % higher than the B&R algorithm and 3.1 % higher than the CBGAT. The LN02 is a classical data to test the algorithm performance in weakly heterogeneous loading problem. Therefore, Experiment 1 proves the excellence of the WBACO in loading weakly heterogeneous scattered cargoes

into a single truck. The 3D rendering of WBACO results is presented in Figure 3 below.



(a) Results on LN11



(b) Results on LN02

Figure 3. The 3D rendering of WBACO results

4.2 Experiment 2

The data contains 12 types of cargoes, whose bottom levels were obtained from Reference [18]. The quantity of cargoes was not mentioned in the original data. It is assumed that the cargoes are evenly distributed in each type. The truck body follows the size of the international standard container (40 ft): 12.025 m in length, 2.34 m in width and 2.67 m in height. The maximum bearing capacity and maximum load of the truck body were set to 22,000 kg and 75 m³, respectively. The size of the cargoes is specified in Table 3. The algorithms were run twenty times each, and the results are recorded in Table 4 below.

The space-based ACO and the WBACO have good comparability, in that both are extended from the ACO and share similar hypotheses and constraints. It can be seen from Table 4 that, the WBACO clearly outperformed the space-based ACO in average volume utilization, optimal volume utilization and standard deviation. The advantages in the three aspects were respectively 6.82 %, 3.35 % and 0.147. The WBACO also outshined the space-based ACO in the average number of loaded cargoes, thanks to its priority to large cargoes, and the leading edge increased with the total number of cargoes. Judging by the objective function value, the WBACO is clearly superior than the space-based ACO. The algorithm also saves more space and performs more stably. The 3D rendering of WBACO results is presented in Figure 4 below.

Table 3. The data of Experiment 2

Length	Width	Height	Weight	Bottom level	Quantity
1.2	0.27	0.48	14.4	5	60
1.2	0.47	0.56	14.4	5	60
0.69	0.31	0.58	11.8	7	60
0.74	0.195	0.89	14	6	60
0.86	0.19	0.77	14.7	4	60
0.74	0.18	0.85	15.5	3	60
0.86	0.19	0.77	15.7	2	60
0.1685	0.2	0.9	15.9	1	60
0.4	0.4	0.6	8.884	8	60
0.2	0.3	0.4	2.221	10	60
0.2	0.2	0.3	1.1105	11	60
0.4	0.2	0.6	4.442	9	60

Table 4. Results of Experiment 2

Algorithm	Quantity	Utilization (%)	Optimal (%)	SD
Space-based ACO	601	82.22	87.3	0.156
WBACO	568	89.04	90.65	0.009

Note: "Quantity" is the average number of loaded cargoes; "Utilization" is the average volume utilization; "Optimal" is the optimal volume utilization; "SD" is the standard deviation of the volume utilizations obtained through the twenty tests.

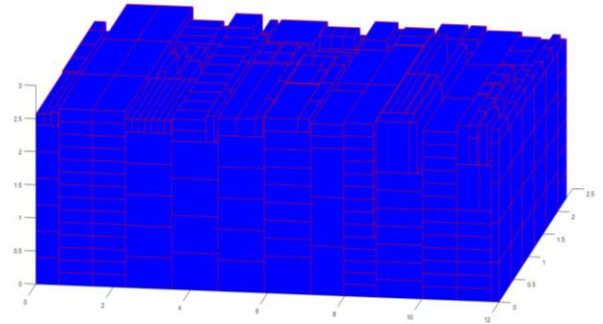


Figure 4. The 3D rendering of WBACO results (Experiment 2)

4.3 Experiment 3

The third dataset is the classical strongly heterogeneous data wtpack7_41, wtpack7_51 and wtpack7_74. The three data each contains 20 cargoes of different sizes. It is assumed that the bearing capacity of a cargo increases with the volume and weight, and each cargo can withstand the same load on all sides. On this basis, the bottom level was configured for each cargo. In addition, the cargoes were allowed to be placed in any of the six directions. The WBACO was run 30 times respectively on wtpack7_41, wtpack7_51 and wtpack7_74. The optimal values on wtpack7_51 and wtpack7_74, and the average value on wtpack7_41 were recorded. The experimental results are shown in Table 5 below.

As shown in Table 5, the WBACO had a slightly poorer performance on wtpack7_51 than the B&R. The performance gap was 0.91 %. However, the WBACO greatly outperformed the latter on wtpack7_74 with an edge of 6.97 %. The WBACO value on wtpack7_41 was 0.85 % higher than the average value of the B&R. Despite the weak comparability between the two algorithms, the above results still prove the feasibility of the WBACO in solving strongly heterogeneous

problems. To sum up, the results of Experiment 3 demonstrate that the WBACO is effective and feasible to tackle strongly heterogeneous problems.

Table 5. Results of Experiment 3

Algorithm	Volume utilization (%)		
	wtpack7_41	wtpack7_51	wtpack7_74
B&R	82.95*	88.28	75.73
WBACO	83.8	87.37	82.7

Note: The B&R results on wtpack7_51 and wtpack7_74 are respectively the maximum and minimum values of the algorithm on the 100 strongly heterogeneous problems in Reference [17]. The asterisk after the 82.95 in the “wtpack7_41” column indicates that the result is the average value of the B&R on the 100 strongly heterogeneous problems. The value of the B&R on wtpack7_41 is not given in Reference [17].

5. CONCLUSIONS

Compared with bulk cargo loading, scattered cargo loading needs to deal with cargoes in multiple types, each with a small quantity. In this paper, the concept of scattered cargoes is defined clearly, and then a mathematical model is established to maximize the volume utilization. The established model is designed for loading scattered cargoes into a single truck for one-to-one distribution. The model constraints include the quantity of cargoes, the volume, load and center of gravity of the truck, the direction of cargo placement, as well as the stability and bearing capacity of the loaded cargoes. Next, a wall-based loading strategy was proposed to ensure the stability of loaded cargoes and rationalize the spatial arrangement, and new expectation functions were designed based on the traditional ACO. Besides, dynamically adjustable heuristic factors were set to enhance the global search ability of the proposed algorithm WBACO. Finally, three experiments were conducted to verify the performance of our algorithm. Experiment 1 was performed on classical weakly heterogeneous data, Experiment 2 on actual production data with weak heterogeneity, and Experiment 3 on classical strongly heterogeneous data. In Experiment 1, the WBACO achieved an objective function value of 92.6 %, 2.6 % higher than the B&R algorithm and 3.1 % higher than the CBGAT. In Experiment 2, the WBACO led the space-based ACO by 6.82 % in average volume utilization and 3.35 % in optimal volume utilization. In Experiment 3, the result of the WBACO was 0.91 % smaller than the B&R algorithm on wtpack7_51, and 6.97 % greater than the latter on wtpack7_74; the mean result of the WBACO on wtpack7_41 was 83.8 %. The experimental results show that the WBACO is feasible to solve weakly heterogeneous problems, actual order problems and strongly heterogeneous problems concerning scattered cargo loading into a single truck, and can converge to the global optimum excellently. The future research will focus on the heterogeneous nature of scattered cargoes.

REFERENCES

[1] George, J.A., Robinson, D.F. (1980). A heuristic for packing boxes into a container. *Computers & Operations Research*, 7(3): 147-156. [https://doi.org/10.1016/0305-0548\(80\)90001-5](https://doi.org/10.1016/0305-0548(80)90001-5)

[2] Gehring, H., Ortfeldt, A. (1997). A genetic algorithm for solving the container loading problem. *International Transactions in Operational Research*, 4(5-6): 401-418.

[https://doi.org/10.1016/S0969-6016\(97\)00033-6](https://doi.org/10.1016/S0969-6016(97)00033-6)

[3] Eley, M. (2002). Solving container loading problems by block arrangement. *European Journal of Operational Research*, 141(2): 382-392. [https://doi.org/10.1016/S0377-2217\(02\)00133-9](https://doi.org/10.1016/S0377-2217(02)00133-9)

[4] Bhattacharya, S., Roy, R., Bhattacharya, S. (1998). An exact depth-first algorithm for the pallet loading problem. *European Journal of Operational Research*, 110(3): 610-625. [https://doi.org/10.1016/S0377-2217\(97\)00272-5](https://doi.org/10.1016/S0377-2217(97)00272-5)

[5] Na, R.S., Cui, X.L., Han, Q.W. (2016). Optimization algorithm for container loading problem with corner constraints. *Industrial Engineering and Management*, 21(1): 1-7.

[6] Lang, M.X., Zhou, X.S., Sun, Y. (2015). Multi-objective optimization of double-deck container train loading. *Transportation System Engineering and Information*, 15(6): 94-100.

[7] Cekała, T., Telec, Z., Trawiński, B. (2015). Truck loading schedule optimization using genetic algorithm for yard management. *Asian Conference on Intelligent Information and Database Systems*, pp. 536-548. https://doi.org/10.1007/978-3-319-15702-3_52

[8] Chen, X., Fan, Y.S., Yu, H.Y., Yang, Z. (2016). Application of adaptive fireworks algorithm in heavy equipment loading. *Science Technology and Engineering*, 16(25): 296-300.

[9] Ndiaye, N.F., Yassine, A., Diarrassouba, I. (2016). An ant colony algorithm to solve the container storage problem. *Forging Connections between Computational Mathematics and Computational Geometry*, pp. 63-73. https://doi.org/10.5176/2251-1911_CMC GS14.37_7

[10] Jovanovic, R., Tuba, M., Voß, S. (2019). An efficient ant colony optimization algorithm for the blocks relocation problem. *European Journal of Operational Research*, 274(1): 78-90.

[11] Yan, S.J. (2016). Research on an improved heuristic method for single-box weak heterogeneous CLP. Dalian Maritime University.

[12] Li, C., Chen, Y.X., Yin, Z.L., Huang, R., Lin, J. (2016). Research on power grid fault recovery decision system based on improved ant colony algorithm. *Electrical Applications*, 35(19): 66-70.

[13] Yang, S.W., Zhang, S.P. (2016). Improved ant colony algorithm and parameter optimization research. *Electronic Technology and Software Engineering*, (13): 186-188.

[14] Pei, Z.B. (2015). Improvement and application of ant colony algorithm. Liaoning University of Science and Technology.

[15] Dorigo, M., Di Caro, G. (1999). Ant colony optimization: a new meta-heuristic. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99*, 2: 1470-1477. <https://doi.org/10.1109/CEC.1999.782657>

[16] Loh, T.H., Nee, A.Y.C. (1992). A packing algorithm for hexahedral boxes. *Proceedings of the Conference of Industrial Automation, Singapore*, pp. 115-126.

[17] Bischoff, E.E., Ratcliff, M.S.W. (1995). Issues in the development of approaches to container loading. *Omega*, 23(4): 377-390. [https://doi.org/10.1016/0305-0483\(95\)00015-G](https://doi.org/10.1016/0305-0483(95)00015-G)

[18] Du, L.N., Zhang, D.Z., Chen, S.F. (2011). Solution to complex container loading problem based on ant colony algorithm. *Journal of Computer Applications*, 31(8): 2275-2278.