ON TWO-DOOR THREE-DIMENSIONAL CONTAINER PACKING PROBLEM UNDER HOME DELIVERY SERVICE

Chun-Cheng Lin*
Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan

Jia-Rong Kang
Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan

Wan-Yu Liu
Department of Forestry, National Chung Hsing University, Taichung 402, Taiwan

Cheng-Chi Li
Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan

ABSTRACT

This paper considers the three-dimensional container packing problem (3DCPP) under home delivery service, which predetermines an ordering of delivering ordered cargo boxes to customers, and aims to load a subset of those boxes according to the delivery ordering into a 3D car container with maximal utilization ratio of the container space while prohibiting the process of delivering cargos to each customer from unloading and subsequent reloading of boxes. In practice, cargos can be taken out from not only the rear door of the delivery container but also the side door. Nonetheless, the previous research only considered the case with only a single door. Therefore, this paper establishes the model for the two-door 3DCPP under home delivery service. Furthermore, the rule of invisible and untouchable items and the subvolume-based approach for the one-door problem are extended to resolve the two-door problem. Last, simulation analysis verifies performance of our approach.

Keywords: Home delivery, container packing problem, transportation, logistics

1. INTRODUCTION

With development of the Internet, advance in living standard, as well as trend of population aging, living habits have been changing, e.g., more and more customers tend to subscribe commodities via e-grocery [1]. Number of e-grocery companies has also grown rapidly with a growth rate of 42% each year since 2003 [2]. The volume of annual sales has also increased speedily, e.g., it is predicted that e-grocery sales in United States would increase to 250 billion dollars by 2014 [3]. To become a successful e-grocery company, a fast and convenient cargo delivery service is one of the key factors. If only the convenience of the fore-end behavior on picking and subscribing orders is concerned, but the convenience and limitation rules on the rear-end last-mile cargo delivery in logistics are not matched up, it is hard to find the niche in the e-grocery market.

Home delivery service is commonly used for the last-mile delivery to customers in e-grocery companies. With increasing development of the e-commerce industry, home delivery service has also been developed as one of the most important business models [4], and is usually a key factor of whether an e-grocery company can succeed [5]. The factors on operations costs in home delivery service include range of the provided commodities, the cargo types that can be received, time window of customer availability, efficiency of handling orders, number of customers, delivery time, and so on [6]. Therefore, for e-grocery companies, it has become a large challenge to

* Corresponding author: cclin321@nctu.edu.tw

effectively pursue development of the home delivery system.

This paper investigates a type of three-dimensional container packing problem (3DCPP), called *two-door 3DCPP under home delivery*. The 3DCPP is concerned with how to pack a number of rectangular items (boxes) orthogonally to a three-dimensional rectangular container, so that utilization of the container is maximized, or the total value of the packed items is as large as possible. The 3DCPP has been one of the most important issues in logistics management. A large number of factors and constraints are required to be concerned, e.g., the cargo to be loaded (including constraints of cargo size, orientation of loading a cargo, loading location, and degree of heterogeneity among cargos to be loaded) and container (size and type, maximal weight to be loaded, and distribution of cargo weights in the container). Hence, a variety of types of 3DCCP existed.

The 3DCPP has been shown to be NP-hard [7]. The work in [8] gave a complete introduction to various types of 3DCPP and their different constraints. In most of the previous works, various modelling methods, exact or approximation algorithms, and metaheuristic algorithms are used to solve different types of 3DCPP [9]. Among them, metaheuristic algorithms are the most popular, e.g., tabu search [9], simulated annealing [10], genetic algorithm [11] were used to cope with various types of 3DCPP. Recently, hybrid metaheuristic algorithms has been developed to solve the 3DCPP incorporated with the vehicle routing problem, e.g., [12].

Our previous work in [13] considered the 3DCPP under home delivery service, in which a consignment routing is assumed to be given (i.e., the delivery ordering of cargos (items) is fixed), and hence, the items should be packed into the container according to the delivery ordering so that utilization of the container is maximized, while the cost of unloading items at each customer is zero. Note that when the delivery car arrives at a customer, the *cost of unloading items* is defined as the number of the items (which are not the items required by the customer) to be unloaded and reloaded. In practice, if the cost of unloading items is not zero, i.e., some items are needed to be uploaded and reloaded before unloading the required item, the deliveryperson are troubled to unload and reload the unrequired items many times. Hence, the "zero unloading cost" should be regarded as a constraint in the 3DCPP under home delivery service [13]. Our previous work in [13] developed a rule of *invisible and untouchable items* to judge whether an item leads to the unloading cost, and then proposed a subvolume-based algorithm for the 3DCPP under home delivery service.

Recently, most large logistics companies, such as FedEx and DHL, have applied two-door containers (i.e., with a rear door and a side (right) door) in their home delivery services, as illustrated in Figure 1. However, most of the previous academic works for 3DCPP (e.g., [13]) considered to pack items in the container with only the rear door, which does not meet the practical situation. Therefore, this work further extends our previous work in [13] with the two-door container type. Note that size of the right door is generally greater than that of the rear door, as shown in Figure 1. From the one-door to the two-door case, this paper is required to make lots of more complex extensions, including novel design of some main components of the subvolume-based algorithm and the unloading cost calculated according to the rule of invisible and untouchable items.

Figure 1. A cargo container with two doors.

The main contributions of this paper is given as follows:

- This paper investigates the two-door 3DCPP, which was never studied in previous academic works. This new problem is able to better reflect the realities of loading and delivery drop-off processes and to effectively raise the cargo space utilization of car container for home delivery services.
- This paper proposes a subvolume-based approach for the two-door 3DCPP, in which more subvolumes are generated to load more cargo items. While the zero unloading cost constraint is satisfied, utilization of the container is as large as possible, to meet the real-world behavior on home delivery service.

The rest of this paper is organized as follows. Section 2 reviews the related works and briefly introduces the subvolume-based approach. Section 3 proposes our subvolume-based approach to the two-door 3DCPP under home delivery service. Section 4 implements the proposed approach and conducts various experimental results. Section 5 is the conclusion.

2. PRELIMINARIES

This section reviews the related work on the problem concerned in this paper, and then introduces the subvolume-based approach.

2.1 Related works

Previous works developed various modelling methods, exact or approximation algorithms, and metaheuristic algorithms for different types of 3DCPP. As for the modelling methods, Junqueira et al. [14] applied a mathematical programming method to solve the vehicle routing problem with real-world three-dimension loading constraints. Martello et al. [15] established a branch and bound algorithm for the 3DCPP. Padberg [16] applied a mathematical programming method to model the 3DCPP, and resolved it by the CPLEX solver. Junqueira et al. [17] considered a lot of constraints for 3DCPP, e.g., multiple drop-off condition, container loading constraint, and so on, and then proposed a mixed-integer programming model.

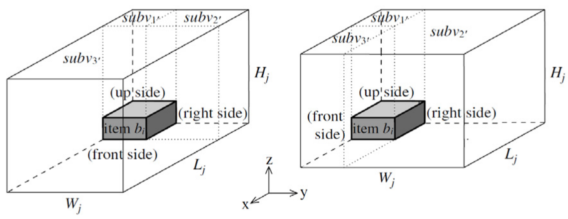
Most conventional heuristic algorithms are based on different packing approaches to pack cargo items. Liu et al. [18] applied a binary tree search heuristic algorithm for the three dimensional constrainer loading problem to maximize the packing volume and satisfy three constraints about support, orientation and guillotine cutting. Hifi et al. [19] presented novel search strategies for the three dimensional sphere packing problem to minimize length of the container having fixed width and height. Pisinger [20] applied a *wall-building* approach, which slices the container into vertical walls, and then packs items into each wall from the front side to the rear side of the container, based on a ranking rule. Different from the above two approaches, Abdou and Elmasry [21] proposed a *subvolume-based* approach, which recursively packs an item into a subvolume and then divides the remaining volume of the subvolume into three smaller subvolumes along the three sides of the packed item. Hifi et al. [22] proposed a hybrid greedy heuristic combined with selection and positioning strategies to deal with the three-dimensional single bin-size bin packing problem. Huang and He [23] proposed a *caving degree* approach, in which each item to be packed is packed at one of the corners of the remaining space according to the caving degree of the corner, which reflects the degree of whether item stacks are connected with each other. Xiuli et al. [24] proposed a container packing approach that incorporates *wall-building* and *caving degree*, to first slice the container into multiple walls, and then select the items to be packed into those walls according to *caving degree*.

Metaheuristic algorithms are suitable for solving large-scale 3DCPPs, e.g., tabu search [9], genetic algorithm (GA) [11], simulated annealing [10], and greedy randomized adaptive search procedure (GRASP) [25], [26]. Recently, hybrid approaches of various metaheuristic algorithms have been proposed to solve 3DCPPs, e.g., Padberg [16] proposed an approach that incorporates GA and ant colony optimization.

2.2 Subvolume-based approach

From the previous literature, the 3DCPPs with loading stability constraint and multi-drop situations were usually solved by subvolume-based approaches. As the 3DCPP under home delivery service also considers the two constraints, this paper also applies the subvolume-based approach to solve the concerned problem as follows. Initially, the rectangular container is regarded as a three-dimensional rectangular subvolume. Consider to iteratively place an item into the subvolume, and then the subvolume is divided into three smaller subvolumes. Repeat the same procedure until there is no subvolume that can accommodates the concerned item.

A subvolume j with length L_j, width W_j, and height H_j can be illustrated in Figure 2, and is denoted as $subv_j(x_j, y_j, z_j, L_j, W_j, H_j)$, in which (x_j, y_j, z_j) is the coordinate of the left-bottom-back corner of the subvolume; (L_j, W_j, H_j) denotes the length, width, and height of the subvolume, respectively. After an item is packed into subvolume j, three smaller subvolumes $subv_{1'}$, $subv_{2'}$, and $subv_{3'}$ are generated from the up, front, and right sides of the packed item, respectively, as shown in Figure 2. Next, one of the available subvolumes is chosen to pack the next item. By doing so, a lot of smaller subvolumes are produced, and hence, a list *SubV* is used to record those subvolumes. Hence, at each iteration of the algorithm to pack an item, the list *SubV* of subvolumes is required to be updated.



(a) (b)

Figure 2. Illustration of subvolumes (a) $L_j > W_j$ and (b) $L_j < W_j$.

When three smaller subvolumes are produced, the longer one between L_j and W_j is always sliced to generate larger subvolumes so that larger-size items are able to be packed, as illustrated in Figures 2(a) and 2(b). Additionally, when the three new subvolumes are added to list *SubV*, two actions are executed on all the subvolumes in list *SubV* as follows. The first action is to *combine* two adjacent subvolumes that have the same width or length into a larger subvolume so that later larger-size items could be packed, as illustrated in Figure 3. The second action is to *repartition* two adjacent subvolumes with the same height into two new subvolumes with different sizes, as illustrated in Figure 4.

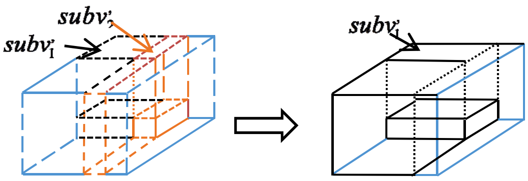


Figure 3. Illustration of the combination operator.

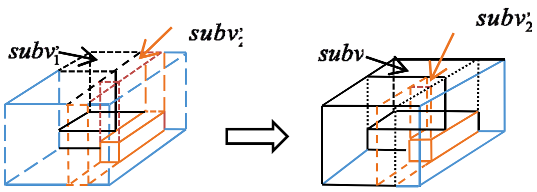


Figure 4. Illustration of the repartitioning operator.

3. OUR SUBVOLUME-BASED APPROACH TO THE TWO-DOOR 3DCPP UNDER HOME DELIVERY SERVICE

This section first gives the problem setting, and then defines the unloading cost precisely. Finally, our approach for the concerned problem is explained.

3.1 Problem setting

This subsection gives the problem settings of the two-door 3DCPP under home delivery service. Note that some of the settings for the two-door case are the same with the one-door case [13], but the others are much more complex.

Consider to pack n items to a car container with two doors (i.e., a rear door and a side (right) door, as shown in Figure 1), so that the utilization rate of the container is as large as possible. Let C denote the two-door container with size of $L \times W \times H$, i.e., $C = \{(x, y, z) \in \mathbb{R}^3 \mid 0 \le x \le L, 0 \le y \le W, 0 \le z \le H\}$, where L, W, and H are length, width, and height of the container, respectively. The n items to be packed to the container (which are n rectangular boxes) constitute set $B = \{b_i \mid b_i = (l_i, w_i, h_i) \in \mathbb{N}^3, i \in \{1, \ldots, n\}\}$, where l_i, w_i, and h_i are length, width, and height of item b_i. Length, width, and height of item b_i cannot exceed size of the container, i.e., $l_i \le L$, $w_i \le W$, and $h_i \le H$. Let S denote set of the packed items in container C, i.e., set S is a subset of set B. The *packing pattern* function P maps each item in S to the location and orientation of the item in the container C, i.e., P: $S \to \mathbb{N}^6$ with

$$P(b_i) = (P^x(b_i), P^y(b_i), P^z(b_i), P^l(b_i), P^w(b_i), P^h(b_i))$$

where $(p^x(b_i), p^y(b_i), p^z(b_i))$ is the coordinate of the left-bottom-back corner of item b_i packed in container C; $p^l(b_i)$, $p^w(b_i)$, and $p^h(b_i)$ are length, width, and height of item b_i packed in container C, respectively. Note that length, width, and height of the packed item could be rotated so that they are different before and after the packing. Hence, for each $b_i \in S$, (l_i, w_i, h_i) and (l_i', w_i', h_i') denote length, width, and height of the item before and after the item is packed, respectively. Location of item b_i in container C is denoted by $R(b_i) = [p^x(b_i), p^x(b_i) + p^l(b_i)] \times [p^y(b_i), p^y(b_i) + p^w(b_i)] \times [p^z(b_i), p^z(b_i) + p^h(b_i)]$. Based on the above notations, the mathematical programming model for the 3DCPP is established as follows [13]:

$$\text{Maximize } \sum_{b_i \in S} (l_i \cdot w_i \cdot h_i) / (L \cdot W \cdot H) \tag{1}$$

Subject to

$$p^x(b_i) + p^l(b_i) \le L, \ \forall\, b_i \in S \tag{2}$$

$$p^y(b_i) + p^w(b_i) \le W, \ \forall\, b_i \in S \tag{3}$$

$$p^z(b_i) + p^h(b_i) \le H, \ \forall\, b_i \in S \tag{4}$$

$$R(b_i) \cap R(b_j) = \varnothing, \ \forall b_i, b_j \in S, i \ne j \tag{5}$$

where Objective (1) is to maximize utilization ratio of the container; Constraints (2)-(4) enforces that the right, top, and front sides of each packed item cannot exceed those of the container; Constraint (5) enforces that every two packed items cannot be overlapped with each other.

As for the operations for subvolumes, consider to pack an item b_i to a subvolume $subv_j(x_j, y_j, z_j, L_j, W_j, H_j)$ in a two-door container with a packing pattern $P(b_i) =$

(x_j, y_j, z_j, $p^l(b_i)$, $p^w(b_i)$, $p^h(b_i)$). Three smaller subvolumes $subv_1$', $subv_2$', and $subv_3$' are generated from the up, front, and right sides of item b_i. Subvolume $subv_1$' is represented as follows:

$$subv_1'(x_j, y_j, z_j + p^h(b_i), p^l(b_i), p^w(b_i), H_j - p^h(b_i))$$

Subvolumes $subv_2$' and $subv_3$' are represented in the following two cases. In the case when $L_j > W_j$ (see Figure 2(a)),

$$subv_2'(x_j, y_j + p^w(b_i), z_j, p^l(b_i), W_j - p^w(b_i), H_j);$$
$$subv_3'(x_j + p^l(b_i), y_j, z_j, L_j - p^l(b_i), W_j, H_j).$$

In the case when $L_j < W_j$ (see Figure 2(b)),

$$subv_2'(x_j, y_j + p^w(b_i), z_j, L_j, W_j - p^w(b_i), H_j);$$
$$subv_3'(x_j + p^l(b_i), y_j, z_j, L_j - p^l(b_i), p^w(b_i), H_j).$$

In the two-door 3DCPP with home delivery service, items are required to be packed in a loading ordering of items. Let $\pi = (I_1, I_2, \ldots, I_n)$ denote the loading ordering of n items, where $I_i \in \{1, 2, \ldots, n\}$ and $I_i \neq I_j$, $\forall\, i \neq j$. Obviously, the unloading ordering of items is the reverse ordering of the loading ordering. Hence, let $\pi^R = (I_n, I_{n-1}, \ldots, I_1)$ denote the unloading ordering of n items. $\forall\, i \in \{1, 2, \ldots, n\}$, the loading ordering and unloading ordering of item b_i are represented by $\pi(b_i)$ and $\pi^R(b_i)$, respectively. Note that the loading ordering is assumed to be given, and hence, the unloading ordering can be obtained.

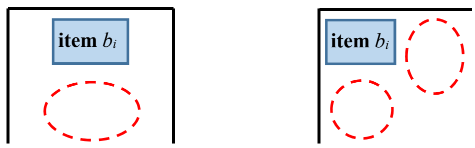
3.2 The unloading cost for two-door 3DCPP under home delivery service

The 3DCPP under home delivery service is concerned with two measures: the container utilization ratio and the unloading cost. This paper continues using the same setting in [13] that aims to maximize the utilization ratio of the container while the unloading cost is restricted to zero, i.e., the zero unloading cost constraint is concerned in the mathematical programming model stated in the previous subsection, because any non-zero unloading cost leads to much inconvenience and troubles of the deliveryperson in practice. In what follows, more details on how to compute the unloading cost and the zero unloading cost constraint in the two-door case are given, as they are much more complex than the one-door case.

Assume that we are unloading item b_i with loading ordering $\pi(b_i) = I_k$ (where $i \leq n$). To unload item b_i, the items with loading orderings in $\{I_t \mid 1 \leq t < k\}$ imped item b_i from being uploaded. Hence, those items are needed to be unloaded, and then are reloaded to the container again for later delivery. Let $un(b_i)$ denote the number of items with loading orderings in $\{I_t \mid 1 \leq t < k\}$ that are needed to be unloaded so that the required item b_i can be uploaded. For simplicity, the *unloading cost* $uc(b_i)$ is set to be equal to the unloading item number $un(b_i)$, i.e., $uc(b_i) = un(b_i)$ [13]. The unloading item number $un(b_i)$ can be calculated according to the following *rule of invisible and untouchable items* [13]:

- *Invisible item*: The required item b_i is blocked by or pressed below some item in $\{I_t \mid 1 \leq t < k\}$ so that the deliveryperson cannot see and unload item b_i.
- *Untouchable item*: The required item b_i cannot be touched by the deliveryperson because of his/her figure limitation.

The rule of invisible and untouchably items for the two-door 3DCPP is much different from the one-door 3DCPP. For the invisible item rule, Figure 5 illustrates the difference of the top views of the two cases. Item b_i is invisible in the one-door case if some item is located in the dotted region in Figure 5(a), but the two-door case is different. Item b_i is invisible in the two-door case if both the two dotted regions in Figure 5(b) have items.



(a) (b)

Figure 5. Illustration of the invisible item rule for (a) the one-door case and (b) the two-door case.

Next, the untouchable item rule for the two-door case is designed as follows. Consider to pack an item b_i in the container with two doors as illustrated in Figure 6, where the *packing pattern* of item b_i is $P(b_i) = (x_i, y_i, z_i, l_i', w_i', h_i')$. Assume that the deliveryperson stands at the location that is the closest to the item and can see the item, no matter whether he/she is from the rear or the right door. To define the unloading cost due to the untouchable item rule, the distances from feet of the deliveryperson to item b_i along the x-axis and z-axis are concerned, but they are different according to the door from which he/she intends to unload the item. See also Figure 6. If intending to unload the item from the rear door (i.e., along the x-axis), the distances from the feet of the deliveryperson to item b_i along the x-axis and z-axis are denoted by $L_{Xtouchable}$ and $H_{Xtouchable}$, respectively; otherwise (i.e., from the right door, or say, along the y-axis), they are denoted by $L_{Ytouchable}$ and $H_{Ytouchable}$, respectively. Hence, if the deliveryperson can use his/her hands to touch item b_i, then $L_{Xtouchable}$, $L_{Ytouchable}$, $H_{Xtouchable}$, and $H_{Ytouchable}$ must satisfy the following conditions:

$$L_{Xtouchable} + H_{Xtouchable} \leq 200 \tag{6}$$

$$L_{Xtouchable} \leq \min\{200 - z_i, 60\} \tag{7}$$

$$L_{Ytouchable} + H_{Ytouchable} \leq 200 \tag{8}$$

$$L_{Ytouchable} \leq \min\{200 - z_i, 60\} \tag{9}$$

where Constraints (6) and (7) (resp., Constraints (8) and (9)) are the condition that the deliveryperson can touch the item if standing at the rear (resp., side) door of the container. Note that item b_i is counted as one unit of the unloading cost due to the untouchability item rule if the deliveryperson cannot touch item b_i from both the rear door and the right door.

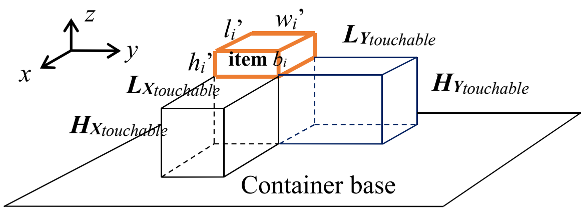


Figure 6. Illustration of packing item b_i to a container with two doors.

3.3 Our approach

This section extends the subvolume-based approach in [13] for the one-door case to solve the two-door case. The flowchart of the approach is illustrated in Figure 7, and is explained as follows. First, the algorithms initializes the subvolume list $SubV$, which includes only one subvolume with size equaling to the container size, and lets the iteration number k be 0. Then, a loop of at most η iterations is executed. In the loop, n items are packed into the container according to the loading order $\pi = (I_1, I_2, \ldots, I_n)$ until there is no suitable subvolume that can accommodate any item. Note that the subvolumes in list $SubV$ are sorted in the (x, y, z)-lexicographic order, in the case when $L > W$. That is, for any $(x_1, y_1, z_1), (x_2, y_2, z_2) \in \mathbb{N}^3$, $(x_1, y_1, z_1) < (x_2, y_2, z_2)$ if and only if 1) $x_1 < x_2$ or 2) $x_1 = x_2$ and $y_1 < y_2$ or 3) $x_1 = x_2$, $y_1 = y_2$ and $z_1 < z_2$.

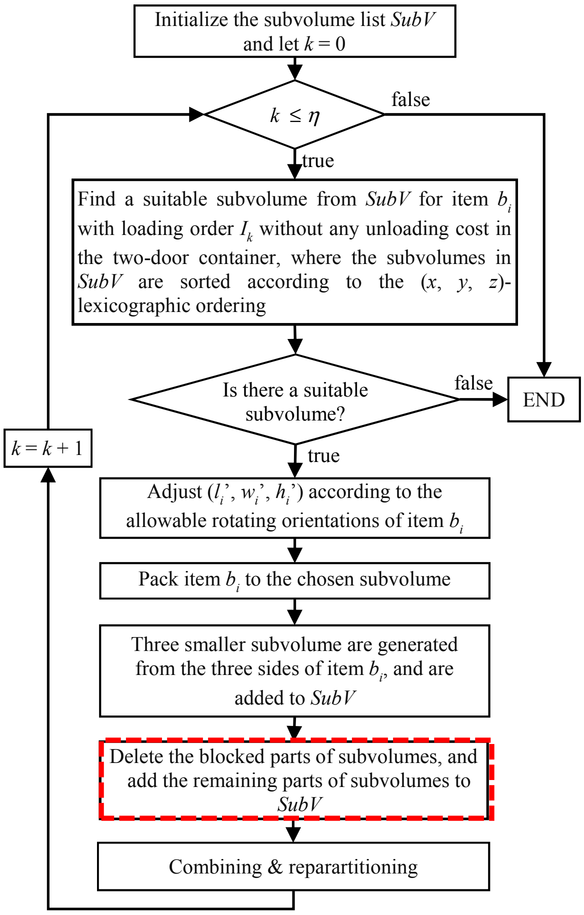


Figure 7. The flowchart of the proposed subvolume-based algorithm.

The algorithm finds a suitable subvolume for item b_i with loading ordering I_k from list $SubV$ according to the (x, y, z)-lexicographic order. If there exists a suitable subvolume, pack the item according to the allowable rotating orientation. After packing item b_i, three smaller subvomes are generated from the up, front, and right sides of item b_i and are added to list $SubV$ according to the (x, y, z)-lexicographic order. Then, since some subvolumes may be blocked by the added subvolumes according to the (x, y, z)-lexicographic order, the blocked parts of those subvolumes are needed to be removed. There are 16 cases for removing the blocked parts, as shown in Figure 8. Note that the 16 cases for the two-door problem is much difficult than the one-door problem. Then, as mentioned in Section 2, the combining and repartitioning operations are conducted for generating larger subvolumes or increasing diversification of the packing pattern.

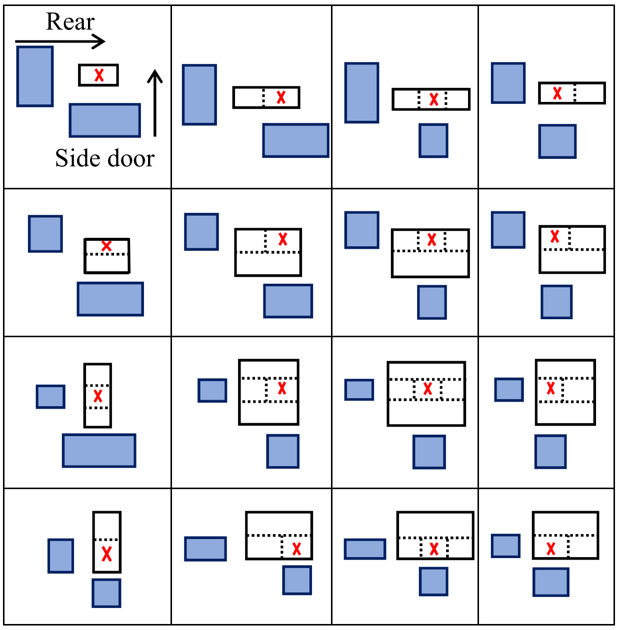


Figure 8. The 16 cases for removing the blocked parts in the two-door 3DCPP under home delivery service, in which the white rectangle in each subfigure is the concerned subvolume; each of the two blue shaded rectangles consists of some items; the blocked part in each white subvolume is marked by a cross symbol.

4. IMPLEMENTATION AND EXPERIMENTAL RESULTS

This section implements our approach to the two-door 3DCPP under home delivery service in C++ programming language. We apply the thpack7 benchmark data (including 100 problem instances) from the OR-LIBRARY, and compare the experimental result with the one-door case in [13]. Those experiments run on a PC with Intel Pentium CPU G645 @ 2.90GHz and memory of 4GB.

Since it takes very soon to run one time of our approach to the two-door case, and one more different run of our approach may generate a better solution, we try to execute different runs of our approach to find how many runs of our approach can obtain a good enough solution within a reasonable running time. 100, 400, 500 runs of our approach for the two-door case are given in Table 1, in which the values at the bottom row are the means of the corresponding columns. From Table 1, performance for 400 and 500 runs does not differ a lot from that for 100 runs. Hence, we apply 100 runs of our approach for comparison with the one-door case.

Table 1. Comparison of executing different runs of our approach to the two-door case

No.	100 runs			400 runs			500 runs		
	Item loaded	Utilize. ratio (%)	Run. time (s)	Item loaded	Utilize. ratio (%)	Run. time (s)	Item loaded	Utilize. ratio (%)	Run. time (s)
1	58	56.2	3.9	58	56.2	62	58	56.2	178
2	67	54.7	6.1	68	56.5	83	68	56.5	187
3	71	58.1	7.3	71	58.1	91	72	59.1	192
4	80	52.0	7.1	80	52.0	89	83	54.0	187
5	68	46.8	5.6	68	46.8	76	70	49.3	182
6	84	51.7	10.7	87	53.8	99	88	54.3	196
7	67	56.1	6.2	71	59.9	84	71	59.9	188
8	60	45.1	5.1	60	45.1	73	60	45.1	178
9	76	56.2	9.7	76	56.2	96	76	56.2	192
10	71	57.8	5.3	75	60.9	74	75	60.9	179
…	…	…	…	…	…	…	…	…	…
100	70	53.60	7.00	71	54.13	85.0	71	54.13	189
Mean	68.3	52.1	6.06	71.2	54.26	77.35	71.58	54.58	184.7

It is of interest to compare one-door and two-door cases, in terms of their performance measures (including item loaded, utilization ratio, and running time). 100,000 runs of the approach in [13] for the one-door case and 100 runs of the approach in this paper for the two-door case on 100 problem instances are compared in Table 2 and Figure 9. On the other hand, 100 runs of the approach in [13] for the one-door case and 100 runs of the approach in this paper for the two-door case on 100 problem instances are compared in Table 3 and Figure 10. From those experimental analysis, the approach proposed in this paper really performs better than the approach proposed in [13].

Table 2. Comparison of performance between 100000 runs for one-door case and 100 runs for two-door case

No.	100000 runs for the one-door case				100 runs for the two-door case				Diff. of utilize. ratios
	Item loaded	Utilize. ratio (%)	Unload. cost	Run. time (s)	Item loaded	Utilize. ratio (%)	Unload. cost	Run. time (s)	
1	45	44.6	0	10.2	58	56.2	0	3.9	11.6
2	57	46.2	0	18.6	68	56.1	0	6.1	9.93
3	57	46.8	0	19.6	67	54.7	0	7.3	7.9
4	63	43.0	0	22.9	80	51.8	0	7.1	8.8
5	56	35.0	0	18.6	66	43.7	0	5.6	8.74
6	63	39.5	0	24.6	83	51.0	0	10.7	11.53
7	52	42.8	0	15.4	67	56.1	0	6.2	13.3
8	41	29.0	0	10.7	60	45.1	0	5.1	16.14
9	72	53.7	0	28.8	76	56.2	0	9.7	2.5
10	45	36.9	0	13.4	65	50.5	0	5.3	13.6
…	…	…	…	…	…	…	…	…	…
100	67	50.6	0	22.2	70	53.6	0	7.0	2.96
Mean	54.41	41.7	0	17.7	67.69	51.7	0	6.1	9.98

Under zero unloading cost constraint, the mean of utilization ratios of using the proposed approach for the two-door case in Tables 2 and 3 is greater than that for the one-door case by 9.98% and 14.69%, respectively. Figures 9 and 10 show that in each case of number of runs, the proposed algorithm for the two-door case always produce results with higher utilization ratios than the one-door case. It is speculated that number of subvolumes for the two-door case is greater than that for the one-door case, so that more items could be loaded, and furthermore, the utilization ratio become higher.

Table 3. Comparison of performance of 100 runs for one-door and two-door cases

No.	100 runs for the one-door case				100 runs for the two-door case				Diff. of utilize. ratios
	Item loaded	Utilize. ratio (%)	Unload. cost	Run. time (s)	Item loaded	Utilize. ratio (%)	Unload. cost	Run. time (s)	
1	36	35.10	0	0.008	58	56.20	0	3.900	21.1
2	54	43.80	0	0.022	68	56.10	0	6.100	12.3
3	58	47.70	0	0.025	67	54.70	0	7.300	7
4	55	36.02	0	0.022	80	51.80	0	7.100	15.78
5	51	31.50	0	0.018	66	43.74	0	5.600	12.24
6	55	34.44	0	0.023	83	51.03	0	10.70	16.59
7	49	40.70	0	0.016	67	56.10	0	6.200	15.4
8	41	29.04	0	0.012	60	45.14	0	5.100	16.1
9	57	44.50	0	0.022	76	56.20	0	9.700	11.7
10	45	36.90	0	0.015	65	50.50	0	5.300	13.6
…	…	…	…	…	…	…	…	…	…
100	55	41.90	0	0.021	70	53.60	0	7.000	11.7
Mean	48.41	36.95	0	0.017	67.69	51.65	0	6.065	14.69

Table 4. Stability analysis of the proposed algorithm

No.	Mean of utilization ratios (%)	Standard deviation
1	57.52	0.73
2	55.26	0.83
3	54.79	1.07
4	51.30	0.55
5	45.97	1.36
6	50.87	0.79
7	54.63	1.00
8	46.36	0.25
9	57.54	0.56
10	58.39	0.57
…	...	...
100	53.12	0.65

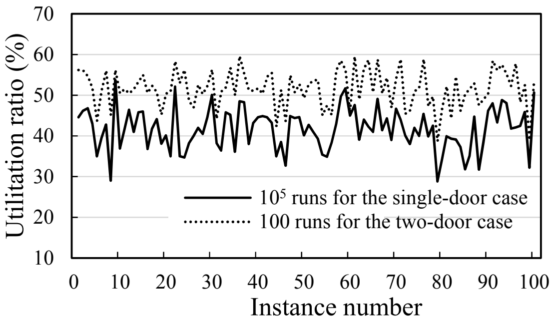


Figure 9. Comparison of utilization ratios of 10^5 runs for the single-door case and 100 runs for the two-door case for 100 problem instances.

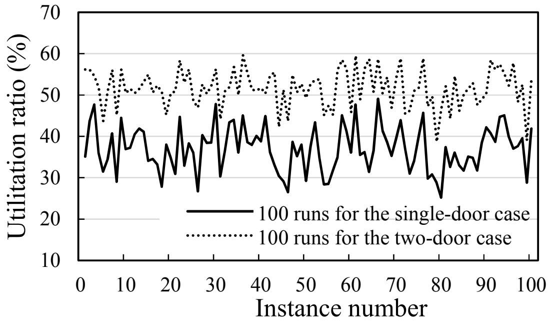


Figure 10. Comparison of utilization ratios of 100 runs for the single-door and the two-door cases for 100 problem instances.

To evaluate the stability of the proposed algorithm, we execute 20 runs of the proposed algorithm for 100 problem instances, and record the mean and standard deviation of the utilization ratios for each problem instance in Table 4. From Table 4, all the standard deviations are small, and less than 1.5. Additionally, to observe the stability of a specific problem instance, utilization ratios of the experimental results of the 100 runs of the proposed algorithm for the problem instance #36 is given in Figure 11, in which the difference of the maximal and minimal values is no greater than 0.05. As a result, it is concluded that the proposed algorithm performs stably.

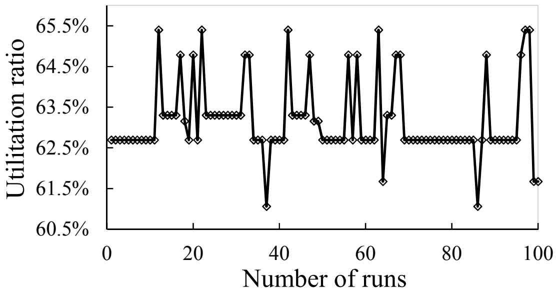


Figure 11. Utilization ratios of the experimental results of the 100 runs of the proposed algorithm for the problem instance #36.

5. CONCLUSION AND FUTURE WORK

This paper has made an extension of the previous work in [13] to solve the two-door 3DCPP under home delivery service with a subvolume-based approach. The motivation behind this work is that the car container has evolved from the one-door design to the two-door design, but the previous related academic works did not investigate the two-door 3DCPP. On the judgment of unloading cost, the *rule of invisible and untouchable items* proposed in [13] is extended to the two-door case. Experimental results show that the approach proposed in this paper for the two-door case really performs better than the approach proposed in [13] for the one-door case. Additionally, the proposed algorithm performs stably. The main reason why the proposed approach performs better is that the two-door 3DCPP generates more subvolumes to accommodate later items to be loaded.

In the future, it is of interest to design the problem setting with more practical requirements for home delivery service such as distribution types about cargo weights, breakable goods, and some urgent items in the container. Additionally, it is also of interest to solve the joint problem of 3DCPP and vehicular routing problem. Improved algorithms for the problem should also be investigated.

ACKNOWLEDGEMENTS

The authors thank the anonymous referees for comments that improved the content as well as the presentation of this paper. This work has been supported in part by MOST 104-2221-E-009-101 and MOST104-2410-H-156-017, Taiwan.

REFERENCES

1. Boyer, K.K., M.P. Andrea, and W. Chung, "The last mile challenge: evaluating the effects of customer density and delivery window patterns," *Journal of Business Logistics*, **30**(1), 185-201 (2009).
2. Berning, C., S. Ernst, and N. H. Hooker, "Are e-grocers serving the right markets?," *Choices*, **20**(1), 77-80 (2005).
3. Schonfeld, E., Forrester forecast: Online retail sales will grow to $250 billion by 2014, Available: *http://techcrunch.com/2010/03/08/forrester-forecast-online-retail-sales-will-grow-to-250-billion-by-2014/*, March 8 (2010).
4. Agatz, N., A. M. Campbell, M. Fleischmann, and M. Savels, "Challenges and opportunities in attended home delivery," in *The Vehicle Routing Problem: Latest Advances and New Challenges*, **43**, 379-396 (2008).
5. Kämäräinen, V. and M. Punakivi, "Developing cost-effective operations for the e-grocery supply chain," *International Journal of Logistics Research and Applications*, **5**(3), 285-298 (2002).
6. Punakivi, M., *Comparing Alternative Home Delivery Models for E-grocery Business*, Ph.D. Dissertation, Industrial Engineering and Management, Helsinki University of Technology, Finland (2003).
7. Garey, M. and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freemann and Company (1979).
8. Bortfeldt, A. and G. Wäscher, "Constraints in container loading: A state-of-the-art review," *European Journal of Operational Research*, **229**(4), 1-20 (2013).
9. Bortfeldt, A., H. Gehring, and D. Mack, "A parallel tabu search algorithm for solving the container loading problem," *Parallel Computing*, **29**(5), 641-662 (2003).
10. Mack, D., A. Bortfeldt, and H. Gehring, "A parallel hybrid local search algorithm for the container loading problem," *International Transactions in Operational Research*, **11**(5), 511-533 (2004).
11. Feng, X., I. Moon, and J. Shin, "Hybrid genetic algorithms for the three-dimensional multiple container packing problem," *Flexible Services and Manufacturing Journal*, **27**(2-3), 451-477 (2015).
12. Bortfeldt, A., "A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints," *Computers & Operations Research*, **39**(9), 2248-2257 (2012).
13. Liu, W.-Y., C.-C. Lin, and C.-S. Yu, "On the three-dimensional container packing problem under home delivery service," *Asia-Pacific Journal of Operational Research*, **28**(5), 601-621 (2011).
14. Junqueira, L., J. F. Oliveira, M. A. Carravilla, and R. Morabito, "An optimization model for the vehicle routing problem with practical three-dimensional loading constraints," *International Transactions in Operational Research*, **20**(5), 645-666 (2013).
15. Martello, S., D. Pisinger, and D. Vigo, "The three-dimensional bin packing problem," *Operations Research*, **48**(2), 256-267 (2000).
16. Padberg, M., "Packing small boxes into a big box," *Mathematical Methods of Operations Research*, **52**(1), 1-21 (2000).
17. Junqueira, L., R. Morabito, and D. S. Yamashita, "MIP-based approaches for the container loading problem with multi-drop constraints," *Annals of Operations Research*, **199**(1), 51-75 (2000).
18. Liu, S., W. Tan, Z. -Y. Xu, and X. -W. Liu "A tree search algorithm for the container loading problem," *Computers & Industrial Engineering*, **75**, 20-30 (2014).
19. Hifi, M. and L. Yousef, "Width beam and hill-climbing strategies for the three-dimensional sphere packing problem," *Federated Conference on Computer Science and Information Systems*, **2**, 7-10 (2014).
20. Pisinger, D., Heuristics for the container loading problem," *European Journal of Operational Research*, **141**(2), 382-392 (2002).
21. Abdou, G. and M. Elmasry, "3D random stacking of weakly heterogeneous palletization problems," *International Journal of Production Research*, **37**(7), 1505-1524 (1999).
22. Hifi, M., S. Negre, and L. Wu, "Hybrid greedy heuristics based on linear programming for the three-dimensional

single bin-size bin packing problem," *International Journal of Production Research*, **21**(1), 59-79 (2014).

23. Huang, W. and K. He, "A caving degree approach for the single container loading problem," *European Journal of Operational Research*, **196**(1), 93-101 (2009).
24. Wu, X., Y. Du, and S. Li, "A new approach for the three-dimensional single container loading problem," *Proceedings of 5th IEEE International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC2013)*, 155-158 (2013).
25. Moura, A. and J. F. Oliveira, "A GRASP approach to the container-loading problem," *Intelligent Systems*, **20**(4), 50-57 (2005).
26. Parreño, F., R. Alvarez-Valdés, J. M. Tamarit, and J. F. Oliveira, "A maximal-space algorithm for the container loading problem," *INFORMS Journal on Computing*, **20**(3), 412-422 (2008).

ABOUT THE AUTHORS

Chun-Cheng Lin received his B.S. degree in Mathematics, M.B.A degree in Business Administration, and Ph.D. degree in Electrical Engineering from National Taiwan University, Taiwan, in 2000, 2007, and 2009, respectively. Since August 2013, he has been an Associate Professor at National Chiao Tung University, which he initially joined as an Assistant Professor in February 2011. He was an Assistant Professor at University of Taipei (2010 - 2011) and National Kaohsiung University of Applied Sciences (2009 - 2010). His main research interests include graph drawing and information visualization, wireless networks, design and analysis of algorithms, as well as computational management science.

Jia-Rong Kang received his Ph.D. degree in Industrial Engineering and Management from National Chiao Tung University, Taiwan, in 2015. His main research interests include production scheduling, staff scheduling, wireless networks, as well as social networks.

Wan-Yu Liu received her B.B.A. degree in Finance from National Chengchi University, Taipei, Taiwan, in 2002; M.S. and Ph.D. degrees both in Agricultural Economics from National Taiwan University, Taipei, Taiwan, in 2004 and 2008, respectively. Since February 2016, she has been an Associate Professor of Forestry at National Chung Hsing University, Taiwan. She was an Associate Professor of Tourism Information (2012-2016) and Assistant Professor of Ecology and Tourism Management (2008-2012) at Aletheia University, Taiwan. Her main research interests include environmental and resource economics analysis, applied econometrics, evaluation of GHG reduction policies, as well as forest economics.

Cheng-Chi Li received his B.S. degree in Information Management from Shih Hsin University and M.S. degree in Industrial Engineering and Management from National Chiao Tung University in 2012 and 2014, respectively. His main research interests include operations research and project management.

(Received July 2015; revised November 2015; accepted December 2015)