Available online at www.HighTechJournal.org

HighTech and Innovation Journal

ISSN: 2723-9535

Vol. 5, No. 2, June, 2024

Optimizing Container Fill Rates for the Textile and Garment Industry Using a 3D Bin Packing Approach

Nguyen Thi Xuan Hoa [1*]

[1] School of Economics and Management, Hanoi University of Science and Technology, Vietnam.

Received 02 February 2024; Revised 22 May 2024; Accepted 28 May 2024; Published 01 June 2024

Abstract

The scarcity of empty containers presents a significant logistical challenge globally. To address this issue, this study proposes the application of the optimal box arrangement in a container with a 3D bin packing problem to enhance fill rates and accommodate the complex packing criteria of the textile and garment industry. The study's objective is to optimize box stacking into containers by considering various factors such as multiple product types, diverse box sizes, varying container sizes, and prioritizing stacking according to purchase orders (PO). In tackling the NP-hard problem with the added constraint of PO-based stacking, this study advocates employing a genetic algorithm combined with a wall-building algorithm to address practical challenges. The genetic algorithm demonstrates optimal efficacy in solving large-scale optimization problems within specified timeframes, yielding high-quality results. In addition, normalization methods are applied to convert box sizes to pallet sizes, expediting problem-solving and facilitating the selection of appropriate container sizes, namely 20- or 40-feet. The research findings indicate that the proposed method achieved a container fill rate of up to 91.67% and minimized the number of containers used.

Keywords: 3D Bin Packing; Genetic Algorithm; Wall Building; Garment and Textile.

1. Introduction

The container shortage issue has become increasingly serious since 2021 due to the COVID-19 pandemic [1]. In 2024, disruptions in Red Sea shipping will further exacerbate the problem, resulting in a 173% increase in freight rates for the Asia–Northern Europe route [2]. This shortage of containers has affected numerous countries and industries, particularly the textile and garment industry, which boasts a market size of US$ 1.7 trillion and contributes approximately 2% to the world's GDP [3]. Container shortages extend lead times and escalate logistic costs [2]. Moreover, with a significant volume of manufactured goods being imported and exported worldwide, constituting 63% of global exports, and textiles and clothing accounting for 4% of manufactured goods [4], optimizing logistics costs and maximizing container fill rates have emerged as crucial concerns. Consequently, addressing the bin packing problem offers a potential solution to minimize transportation costs in container shipping, air cargo loading, and rail transport. The bin packing problem has received extensive attention and development aimed at optimizing container use. It is classified as an NP-hard problem with the objective of efficiently packing items of varying volumes into containers to minimize the number of containers required. This problem has been the subject of study by numerous researchers. The variants of bin packing considered important factors such as multidimensional items, item sizes, item stacking, and rotation [5].

* Corresponding author: hoan65110@gmail.com

http://dx.doi.org/10.28991/HIJ-2024-05-02-017

The problem of packing boxes into containers is challenging with many variables, especially considering the enormous volume of goods exported through seaports. Each exported cargo container that is not optimally filled will result in significant waste in the context of the shortage of empty containers and the current high logistic costs. Therefore, the optimization problem of packing goods into containers is of particular significance in the current global transportation environment. Research on the bin packing issue is primarily based on various variants depending on the characteristics of the container, the nature of the problem, and the characteristics of the carton. The research on the bin packing problem has been extensive, including limited rotations of bin packing [6], constraints of weight limits [7], and stability of stacking [5, 8]. However, studies specifically focusing on the textile and garment industries remain scarce. Considering the unique characteristics of this industry in exportation, priorities in purchase orders extend beyond mere constraints such as box orientation limits, weight limits, and container dimensions. When stacking, adherence to prioritizing orders becomes imperative to streamline unloading at the destination. Nevertheless, previous studies have primarily addressed the arrangement of goods into containers without considering the priority of packing according to the purchase order. Given the industry's specific nature in exports, aligning with the purchase order sequence is also a criterion requiring consideration to facilitate efficient unloading at the destination. Therefore, this paper's contribution focuses on developing and proposing practical and realistic bin packing models suitable for textile and garment enterprises, such as 3D bin packing with constraints related to limited rotation, weight, dimension, and purchase orders (PO). To address the 3D bin packing problem in this research, genetic algorithms and wall-stacking techniques were employed. Unlike previous studies that randomly selected container sizes, this study uses normalization coefficients to determine the appropriate container size.

2. Literature Review

2.1. Bin Packing Problem

The bin-packing problem has become an important issue in many fields, especially in the logistics industry. Cid-Garcia & Rios-Solis [9] solved the two-dimensional (2D) bin-packing problem with a 90° rotation condition using exact algorithms of positioning and covering. Huan et al. [10] proposed a fitting method with direct and indirect algorithms to solve the 2D bin-packing problem, as well as a pseudo-language and complexity evaluation of the algorithms. Kundu et al. [11] proposed a deep reinforcement learning method to solve the 2D bin-packing problem. This method optimized the gaps in a short time and showed superior results compared to other methods at the same time, as well as an easy extension of the problem model into the 3D bin-packing form. The 3D bin-packing problem is an NP-hard problem and has gradually received a lot of attention from researchers over the years because of its highly practical applications. Variants of the 3D bin-packing problem will also be analyzed, including container properties, product properties, packing methods, and safety considerations. Regarding containers, fundamental factors such as type, size, weight limits, and weight distribution will be considered. In reality, logistic companies have various types of containers with different sizes and weight limits. The weight limit of the container ensures that goods are loaded into containers with a weight that does not exceed the maximum recommended weight by research groups [7]. Weight distribution constraints ensure that goods are evenly distributed on the container, avoiding situations where goods are overloaded at the front or rear, causing difficulties in transportation [12]. The stacking constraint ensures that stacking containers on top of each other will not cause damage or breakage. The weight or pressure that a container can withstand depends on four main factors: the material of the container, the products inside the container, the rigidity of the container edges, and transportation-related factors such as time and humidity. This is a very important practical factor, and researchers have focused on developing models that limit the weight and pressure that a carton container can withstand per unit area [13, 14]. In addition, other researchers such as Nishiyama et al. [15] and Paquay et al. [16] have expanded their research directions by limiting the stacking of heavy containers on light containers and not placing fragile items on top of other fragile items.

On another front, when studies address the real-world challenge of handling various types of goods with different delivery priorities, they often discuss the prioritization of goods' lead times as a minor variant of the bin packing problem. The constraint of priority goods often considers optimizing output due to limited container quantities. Therefore, it is necessary to prioritize products with special characteristics, high value, and short lead times. This helps meet customer demand promptly and ensures delivery schedules and minimal losses. In the study by Sheng et al. [17], the issue of prioritizing orders for goods with approaching deadlines over orders with longer deadlines was addressed. The authors used a simulated annealing algorithm and then packed the product containers into containers through a treegraph search process. The results achieved after testing were around an 85% container fill rate.

The product orientation constraint in container stacking ensures that carton containers can be stacked in six orientations parallel to the container edges, with some orientations restricted based on product characteristics. Researchers have developed variations, including unlimited orientations, to accommodate different product types, as introduced by various research groups (Kurpel et al. [18], Mahvash et al. [19], Sharma [20]). The product flow constraint ensures that different types of products can be packed into the same container or not. The complete delivery constraint is crucial for orders where customers request all items to be delivered in the same shipment. For safety issues in cargo stacking, ensuring the stability and integrity of containers is crucial because instability can damage cargo, pose safety risks, and cause injury to workers during cargo handling. Variants of the bin-packing problem that ensure stability have been significantly emphasized in studies by Olsson et al. [8] and Oliveira et al. [21]. Stability is divided into two main types: static stability and dynamic stability. Static stability (in the vertical direction) is applied when the containers are not moving. This constraint ensures that the containers are securely standing on the container floor and the surfaces of other boxes [22]. Dynamic stability (in the horizontal direction) is relevant when containers are in motion, experiencing forces such as acceleration, braking, and inertia [8, 15]. As for cargo stacking positions, products with specific size and weight requirements may need to be positioned on the floor or in designated spots within the container [23]. Weiwang Zhu et al. [5] solved the 3D bin packing problem with stacking constraints using two subproblems: the stacking problem and the 2D bin packing problem. The brand and price algorithm was applied to minimize the total number of containers used. The studies on the bin-packing problem that have been conducted previously addressed one or more constraints simultaneously, such as weight, priority of packing, orientation, hazardous goods, and multiple delivery points, in order to maximize the amount of cargo loaded into containers [24, 25].

The approaches for solving the bin packing problem include exact and approximate algorithms. These exact algorithm models can solve 3D packing, the Single Knapsack Problem (SKP), the Single Large Object Placement Problem (SLOPP), the Container Loading Problem (CLP), and the Multiple Bin-Size Bin Packing Problem (MBSBPP) [26, 27]. They optimize solutions while considering real-world constraints like weight distribution, stability, and item prioritization, with achievements including optimal solutions for various case scenarios. However, exact algorithms can find optimal solutions for small-sized problems but often face difficulties when solving large and complex problems. Kevin Tole et al. introduced a new variant of the bin packing problem called the Circular Bin Packing Problem with Rectangular Items (CBPP-RI), which involves the dense orthogonal packing of rectangular items into a minimum number of bins. They applied Simulated Annealing (SA) to present the its effectiveness compared to previous algorithms in solving a bin packing problem and CBPP-RI [28].

2.2. Genetic Algorithm for Solving Bin Packing Problems

Then, approximate algorithms often provide the best possible solutions within a given time by exploring a search space and iteratively making changes to complete solutions to improve their quality. Their advantage lies in their ability to solve complex problems with real-world constraints and large scales in the shortest possible time. This is a strength of approximate algorithms compared to exact algorithms in optimizing solutions. As a result, many researchers have applied and developed exact algorithms to solve bin packing problems, especially 3D bin packing problems.

The Genetic Algorithm (GA) uses a fitness function in each iteration to measure the quality of each generation and combines the most desired characteristics to improve the quality of solutions. Due to its excellent result quality and runtime, it has been widely applied to 3D-PPs problems. Olsson et al. [8] used a greedy-style heuristic where boxes are first packed into piles, and then potential positions are evaluated using a weighted scoring function. The authors used a GA to automatically adjust the weighted scoring function and improve packing quality. Xiang et al. [29] also used box order and rotation as two encodings in their GA. Building on a wall-building heuristic, the authors developed an adaptive GA that adjusts the probability of crossover and mutation based on individual fitness.

The multiple bin size and box size problem with constraints such as weight and dimensions has been addressed using hybrid metaheuristics based on simulated annealing (SA) and genetic algorithms (GA) to minimize the number of boxes needed to be packed. The algorithm demonstrates that it can find an optimal arrangement model with minimal cost [30]. Ananno & Ribeiro [31] applied a genetic algorithm to minimize the number of pallets used, maximize the compactness of the packed items, and minimize the heterogeneity of item types in each pallet. They proposed a model to satisfy eight different constraints: item orientation, non-conllision, stability, support, pattern complexity, complete shipment, customer positioning, and layer interlocking. The model respects volume utilization when introducing additional load carriers, which is commonly practiced in the F&B industry. Ying Yang et al. explored a novel approach by redesigning bin sizes to fit items ready to be packed. They considered a general three-dimensional open-dimension probem (3D-ODP) where all dimensions of a number of heterogeneous bin types are unknown. The objective of this study is to minimize total costs based on the designed bin types and packing scheme. The combination of the 3D ODP and the three-dimensional multiple bin size bin packing problem (3D MBSBPP) was solved by two-layer heuristics, including a genetic algorithm (GA) and an inner deterministic constructive heuristic [32].

In summary, bin packing problems have been studied and solved for various cases with constraints such as size and weight, aiming to minimize the number of boxes packed into containers. However, for the specific nature of garment exports, in addition to constraints on box size and container size, the items need to be prioritized according to each purchase order. Therefore, this study focuses on the problem of real constraints such as multiple container types, multiple product types, and multiple purchase orders (PO) with different sizes and colors of products. Moreover, the research illustrated the efficiency of heuristic algorithms in tackling.

3. Methodology and the Model

In this study, the research process (Figure 1) is divided into the following main stages: research model development, algorithm model construction, data analysis execution, and model evaluation.

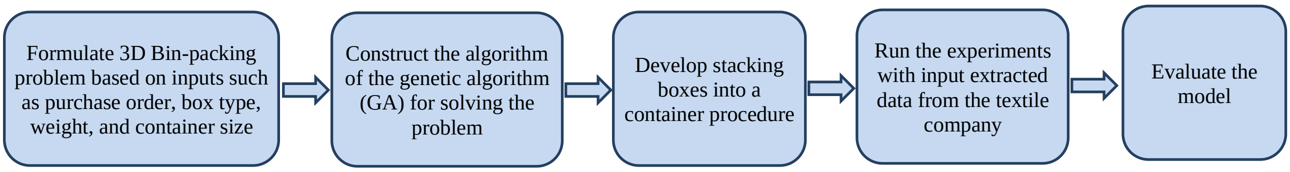


Figure 1. Process of research

3.1. 3D Bin Packing

The 3D bin-packing problem is an NP-hard problem where the initial input consists of small carton containers that need to be packed into larger containers (bins) to minimize the number of containers used. For the textile and garment industry, which has unique characteristics such as high volume, diverse designs, numerous orders, and varied sizes and colors, the 3D bin-packing problem is tailored with the following features:

- Carton containers from the same purchase order (PO) are stacked together (without mixing items from different POs).
- Various types of containers are used, including 20- and 40-feet containers.
- Carton containers can be rotated.
- Other constraints, such as load distribution and stacking, are considered due to the lightweight nature of textile and garment products.

The problem of packing carton box (i, j) needs to be arranged to container k. Each box i, j has three corresponding dimensions $(l_i,\ w_i,\ h_i)$ (cm) representing the length, width, and height. Each box also has a specific weight q_i (kilograms). Container k has similar dimensions and load capacity (L_k, W_k, H_k) (in centimeters) for length, width, and height, with a load capacity Q_k (in kilograms). In this mathematical model, the research team employs the "front-left-bottom" (FLB) method to arrange the containers within the container. For example, each box i, when arranged within container k, is determined by coordinates $(x_{ik},\ y_{ik},\ z_{ik})$ corresponding to a coordinate axis attached to the container. Each box i has initial dimensions of length, width, and height represented as $(l_i,\ w_i,\ h_i)$ in centimeters. However, when placing this box into a larger container, there are six different orientations $(\delta_{1i},\ \delta_{2i},\ \delta_{3i},\ \delta_{4i},\ \delta_{5i},\ \delta_{6i})$, leading to changes in dimensions compared to the original size of box i. The actual dimensions of box i when placed into container k will be $(l'_i,\ w'_i,\ h'_i)$ corresponding to the various rotations of the box.

When using the FLB method (front – left – bottom), box i will have three relative positions with respect to other boxes within container k (front, left, or bottom of box j), which are determined by three corresponding variables: $(a_{ijk}, b_{ijk}, c_{ijk})$ (Figure 2, Tables 1 to 3).

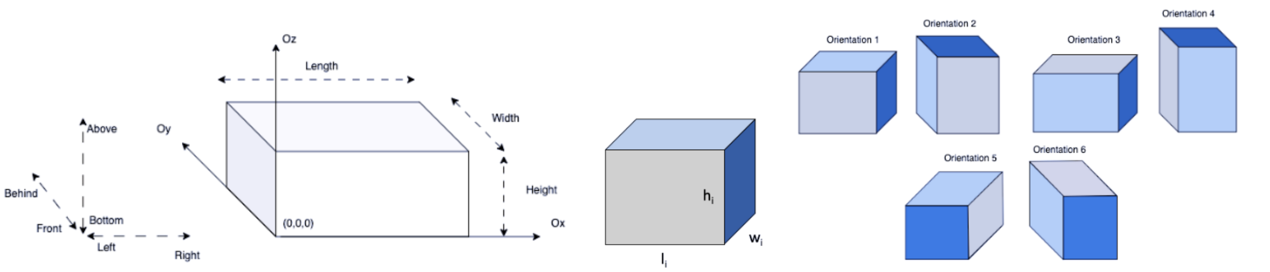


Figure 2. Specification of carton box *i* and different orientations for stacking carton box *i*

Table 1. Sets and Indexes

Index	Description
n	The number of carton boxes that need to be packed.
i,j = {1,…n}	Box
k	Container
V {n}	Container set
A {(i,j)}	Box set

Table 2. Parameters

Index	Description
l_i	Length of box i
w_i	Width of box i
h_i	Height of the box i
L_k	Length of container k
W_k	Width of container k
H_k	Height of container k
q_i	Weight of box i
Q_k	Max weight of container K

Table 3. Variables

Index	Description
a_{ijk}	Binary variable. If product i is placed to the left of product j in container k.
b_{ijk}	Binary variable. If product i is placed behind product j in container k.
c_{ijk}	Binary variable. If product i is placed below product j in container k.
(x_{ik}, y_{ik}, z_{ik})	Continuous variable. The coordinates of box i in container k according to the FLB method.
γ_k	Binary variable. If container k is used.
β_{ik}	Binary variable. If box i is packed into container k.
$\delta_{1i}, \delta_{2i}, \delta_{3i}, \delta_{4i}, \delta_{5i}, \delta_{6i}$	Binary variable. Rotations of box i.

Objective function: Minimize the number of containers used.

$$\textbf{MIN}\ (\textstyle\sum_{k\in V}\gamma_k)$$

Constraints:

* *Packing:*

$$\beta_{ik} \le \gamma_k\ ,\ \forall\ \text{i} \in \text{A},\ \forall\ \text{k} \in \text{V} \tag{1}$$

$$\textstyle\sum_{k=1}\beta_{ik} = 1\ ,\ \forall\, i \in A,\ \forall\, k \in V \tag{2}$$

$$a_{ijk} + b_{ijk} + c_{ijk} = 1,\ \forall\, i, j \in A,\ \forall\, k \in V \tag{3}$$

$$\delta_{1i} + \delta_{2i} + \delta_{3i} + \delta_{4i} + \delta_{5i} + \delta_{6i} = 1\ ,\ \forall\, i \in A \tag{4}$$

* *Dimensions:*

$$l'_i = \delta_{1i}.\ l_i + \delta_{2i}, l_i + \delta_{3i}.\ w_i + \delta_{4i}, w_i + \delta_{5i}.\ h_i + \delta_{6i}.\ h_i\ ,\ \forall\ \text{i} \in \text{A} \tag{5}$$

$$w'_i = \delta_{1i}.\ w_i + \delta_{2i}, h_i + \delta_{3i}.\ l_i + \delta_{4i}, h_i + \delta_{5i}.\ l_i + \delta_{6i}.\ w_i,\ \forall\, i \in A \tag{6}$$

$$h'_i = \delta_{1i}.\ h_i + \delta_{2i}, w_i + \delta_{3i}.\ h_i + \delta_{4i}, l_i + \delta_{5i}.\ w_i + \delta_{6i}.\ l_i,\ \forall\, i \in A \tag{7}$$

* *Container dimension:*

$$x_{ik} - x_{jk} + L_k.\ a_{ijk} \le L_k - l'_i\ ,\ \forall\ \text{i, j} \in \text{A},\ \forall\ \text{k} \in \text{V} \tag{8}$$

$$y_{ik} - y_{jk} + W_k.\ b_{ijk} \le W_k - w'_i,\ \forall\, i, j \in A,\ \forall\, k \in V \tag{9}$$

$$z_{ik} - z_{jk} + H_k.\ c_{ijk} \le H_k - h'_i,\ \forall\, i, j \in A, \forall\, k \in \tag{10}$$

$$0 \le x_{ik} \le L_k - l'_i\ ,\ \forall\, i \in A,\ \forall\, k \in V \tag{11}$$

$$0 \le y_{ik} \le W_k - w'_i\ ,\ \forall\, i, \in A,\ \forall\, k \in V \tag{12}$$

$$0 \le z_{ik} \le H_k - h'_i,\ \forall\, i \in A,\ \forall\, k \in V \tag{13}$$

* *Weight:*

$$\textstyle\sum_{i=1}\beta_{ik} * q_i\ \le\ Q_k * \gamma_k,\ \forall\ \text{i} \in \text{A},\ \forall\ \text{k} \in \text{V} \tag{14}$$

** Variables:*

$$x_{ik},\ y_{ik},\ z_{ik} \in N,\ \forall\ i \in A,\ \forall\ k \in V \tag{15}$$

$$\delta_{1i},\ \delta_{2i},\ \delta_{3i},\ \delta_{4i},\ \delta_{5i},\ \delta_{6i}\ \in \{0,1\},\ \forall i \in A \tag{16}$$

$$\gamma_k \in \{0,1\}\ ,\ \forall k \in V \tag{17}$$

$$\beta_{ik} \in \{0,1\}\ ,\ \forall i \in A,\ \forall k \in V \tag{18}$$

$$a_{ijk}\ ,\ b_{ijk}\ ,\ c_{ijk} \in \{0,1\}\ ,\ \forall i,j \in A,\ \forall k \in V \tag{19}$$

Equation 1: Box *i* is placed into container *k* if and only if container *k* is used. Equation 2: Box *i* is only allowed to be placed into one container, *k*. Equation 3: Product *i* can only have one relative position with respect to product *j*. Equation 4: Product *i* can only be packed into container *k* in one way. Equations 5 to 7: The actual dimensions of product *i* depend on its packing orientation. Equations 8 to 10: Represent the relationship and relative positions of products *i*, *j* packed into container *k*, as shown in Figure 3. Equations 11 to 13: The size of box *i* must not exceed the size of container *k*. Equation 14: The total weight of all boxes *i* must not exceed the weight capacity of container *k*. Equations 15 to 19: Ensure binary variables and non-negativity. Figure 3 illustrates the stacking in the case of multiple boxes.

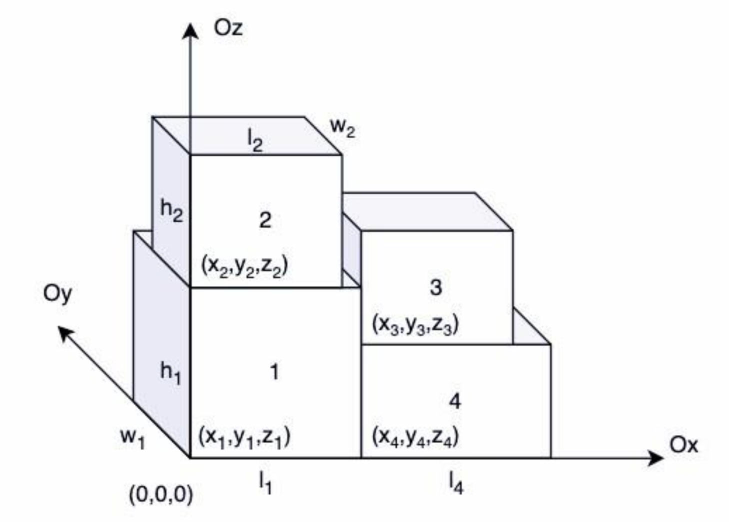


Figure 3. Coordination of each carton box

3.2. The Solution Using the Genetic Algorithm (GA) to Solve the Problem

Steps of the Genetic Algorithm:

```
Begin:
    t = 0;
    Initialize the initial generation P(t);
    Evaluate the fitness of individuals in P(t);
Repeat:
    t = t + 1;
    Generate a new generation P(t) from generation P(t-1) by:
    (i) Selection;
    (ii) Crossover;
    (iii) Mutation;
    Evaluate population P(t) (according to fitness function);
    Termination condition;
End.
```

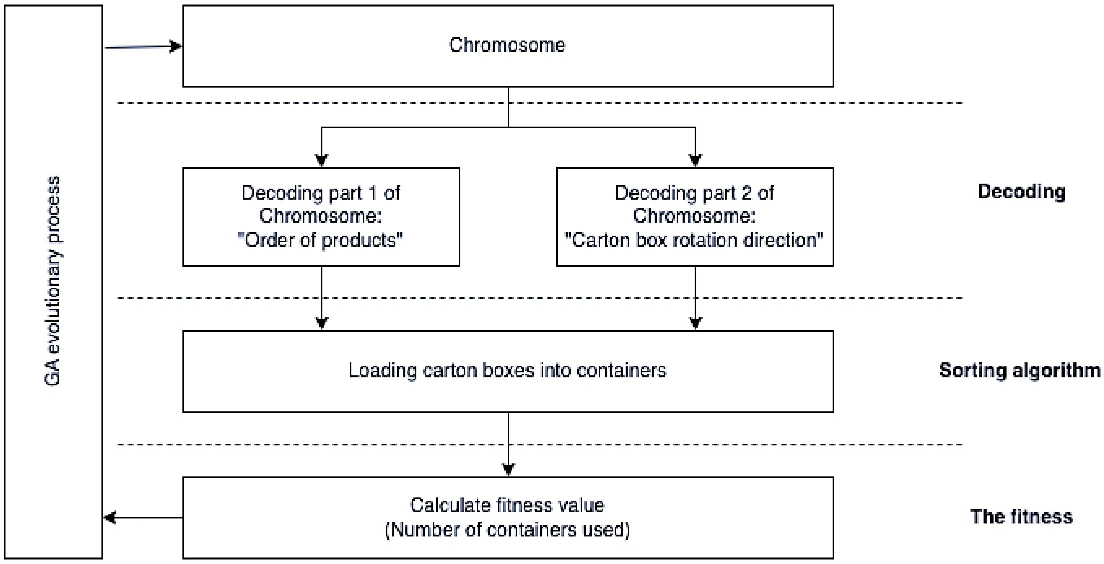


Figure 4. Structure of GA for solving problem

Considering the structure of the genetic algorithm (Figure 4), we have three main steps: decoding, packing algorithm, and fitness computation. In this genetic algorithm, a solution (also known as a "chromosome") is encoded as an array generated from 2n genes containing genetic information about "Item order" and "Box orientation." The first half of the array consists of n genes, representing the order in which items are packed into the container. Each gene in this part is a real number with a value from 0 to 1. The second half of the chromosome consists of n genes, each with a value from 1 to 6, indicating the orientation of the boxes. The actual dimensions (l'_i, w'_i, h'_i) of the items along the *x*, *y*, and *z* axes corresponding to the boxes' orientations have been explained above.

Before constructing any solution, the chromosome must be decoded into the packing order and orientation of the boxes so that the algorithm can convert them into packing positions of the items and compute the box sizes. The decoded genes are represented as two vectors: a vector of Box sequence, VBS, and a vector of Box orientation, VBO. This vector can be obtained by copying the second half of the encoded chromosome: $VBO_i = Gene_{n+i}$, $\forall i = 1...n$.

The wall-building packing algorithm is highly suitable for the packing needs of the textile and garment industries. With various types of purchase orders (PO), diverse designs, and colors, there is a high potential for confusion during the packing and unpacking process, leading to time and cost inefficiencies for businesses. The wall-building method can differentiate between POs using separate rows (layers), making management easier. Figure 5 shows the process of stacking cartons into containers.

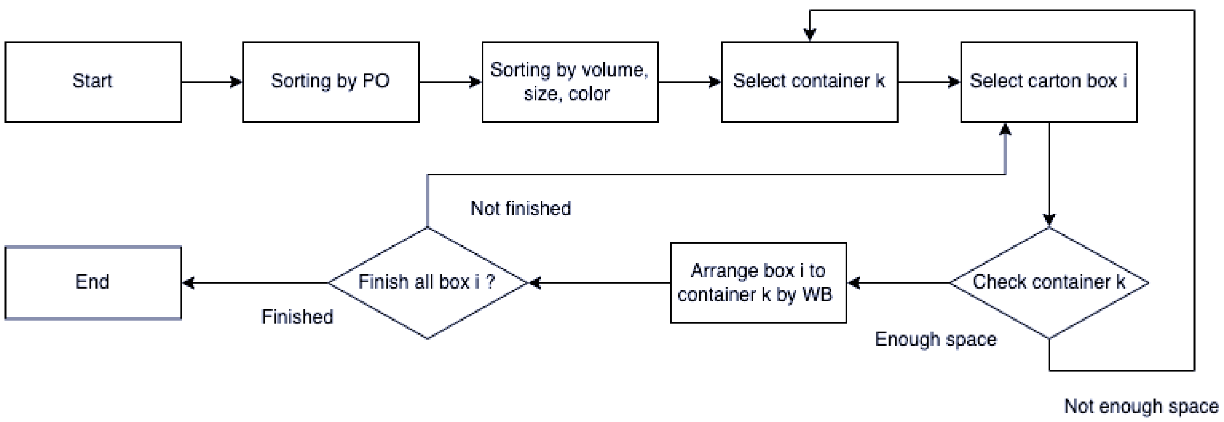


Figure 5. General stacking procedure

In the process of stacking cartons into containers using the wall-building method, the goods are arranged and selected based on the following criteria: Priority 1: Choose boxes with a larger base area (i.e., length and width); Priority 2: Choose boxes with a larger base area; Priority 3: Choose boxes with a larger length; Priority 4: Choose boxes with a larger width; Priority 5: Choose boxes with a larger height.

These criteria establish a hierarchical order for selecting boxes during the packing process. The algorithm prioritizes boxes with larger dimensions, or base areas, according to the specified hierarchy. These criteria are used to prioritize box selection; therefore, boxes with a larger base area, i.e., length and width, are selected first and packed in a lower position. Since this process stacks boxes in sequence, the boxes at the bottom should have a larger base area to ensure stability. Boxes can be rotated in different directions. In addition to the size criteria, due to the nature of textile and garment goods, priority is given to stacking products of the same color and size together. Then, suitable spaces for stacking cartons are considered by comparing the dimensions of the boxes with those of the empty spaces. The following criteria rank suitable spaces by examining their reference points, with higher priority given to spaces inside, as follows: Priority 1: Choose the space with the smallest x-coordinate of its reference point; Priority 2: Choose the space with the smallest y-coordinate of its reference point; Priority 3: Choose the space with the smallest z-coordinate of its reference point. The following process (Figure 6) will clarify how stacking is performed using the wall-building method:

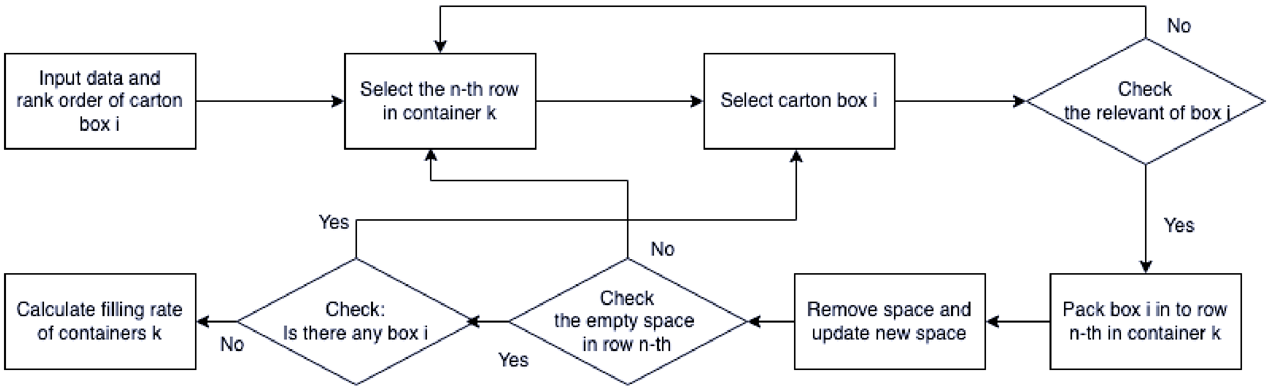


Figure 6. The stacking procedure follows the wall–building

4. Experiment and Results

To analyze and evaluate the 3D pin-packing model along with its variants as described in the mathematical model section, this study employs a comparative approach by comparing the results with the simulation cargo stacking software "Easy Cargo" [33] that Erbayrak et al. [34] used in the stacking container. The input data utilizes two different scenarios for comparison.

4.1. Scenario 1

Based on the algorithm published above, the research team collected actual data from a garment company. To ensure security, the quantity of goods will be marked separately. Below is a data table (Table 4) that has collected the packaging parameters of products that can be produced by the company. Table 4 displays the input data for the model, detailing specifications of multiple products, including size, color codes, and stacking prioritized by purchase order to meet the requirement of convenient unloading at the destination. The parameters of the orders to be stacked onto containers include information about the purchase order (PO), dimensions, and specifications. The objective is to select the container size (20 ft, 40 ft) to maximize the fill rate and minimize the number of containers used.

Table 4. Input data in scenario 1

Production code	Purchase order	Size	Color code	Specification				Quantity (boxes)
				Length (L) (cm)	Width (W) (cm)	High (H) (cm)	Weight (Kg)	
ABC1	PO01	S	C01	40	30	40	6	135
		M	C02	40	45	50	10	200
		L	C01	40	40	30	8	90
ABC2		XL	C01	40	45	50	10	160
ABC2	PO02	S	C01	40	30	40	5,5	105
		M	C01	40	30	40	6,5	300
		L	C02	40	40	30	7,5	270
ABC3	PO03	S	C01	40	30	40	4	120
		M	C01	40	30	40	5,5	150
ABC1		L	C02	40	45	50	5,5	80
		XL	C02	40	45	50	7	100
Total								**1710**

Below are the specifications of 20- and 40-feet containers commonly used in the process of transporting garments (Table 5).

Table 5. Container specification

Type of container	Specification	
20 feet	Length	6 m
	Width	2,4 m
	High	2,4 m
	Capacity	28280 kg
40 feet	Length	12m
	Width	2,4 m
	High	2,4 m
	Capacity	26750 kg

** Normalize the input data*

Due to the relatively large number of data inputs, the research team decided to normalize the data before running the algorithm, which will help reduce processing time and increase the quality of the solution. Previously, for items that could be defaulted, they were usually light items; therefore, the team would standardize carton types and the specific weight of each carton based on the β coefficient (calculated by Equation 20). The method employed in this study involves standardizing the data to automatically select the appropriate container size, either 20- or 40-feet.

$$\beta = \frac{The\ weight\ of\ carton\ i}{The\ volume\ of\ carton\ i} \tag{20}$$

Based on Equation 20, with *n* cartons *i* that need to be packed into a 20- or 40-feet container, we have:

$$\frac{\sum_{i=1}^{n} \beta_{carton\ i}}{n} < \beta_{40\ feet} < \beta_{20\ feet} \tag{21}$$

Therefore, when loading boxes into containers, we will pay attention to the volume of the boxes because the average coefficient $\beta_{carton\ i}$ compared to 20- and 40-feet containers is small. Furthermore, in order to expedite problem-solving, especially with large-scale problems, this study utilizes a method of converting from boxes to pallets when loading boxes with dimensions (l_i, w_i, h_i). For cartons with dimensions of 40×30×40 (cm), they will be stacked into one block consisting of 3×3×3 (27 boxes) on pallet 1, which has dimensions of 120×90×120 cm. Similarly, boxes measuring 40×45×50 (cm), stacked in a configuration of 3×3×2 (18 boxes), will be converted into pallet 2, with dimensions of 120×135×100 (cm). Additionally, boxes sized 40×40×30 cm, stacked in a configuration of 3×3×2 (18 boxes), will be converted into pallet 3, measuring 120×120×60 (cm). The conversion method is described in Table 6 as follows:

Table 6. Convert the number of boxes into pallets

Box dimension	Number of stacking boxes per pallet	Convert to the Pallet dimension	Pallet type
40×30×40 (cm)	3×3×3 = 27 boxes	120×90×120 (cm)	Pallet 1
40×45×50 (cm)	3×3×2 = 18 boxes	120×135×100 (cm)	Pallet 2
40×40×30 (cm)	3×3×2 = 18 boxes	120×120×60 (cm)	Pallet 3

From the data in Table 1, boxes with the same box size will be converted into pallets while still maintaining the order sequence according to each purchase order (PO). This process results in the data shown in Table 7. Below is a table (Table 7) that normalizes the number of cartons based on the original data:

Table 7. Standardize the number of boxes in scenario 1

Purchase order	Carton box code	Carton specifications			Quantity (box)	Number of boxes/pallet	Quantity of the standardized pallet
		Length (L) (cm)	Width (W) (cm)	High (H) (cm)			
PO01	B01	40	30	40	135	27	5 pallet 1
	B02	40	45	50	360	18	20 pallet 2
	B03	40	40	30	90	18	5 pallet 3
PO02	B01	40	30	40	405	27	15 pallet 1
	B03	40	40	30	270	18	15 pallet 3
PO03	B01	40	30	40	270	27	10 pallet 1
	B02	40	45	50	180	18	10 pallet 2
Total					**1710**		**80**

From the data in Table 5, the normalized data in Table 7 is obtained to reduce the problem size, thereby helping the program run faster. The principle of solving GA and wall-building problems is to stack the goods sequentially, employing a trial-and-error approach. With large-scale problem sizes like the input data, the program runs slowly. To expedite the solution process and ensure that similar types of boxes are stacked closely together according to the purchase order, boxes are converted into pallets for stacking onto the container. For instance, 1710 boxes are standardized to 80 pallets. This approach accelerates problem-solving and maintains the criteria of stacking similar items together and following the purchase order. Standardized pallet quantity in scenario 1 (Table 8).

Table 8. Standardized pallet quantity in scenario 1

The type of pallet	Pallet 1 (120×90×120) cm	Pallet 2 (120×135×100) cm	Pallet 3 (120×120×60) cm
Quantity	30	30	20

For garments, the carton is only allowed to rotate facing up. Table 9 is a matrix of feasible cases when rotating the carton, where 1 is feasible and 0 is not feasible.

Table 9. Matrix of rotation direction

Direction allowance	δ_{1i}	δ_{2i}	δ_{3i}	δ_{4i}	δ_{5i}	δ_{6i}
Relevant/ Irrelevant	1	0	0	0	1	0

4.2. Using GA to Solve Scenario 1

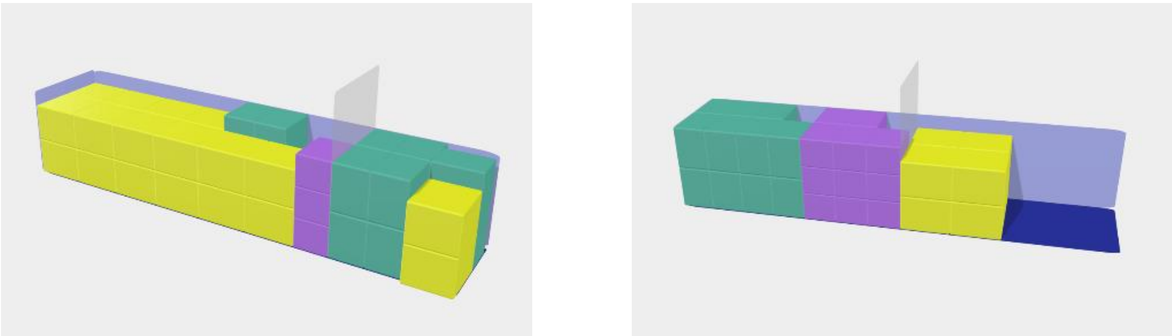
After solving the bin packing problem with the input data above, the result is to use two 40-feet containers to transport all the boxes of three POs. Each standard pallet will be represented as reference coordinates (x, y, z), and the method of stacking (rotation direction) is shown in Tables 10 and 11 and demonstrated in Figure 7:

Table 10. Container 1 results in scenario 1

Type of bin	x	y	z	Direction	Type of bin	x	y	z	Direction
B02 – PO01	0	0	0	δ_{5i}	B01 – PO01	63	12	12	δ_{5i}
B02 – PO01	0	12	0	δ_{5i}	B01 – PO01	72	12	0	δ_{5i}
B02 – PO01	0	0	10	δ_{5i}	B01 – PO01	67,5	0	0	δ_{5i}
B02 – PO01	0	12	10	δ_{5i}	B01 – PO01	67,5	0	10	δ_{5i}
B02 – PO01	13,5	0	0	δ_{5i}	B03 – PO01	81	0	0	δ_{5i}
B02 – PO01	13,5	12	0	δ_{5i}	B03 – PO01	81	12	0	δ_{5i}
B02 – PO01	13,5	0	10	δ_{5i}	B03 – PO01	81	0	8	δ_{5i}
B02 – PO01	13,5	12	10	δ_{5i}	B03 – PO01	81	12	8	δ_{5i}
B02 – PO01	27	0	0	δ_{5i}	B03 – PO01	81	0	16	δ_{5i}
B02 – PO01	27	12	0	δ_{5i}	B01 – PO03	90	0	0	δ_{5i}
B02 – PO01	27	0	10	δ_{5i}	B01 – PO03	90	12	0	δ_{5i}
B02 – PO01	27	12	10	δ_{5i}	B01 – PO03	90	0	12	δ_{5i}
B02 – PO01	40,5	0	0	δ_{5i}	B01 – PO03	90	12	12	δ_{5i}
B02 – PO01	40,5	12	0	δ_{5i}	B01 – PO03	99	0	0	δ_{5i}
B02 – PO01	40,5	0	10	δ_{5i}	B01 – PO03	99	12	0	δ_{5i}
B02 – PO01	40,5	12	10	δ_{5i}	B01 – PO03	99	0	12	δ_{5i}
B02 – PO01	54	0	0	δ_{5i}	B01 – PO03	99	12	12	δ_{5i}
B02 – PO01	54	12	10	δ_{5i}	B02 – PO03	108	0	0	δ_{1i}
B02 – PO01	54	12	0	δ_{5i}	B01 – PO03	108	13,5	0	δ_{1i}
B02 – PO01	54	12	12	δ_{5i}	B02 – PO03	108	0	10	δ_{1i}
B01 – PO01	63	12	0	δ_{5i}	B01 – PO03	108	13,5	12	δ_{1i}

Table 11. The container 2 results in scenario 1

Type of bin	x	y	z	Direction	Type of bin	x	y	z	Direction
B01 – PO02	0	0	0	δ_{5i}	B03 – PO02	36	0	16	δ_{5i}
B01 – PO02	0	12	0	δ_{5i}	B03 – PO02	36	12	16	δ_{5i}
B01 – PO02	0	0	12	δ_{5i}	B03 – PO02	45	0	0	δ_{5i}
B01 – PO02	0	12	12	δ_{5i}	B03 – PO02	45	12	0	δ_{5i}
B01 – PO02	9	0	0	δ_{5i}	B03 – PO02	45	0	8	δ_{5i}
B01 – PO02	9	12	0	δ_{5i}	B03 – PO02	45	12	8	δ_{5i}
B01 – PO02	9	0	12	δ_{5i}	B03 – PO02	45	0	16	δ_{5i}
B01 – PO02	9	12	12	δ_{5i}	B03 – PO02	45	12	16	δ_{5i}
B01 – PO02	18	0	0	δ_{5i}	B03 – PO02	54	0	0	δ_{5i}
B01 – PO02	18	12	0	δ_{5i}	B03 – PO02	54	0	8	δ_{5i}
B01 – PO02	18	0	12	δ_{5i}	B03 – PO02	54	0	16	δ_{5i}
B01 – PO02	18	12	12	δ_{5i}	B02 – PO03	63	0	0	δ_{5i}
B01 – PO02	27	0	0	δ_{5i}	B02 – PO03	63	12	0	δ_{5i}
B01 – PO02	27	12	0	δ_{5i}	B02 – PO03	63	0	10	δ_{5i}
B01 – PO02	27	0	12	δ_{5i}	B02 – PO03	63	12	10	δ_{5i}
B03 – PO02	36	0	0	δ_{5i}	B02 – PO03	76,5	0	0	δ_{5i}
B03 – PO02	36	12	0	δ_{5i}	B02 – PO03	76,5	12	0	δ_{5i}
B03 – PO02	36	0	8	δ_{5i}	B02 – PO03	76,5	0	10	δ_{5i}
B03 – PO02	36	12	8	δ_{5i}	B02 – PO03	76,5	12	10	δ_{5i}



(a) (b)

Figure 7. (a) Container 1 results in scenario 1; (b) Container 2 results in scenario 1

Based on the running results of the algorithm, the fill rate of each container *k* is calculated according to the following formula:

$$\Delta_k = \frac{\sum_{i=1}^{n} l_i w_i h_i}{L_k W_k H_k} \quad (22)$$

In this study, we conduct a comparative analysis of the fill rates obtained from the genetic algorithm (GA) model utilized herein and those acquired from Easy Cargo. This comparison aims to elucidate the feasibility and efficacy of our research. By juxtaposing the results from Easy Cargo with those of the GA model, we demonstrate the performance and validity of our approach.

Table 12. Comparison of research result and Easy cargo

Bin packing is solved by GA model of the study	Simulated by Easy cargo
The fill rate of container 1: 85,94 %	The fill rate of container 1: 83,125%
The fill rate of container 2: 65,63%	The fill rate of container 2: 68,43%
Average fill rate: 75,79%	Average fill rate: 75,78%

Table 12 provides a comparison between the results obtained from the bin packing problem solved by the genetic algorithm (GA) model used in the study and the results simulated by the Easy Cargo software. The fill rate of container 1 achieved through the GA model is 85.94%, whereas Easy Cargo achieves a fill rate of 83.125% for the same container. For container 2, the fill rate obtained through the GA model is 65.63%, whereas Easy Cargo achieves a slightly higher fill rate of 68.43%. The average fill rate across both containers is 75.79% when using the GA model, whereas Easy Cargo achieves a very similar average fill rate of 75.78%.

Overall, the results demonstrate that the GA model used in the study performs comparably with Easy Cargo in terms of average fill rate. However, there are slight differences in the fill rates for individual containers, with the GA model outperforming Easy Cargo for container 1 but underperforming for container 2. These differences may be attributed to the specific algorithms and optimization techniques employed by each method, and variations in the input parameters and constraints considered. When the amount of cargo stacked into containers is less than the total capacity of the container, the average fill rate may not clearly reflect the results of this research when solved using GA. However, the algorithm used in this study will address the stacking problem with constraints related to purchase orders, whereas Easy Cargo currently cannot meet this requirement.

4.3. Scenario 2

The input data in Scenario 2 are presented in Table 13.

Table 13. Input data in scenario 2

Production code	Purchase order	Size	Color code	Specification				Quantity (boxes)
				Length (L) (cm)	Width (W) (cm)	High (H) (cm)	Weight (Kg)	
ABC1	PO01	S	C01	30	40	40	6	216
		M	C02	50	30	30	8	320
		L	C01	50	40	40	10	135
ABC2		M	C01	50	30	30	10	160
ABC2	PO02	S	C01	30	40	40	6,5	216
		M	C01	50	30	30	11	160
		L	C02	50	40	40	7,5	270
ABC3	PO03	S	C01	30	40	40	4	108
		M	C01	50	30	30	5,5	160
ABC1		L	C02	50	30	30	5,5	160
		XL	C02	50	40	40	7	270
Total								**2175**

For cartons with dimensions of 50×40×40 cm, they will be stacked into one block consisting of 3×3×3 (27 boxes) on pallet 1, which has dimensions of 150×120×120 cm. Similarly, boxes measuring 50×30×30 (cm), stacked in a configuration of 2×4×4 (32 boxes), will be converted into pallet 2, with dimensions of 100×120×120 (cm). Additionally, boxes sized 30×40×30 cm, stacked in a configuration of 3×3×2 (18 boxes), will be converted into pallet 3, measuring 90×120×60 (cm). The conversion method is described in Table 14 as follows:

Table 14. Convert the number of boxes into pallets

Box dimension	Numbers of stacking box per pallet	Convert to Pallet dimension
50×40×40 (cm)	3×3×3 = 27 boxes	150×120×120 (cm)
50×30×30 (cm)	2×4×4 = 32 boxes	100×120×120 (cm)
30×40×30 (cm)	3×3×2 = 18 boxes	90×120×60 (cm)

Table 15 normalizes the number of cartons based on the original data table:

Table 15. Standardize the number of boxes in scenario 2

Purchase order	Carton box code	Carton specifications			Quantity (box)	Number of boxes per pallet	Quantity of the standardized pallet
		Length (L) (cm)	Width (W) (cm)	High (H) (cm)			
PO01	B01	50	30	30	135	27	5 pallet 1
	B02	50	40	40	480	32	15 pallet 2
	B03	30	40	40	216	18	12 pallet 3
PO02	B01	50	30	30	270	27	10 pallet 1
	B02	50	40	40	160	32	5 pallet 2
	B03	30	40	40	216	18	12 pallet 3
PO03	B01	50	30	30	270	27	10 pallet 1
	B02	50	40	40	320	32	10 pallet 2
	B03	30	40	40	108	18	6 pallet 3
Total					**2175**		**85**

The standardized pallet quantity is presented in Table 16:

Table 16. Standardized pallet quantity in scenario 2

Type of pallet	Pallet 1 (100×120×120) cm	Pallet 2 (150×120×120) cm	Pallet 3 (90×120×80) cm	Total
Number	25	30	30	85

4.4. Using GA to Solve Scenario 2

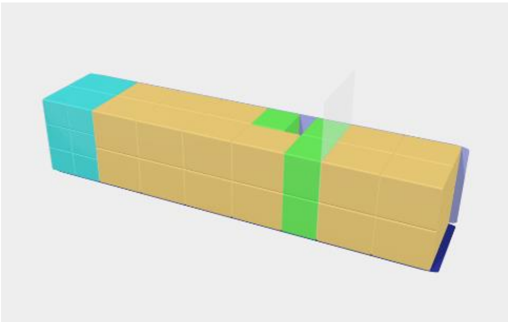
After solving the bin packing problem with the input data above, the result is to use two 40-feet containers to transport all the boxes of three POs. Each standard pallet will be represented as reference coordinates (x, y, z), and the method of stacking (rotation direction) is shown in Tables 17 and 18 and Figure 8:

Table 17. Container 1 results in scenario 2

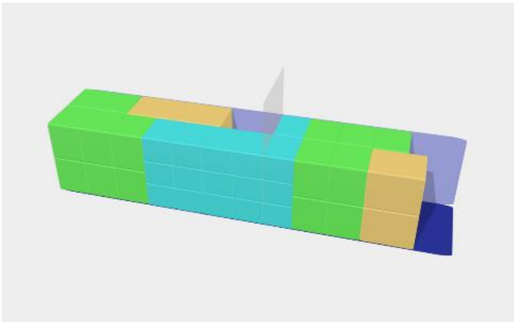
Type of bin	x	y	z	Direction	Type of bin	x	y	z	Direction
B03 – PO1	0	0	0	δ_{1i}	B02 – PO1	48	0	0	δ_{1i}
B03 – PO1	0	12	0	δ_{1i}	B02 – PO1	48	12	0	δ_{1i}
B03 – PO1	0	0	8	δ_{1i}	B02 – PO1	48	0	12	δ_{1i}
B03 – PO1	0	12	8	δ_{1i}	B02 – PO1	48	12	12	δ_{1i}
B03 – PO1	0	0	16	δ_{1i}	B02–PO1	63	0	0	δ_{1i}
B03 – PO1	0	12	16	δ_{1i}	B02–PO1	63	12	0	δ_{1i}
B03 – PO1	9	0	0	δ_{1i}	B02–PO1	63	0	12	δ_{1i}
B03 – PO1	9	12	0	δ_{1i}	B01–PO1	63	12	12	δ_{1i}
B03 – PO1	9	0	8	δ_{1i}	B01–PO1	78	0	0	δ_{1i}
B03 – PO1	9	12	8	δ_{1i}	B01–PO1	78	12	0	δ_{1i}
B03 – PO1	9	0	16	δ_{1i}	B01–PO1	73	12	12	δ_{1i}
B03 – PO1	9	12	16	δ_{1i}	B01 – PO1	78	0	12	δ_{1i}
B02 – PO1	18	0	0	δ_{1i}	B02 – PO03	88	0	0	δ_{1i}
B02 – PO1	18	12	0	δ_{1i}	B02 – PO03	88	12	0	δ_{1i}
B02 – PO1	18	0	12	δ_{1i}	B02 – PO03	88	0	12	δ_{1i}
B02 – PO1	18	12	12	δ_{1i}	B02 – PO03	88	12	12	δ_{1i}
B02 – PO1	33	0	0	δ_{1i}	B02 – PO03	103	0	0	δ_{1i}
B02 – PO1	33	12	0	δ_{1i}	B02 – PO03	103	12	0	δ_{1i}
B02 – PO1	33	0	12	δ_{1i}	B02 – PO03	103	0	12	δ_{1i}
B02 – PO1	33	12	12	δ_{1i}	B02 – PO03	103	12	12	δ_{1i}

Table 18. Container 2 results in scenario 2

Type of bin	x	y	z	Direction	Type of bin	x	y	z	Direction
B01 – PO02	0	0	0	δ_{1i}	B02 – PO02	50	12	0	δ_{1i}
B01 – PO02	0	12	0	δ_{1i}	B03 – PO02	57	0	0	δ_{1i}
B01 – PO02	0	0	12	δ_{1i}	B03 – PO02	57	0	8	δ_{1i}
B01 – PO02	0	12	12	δ_{1i}	B03 – PO02	57	0	16	δ_{1i}
B01 – PO02	10	0	0	δ_{1i}	B03 – PO03	66	0	0	δ_{1i}
B01 – PO02	10	12	0	δ_{1i}	B03 – PO03	66	12	0	δ_{1i}
B01 – PO02	10	0	12	δ_{1i}	B03 – PO03	66	0	8	δ_{1i}
B01 – PO02	10	12	12	δ_{1i}	B03 – PO03	66	12	8	δ_{1i}
B01 – PO02	20	0	0	δ_{1i}	B03 – PO03	66	0	16	δ_{1i}
B02 – PO02	20	12	0	δ_{1i}	B03 – PO03	66	12	16	δ_{1i}
B01 – PO02	20	0	12	δ_{1i}	B01 – PO03	75	0	0	δ_{1i}
B02 – PO02	20	12	12	δ_{1i}	B01 – PO03	75	12	0	δ_{1i}
B03 – PO02	30	0	0	δ_{1i}	B01 – PO03	75	0	12	δ_{1i}
B03 – PO02	30	0	8	δ_{1i}	B01 – PO03	75	12	12	δ_{1i}
B03 – PO02	30	0	16	δ_{1i}	B01 – PO03	94	0	0	δ_{1i}
B02 – PO02	35	12	0	δ_{1i}	B01 – PO03	94	12	0	δ_{1i}
B02 – PO02	35	12	12	δ_{1i}	B01 – PO03	94	0	12	δ_{1i}
B03 – PO02	39	0	0	δ_{1i}	B01 – PO03	94	12	12	δ_{1i}
B03 – PO02	39	0	8	δ_{1i}	B02 – PO03	104	0	0	δ_{1i}
B03 – PO02	39	0	16	δ_{1i}	B02 – PO03	104	12	0	δ_{1i}
B03 – PO02	48	0	0	δ_{1i}	B01 – PO03	104	0	12	δ_{1i}
B03 – PO02	48	0	8	δ_{1i}	B01 – PO03	104	12	12	δ_{1i}
B03 – PO02	48	0	16	δ_{1i}					



(a)



(b)

Figure 8. (a) Container 1 results in scenario 2; (b) Container 2 results in scenario 2

Table 19. Comparison of research results and Easy cargo for scenario 2

Bin packing is solved by the GA model of the study	Simulated by Easy cargo
The fill rate of container 1 (40 feet): 97,3%	The fill rate of container 1 (40 feet): 72,29%
The fill rate of container 2 (40 feet): 86%	The fill rate of container 2 (40 feet): 96,45%
	The fill rate of container 3 (20 feet): 37,5%
Average fill rate: 91,67%	**Average fill rate: 68,74%**

From Table 19, the average fill rate across all containers when using the GA model is significantly higher than that achieved by Easy Cargo. These results suggest that the GA model used in the study outperforms Easy Cargo in terms of the overall container fill rate, with particularly notable differences observed in the fill rates of containers 1 and 2. In Scenario 2, the solution provided by Easy Cargo requires the use of two 40-ft containers and one 20-ft container, with an average fill rate of 68.74%. In contrast, the optimal solution from the algorithm requires only two 40-ft containers, achieving an average fill rate of 91.67%. This highlights the superiority of the proposed algorithm in providing accurate solutions. Through comparison with the Easy Cargo stacking method, it can be observed that as the quantity of goods stacked into the container approaches the capacity of the container, the results of this research demonstrate a significant improvement in the fill rate and the number of containers used.

Through the two scenarios above, it can be observed that with different box data, the algorithm can help determine the optimal arrangement to achieve the highest container packing ratio. This method can assist the export department in calculating the minimum number of containers needed to transport goods to international destinations. When compared to Easy Cargo, the algorithm developed by the research team automatically selects containers without the need for manual selection of each container. When stacking goods using the method of categorizing merchandise, the research algorithm can stack each purchase order separately, with separators if needed. However, the Easy Cargo software still stacks products from different purchase orders together.

5. Conclusion

The study analyzed the challenges encountered within shortage containers recently, which lead to high transportation costs for export and import volumes. Optimizing the bin packing problem for the garment and textile industry is a significant issue to maximize container capacity. In a complex problem like bin packing, which involves multiple constraints such as various box sizes with weight and container size restrictions, this study introduces the additional constraint of prioritizing stacking according to purchase orders (PO), which is a practical consideration in the garment and textile industry to meet the requirement for convenient unloading. In addition to synthesizing fundamental theories and variations of the bin packing problem, this study proposes the use of approximate algorithms. This study demonstrated the effectiveness of applying genetic algorithms and wall-building algorithms to address the complex practical constraints of the textile industry in solving the bin packing problem. Genetic algorithms were found to exhibit optimal performance in solving large-scale optimization problems within stipulated timeframes, yielding high-quality results. Furthermore, the research developed a loading method tailored to meet the specific requirements of the textile industry, accommodating various types of orders, diverse designs, and multiple colors and sizes.

This research has demonstrated that employing optimal construction methods and using approximate algorithms can effectively address complex problems such as the bin packing problem, which involves multiple box sizes, various container sizes, diverse product categories, and different purchase orders. What sets this study apart from previous research is the incorporation of additional constraints related to purchase orders. These constraints are commonly encountered in real-world scenarios within the garment exporting industry, thereby enhancing the problem's complexity compared to previous studies.

The outcomes of this research indicate that the proposed approach can optimize the container fill rate up to 91.67%, significantly surpassing the fill rate achieved using the Easy Cargo stacking method. Presently, forwarder companies utilize EasyCargo3D software for cargo stacking checks. However, this software lacks functionality for sorting by purchase order, calculating fill rates, checking by container type, or providing solutions for stacking multiple containers simultaneously or automatically selecting container sizes from the available container pool. This indicates that the algorithm can efficiently optimize the number of containers required to the lowest level, thereby minimizing transportation costs in situations of empty container shortages and high international shipping costs. In practice, manually arranging boxes into containers is not optimized and is often time-consuming. Therefore, with the model and solving method proposed in this study, the task of packing cargo into containers will be optimized, thereby reducing the time required for arranging.

Despite its positive contributions, the research has certain limitations, such as addressing less than container load (LCL) scenarios where boxes are stacked with other types of goods, introducing additional complexities related to separating different types of goods during stacking. For future research endeavors, integrating the bin-packing problem with the vehicle routing problem could optimize the transportation process throughout the entire supply chain, from its inception to its culmination. Proposal to combine the bin packing problem with the vehicle routing problem to optimize the transportation process from the beginning to the end of the supply chain.

6. Declarations

6.1. Data Availability Statement

The data presented in this study are available in the article.

6.2. Funding

This research was funded by the Hanoi University of Science and Technology (HUST) under project number T2022-PC-078.

6.3. Institutional Review Board Statement

Not applicable.

6.4. Informed Consent Statement

Not applicable.

6.5. Declaration of Competing Interest

The author declares that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7. References

[1] Kuehne-Nagel. (2021). Why does the current container shortage happen? Kuehne + Nagel International AG, Feusisberg, Switzerland. Available online: https://home.kuehne-nagel.com/-/knowledge/market-insights/container-shortage (accessed on March 2023).

[2] Freightos. (2023). Shipping & Freight Cost Increases, Current Shipping Issues, and Shipping Container Shortage. Freightos, Ramallah, Israel. Available online: https://www.freightos.com/freight-blog/shipping-delays-and-cost-increases/ (accessed on March 2023).

[3] Kunz, G. I., Karpova, E., & Garner, M. B. (2021). Textile and Apparel Trade: Barriers, Regulations, and Politics. (2021). Going Global, 107–132. doi:10.5040/9781501338700.ch-005.

[4] WTO. (2023). World trade statistical review 2023. World Trade Organization, Geneva, Switzerland. Available online: https://www.wto.org/english/res_e/publications_e/wtsr_2023_e.htm (accessed on March 2023).

[5] Zhu, W., Chen, S., Dai, M., & Tao, J. (2024). Solving a 3D bin packing problem with stacking constraints. Computers and Industrial Engineering, 188, 109814. doi:10.1016/j.cie.2023.109814.

[6] Cai, S., Deng, J., Lee, L. H., Chew, E. P., & Li, H. (2023). Heuristics for the two-dimensional irregular bin packing problem with limited rotations. Computers and Operations Research, 160, 106398. doi:10.1016/j.cor.2023.106398.

[7] Deplano, I., Lersteau, C., & Nguyen, T. T. (2021). A mixed-integer linear model for the multiple heterogeneous knapsack problem with realistic container loading constraints and bins' priority. International Transactions in Operational Research, 28(6), 3244–3275. doi:10.1111/itor.12740.

[8] Olsson, J., Larsson, T., & Quttineh, N. H. (2020). Automating the planning of container loading for Atlas Copco: Coping with real-life stacking and stability constraints. European Journal of Operational Research, 280(3), 1018–1034. doi:10.1016/j.ejor.2019.07.057.

[9] Cid-Garcia, N. M., & Rios-Solis, Y. A. (2020). Positions and covering: A two-stage methodology to obtain optimal solutions for the 2d-bin packing problem. PLoS ONE, 15(4), 229358. doi:10.1371/journal.pone.0229358.

[10] Huan, P. T., Canh, H. T., Thai, V. D., & Tuan, B. N. 2D bin packing problem and application in maritime transportation. TNU Journal of Science and Technology, 226(07), 226–234.

[11] Kundu, O., Dutta, S., & Kumar, S. (2019). Deep-Pack: A Vision-Based 2D Online Bin Packing Algorithm with Deep Reinforcement Learning. 2019 28th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2019, 1–7. doi:10.1109/RO-MAN46459.2019.8956393.

[12] Ramos, A. G., Silva, E., & Oliveira, J. F. (2018). A new load balance methodology for container loading problem in road transportation. European Journal of Operational Research, 266(3), 1140–1152. doi:10.1016/j.ejor.2017.10.050.

[13] Gzara, F., Elhedhli, S., & Yildiz, B. C. (2020). The Pallet Loading Problem: Three-dimensional bin packing with practical constraints. European Journal of Operational Research, 287(3), 1062–1074. doi:10.1016/j.ejor.2020.04.053.

[14] Lemechshenko, O., Nakipova, G., & Akhmet, G. (2022). Improving the Program-Targeted Management Methodology and Its Practical Application for the Sustained and Environment Development of Agro-Industrial Complex. Journal of Environmental Management and Tourism, 13(3), 769. doi:10.14505/jemt.v13.3(59).16.

[15] Nishiyama, S., Lee, C., & Mashita, T. (2020). Designing a flexible evaluation of container loading using physics simulation. In Communications in Computer and Information Science: Vol. 1173 CCIS, Springer International Publishing, 255–268. doi:10.1007/978-3-030-41913-4_21.

[16] Paquay, C., Schyns, M., & Limbourg, S. (2016). A mixed integer programming formulation for the three-dimensional bin packing problem deriving from an air cargo application. International Transactions in Operational Research, 23(1–2), 187–213. doi:10.1111/itor.12111.

[17] Sheng, L., Xiuqin, S., Changjian, C., Hongxia, Z., Dayong, S., & Feiyue, W. (2017). Heuristic algorithm for the container loading problem with multiple constraints. Computers and Industrial Engineering, 108, 149–164. doi:10.1016/j.cie.2017.04.021.

[18] Kurpel, D. V., Scarpin, C. T., Pécora Junior, J. E., Schenekemberg, C. M., & Coelho, L. C. (2020). The exact solutions of several types of container loading problems. European Journal of Operational Research, 284(1), 87–107. doi:10.1016/j.ejor.2019.12.012.

[19] Mahvash, B., Awasthi, A., & Chauhan, S. (2018). A column generation-based heuristic for the three-dimensional bin packing problem with rotation. Journal of the Operational Research Society, 69(1), 78–90. doi:10.1057/s41274-017-0186-7.

[20] Sharma, E. (2021). Harmonic Algorithms for Packing d-Dimensional Cuboids into Bins. Leibniz International Proceedings in Informatics, LIPIcs, 213, 1–22. doi:10.4230/LIPIcs.FSTTCS.2021.32.

[21] Oliveira, L. de A., de Lima, V. L., de Queiroz, T. A., & Miyazawa, F. K. (2020). The container loading problem with cargo stability: a study on support factors, mechanical equilibrium and grids. Engineering Optimization, 53(7), 1–20. doi:10.1080/0305215X.2020.1779250.

[22] Zhou, Q., & Liu, X. (2017). A swarm optimization algorithm for practical container loading problem. Proceedings IECON 43rd Annual Conference of the IEEE Industrial Electronics Society, 2017-January, 5690–5695. doi:10.1109/IECON.2017.8216987.

[23] Moura, A., & Bortfeldt, A. (2017). A two-stage packing problem procedure. International Transactions in Operational Research, 24(1–2), 43–58. doi:10.1111/itor.12251.

[24] Gajda, M., Trivella, A., Mansini, R., & Pisinger, D. (2021). An Optimization Approach for a Complex Real-Life Container Loading Problem. SSRN Electronic Journal, 1-34. doi:10.2139/ssrn.3740046.

[25] Otelbaev, M., Durmagambetov, A. A., & Seitkulov, Y. N. (2008). Conditions for the existence of a global strong solution to a class of nonlinear evolution equations in a Hilbert space. Proceedings of the Steklov Institute of Mathematics, 260(1), 194–203. doi:10.1134/S0081543808010148.

[26] Ocloo, V., & Fügenschuh, A. (2020). A New Mathematical Model for a 3D Container Packing Problem on Modelling of 3D-Container Packing Problem. Brandenburg University of Technology, Cottbus, Germany. doi:10.26127/BTUOpen-5088.

[27] Nascimento, O. X. do, Alves de Queiroz, T., & Junqueira, L. (2021). Practical constraints in the container loading problem: Comprehensive formulations and exact algorithm. Computers and Operations Research, 128, 105186. doi:10.1016/j.cor.2020.105186.

[28] Tole, K., Moqa, R., Zheng, J., & He, K. (2023). A Simulated Annealing approach for the Circle Bin Packing Problem with Rectangular Items. Computers and Industrial Engineering, 176, 109004. doi:10.1016/j.cie.2023.109004.

[29] Xiang, X., Yu, C., Xu, H., & Zhu, S. X. (2018). Optimization of heterogeneous container loading problem with adaptive genetic algorithm. Complexity, 2018, 1–12. doi:10.1155/2018/2024184.

[30] Tsao, Y. C., Tai, J. Y., Vu, T. L., & Chen, T. H. (2024). Multiple bin-size bin packing problem considering incompatible product categories. Expert Systems with Applications, 247, 123340. doi:10.1016/j.eswa.2024.123340.

[31] Ananno, A. A., & Ribeiro, L. (2024). A Multi-Heuristic Algorithm for Multi-Container 3-D Bin Packing Problem Optimization Using Real World Constraints. IEEE Access, 12, 42105–42130. doi:10.1109/ACCESS.2024.3378063.

[32] Yang, Y., Wu, Z., Hao, X., Liu, H., & Qi, M. (2023). Two-layer heuristic for the three-dimensional bin design and packing problem. Engineering Optimization, 1–38. doi:10.1080/0305215X.2023.2269868.

[33] EasyCargo. (2021). Container and truck loading software EasyCargo. EasyCargo, Liben, Czech Republic. Available online: https://www.easycargo3d.com/ (accessed on March 2024).

[34] Erbayrak, S., Özkır, V., & Mahir Yıldırım, U. (2021). Multi-objective 3D bin packing problem with load balance and product family concerns. Computers and Industrial Engineering, 159, 107518. doi:10.1016/j.cie.2021.107518.