A product arrangement optimization method to reduce packaging environmental impacts

C.T.T. Truong¹, L. Amodeo¹, F. Yalaoui¹, J.C. Hautefaye², S. Birebent²

¹ Laboratory of Optimization of Industrial Systems (LOSI), University of Technology of Troyes, Troyes 10300, France
² DS Smith Packaging Systems, Chenôve 21300, France

*E-mail: cong.tan.trinh.truong@dssmith.com

Abstract. The Three-dimensional Open Dimension Rectangular Packing Problem (3D-ODRPP) is one of the most important optimization problems arise in reducing waste and shipping cost of packing and shipping industries. The 3D-ODRPP aims at seeking the length, width and height of a rectangular box of minimal volume that can pack a given rectangular products. Most 3D-ODRPP models in the literature use too many extra binary variables and don’t consider the equilibrium of product placement. This study presents an improved mathematical model for the 3D-ODRPP. The proposed model uses fewer decision variables and constraints to define product orientation. It also adopts the Full base support (FBS) constraint as a product supporting condition to guarantee packaging stability and avoid infeasible arrangements. Literature instance tests show the improvement in packaging stability of the proposed method compared with existing methods. Industrial benchmark demonstrates that, by solving the 3D-ODRPP, the proposed method can reduce package volume and create economic and ecologic gains.

1. Introduction

The basic 3D-ODRPP represented in the typology of Cutting and Packing problems (C&P) [5] is one of the most studied three-dimensional packing problems, besides the Bin Packing (3D-BPP), Container Rectangular Loading (3D-CRLP) and Knapsack Problem (3D-KP). It focuses on finding the length, width and height of a minimal volume box that can accommodate a given set of rectangular products.

One of the first mathematical models of three-dimensional packing problems was introduced by Chen et al [1]. Their model deals with the 3D-CRLP. Based on the model of Chen et al. [1], Tsai et al. [3] present a mixed integer programming mathematical model for the 3D-ODRPP. Tsai et al. [3] adopt the piecewise linearization techniques presented by Vielma et al. [4] to linearize their model. However, the model of Tsai et al. [3] includes no product supporting constraint, so, in many cases, solutions are not feasible. A simple and effective formula of product supporting constraint for the 3D-CLP is the Full base support (FBS) condition [2]. The FBS condition requires every product’s bottom face must be entirely in contact with the bottom of the box or with the top face of another product.

This study proposes a new mathematical model to solve a practical 3D-ODRPP arises in a packaging system where right-sized boxes are made to pack given products. The objective is to determine the optimal arrangement the products and minimize box volume. As a result, packaging waste and shipping costs are reduced. Two contributions of this study are summarized next: firstly, the proposed method reformulates product orientation to use fewer decision variables. Secondly, this method implies the FBS
into the basic 3D-ODRPP to make sure that all products are well supported inside the box so that the solution is feasible.

The remaining of this paper is organized as follows: Section 2 presents the proposed mathematical models; Section 3 presents numerical experiments and analysis; conclusions and perspectives for future works are in Section 4.

2. Proposed models

2.1. Mathematical formulation

The 3D-ODRPP addressed in this work can be stated as follows: a given set of \( n \) products of parallelepiped shape are characterized by their length, width and height (the longest, second longest and shortest side of the product, respectively) are to be loaded into a box of parallelepiped shape whose length, width and height are variable. The objective is to achieve the minimum of box volume while meeting the following loading constraints:

- Every face of a product must be parallel to one of the box faces.
- There must be no intersection between any pair of products.
- All products must be placed entirely within the box.
- Full base support constraint (FBS): The bottom face of each product must be entirely in contact with the floor of the box or with the top face of another product.
- The dimensions of the box lie parallel to the \( x, y \), and \( z \)-axis, respectively, of the coordinate system, with the back-bottom-left corner being at the origin \( O \). The coordinate of a product is the point of its back-left-bottom corner.

The mathematical model \textbf{ODP_1} is represented as follows:

\begin{align}
\textbf{Parameters:} \\
& n: \text{Number of products to be packed.} \\
& p_i, q_i, r_i: \text{Length, width and height of product } i; p_i \geq q_i \geq r_i \quad \forall i \in \{1 \ldots n\}. \\
& M: \text{Big number used in the model. } M = \sum_{i=1}^{n} p_i
\end{align}

\begin{align}
\textbf{Variables:} \\
& x_i, y_i, z_i (i \in \{1 \ldots n\}): \text{Continuous variables indicating coordinates of products.} \\
& X, Y, Z: \text{Continuous variables for the length, width, and height of the box.} \\
& a_{i,j} \ (i \in \{1 \ldots n\}; \ j \in \{1 \ldots 6\}): \text{Binary variables indicating whether the product } i \text{ has orientation } j. \text{ The orientations are defined as shown in Table 1.} \\
& a_{i,j}, b_{i,j}, c_{i,j}, d_{i,j}, e_{i,j}, f_{i,j} \ (i,j \in \{1 \ldots n\}): \text{Binary variables indicating relative positions (on the left, on the right, behind, in front, below, above) of products } i \text{ and } j \text{ [1]. For example, if product 2 is on the left side of product 3 then } a_{2,3} = 1, \text{ otherwise, } a_{2,3} = 0. \\
& \zeta_{i,j} \ (i,j \in \{1 \ldots n\}): \text{Binary variable indicating if the product } i \text{ is supported by the product } j.
\end{align}

\begin{table}[h]
\centering
\begin{tabular}{lcccccc}
\hline
Orientations & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
Side parallel to x-axis & \( p \) & \( p \) & \( q \) & \( q \) & \( r \) & \( r \) \\
Side parallel to x-axis & \( q \) & \( r \) & \( p \) & \( r \) & \( p \) & \( q \) \\
Side parallel to x-axis & \( r \) & \( q \) & \( r \) & \( p \) & \( q \) & \( p \) \\
\hline
\end{tabular}
\caption{Product orientations.}
\end{table}

Objective function:

\begin{equation}
\text{Minimize } \quad X \times Y \times Z
\end{equation}

Subject to:

\begin{align}
\sum_{j=1}^{n} a_{i,j} &= 1 \quad \forall i \in \{1 \ldots n\} \\

a_{i,j} + b_{i,j} + c_{i,j} + d_{i,j} + e_{i,j} + f_{i,j} &\geq 1 \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
x_i + p_i(a_{i,3} + a_{i,2}) + q_i(a_{i,3} + a_{i,4}) + r_i(a_{i,5} + a_{i,6}) &\leq x_j + M(1 - a_{i,j}) \quad \forall i, j \in \{1 \ldots n\}; i \neq j
\end{align}
\begin{align}
x_j + p_i (o_{j3} + o_{j2} + q_i (o_{j3} + o_{j4}) + r_j (o_{j5} + o_{j6}) & \leq x_i + M (1 - b_i) \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
y_i + p_i (o_{i3} + o_{i2} + q_i (o_{i1} + o_{i4}) + r_i (o_{i2} + o_{i4}) & \leq y_j + M (1 - c_i) \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
y_j + p_j (o_{j3} + o_{j2} + q_j (o_{j1} + o_{j6}) + r_j (o_{j2} + o_{j4}) & \leq y_i + M (1 - d_i) \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
z_i + p_i (o_{i4} + o_{i6}) + q_i (o_{i2} + o_{i5}) + r_i (o_{i1} + o_{i3}) & \leq z_j + M (1 - e_i) \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
z_j + p_j (o_{j4} + o_{j6}) + q_j (o_{j2} + o_{j5}) + r_j (o_{j1} + o_{j3}) & \leq z_i + M (1 - f_i) \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
X & \geq x_i + p_i (o_{i1} + o_{i2} + q_i (o_{i3} + o_{i4}) + r_i (o_{i5} + o_{i6}) \quad \forall i \in \{1 \ldots n\} \\
Y & \geq y_i + p_i (o_{i3} + o_{i5}) + q_i (o_{i1} + o_{i4}) + r_i (o_{i2} + o_{i4}) \quad \forall i \in \{1 \ldots n\} \\
Z & \geq z_i + p_i (o_{i4} + o_{i6}) + q_i (o_{i2} + o_{i5}) + r_i (o_{i1} + o_{i3}) \quad \forall i \in \{1 \ldots n\} \\
M \times \sum_{i=1, i \neq i}^{\infty} (\zeta_{i,j}) & \geq z_i \quad \forall i \in \{1 \ldots n\} \\
x_i + M (1 - \zeta_{i,j}) & \geq x_j \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
y_i + M (1 - \zeta_{i,j}) & \geq y_j \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
x_j + p_j (o_{j3} + o_{j2} + q_j (o_{j3} + o_{j4}) + r_j (o_{j5} + o_{j6}) + M (1 - \zeta_{i,j}) & \geq x_i \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
y_j + p_j (o_{j3} + o_{j2} + q_j (o_{j1} + o_{j6}) + r_j (o_{j2} + o_{j4}) + M (1 - \zeta_{i,j}) & \geq y_i \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
z_j + p_j (o_{j4} + o_{j6}) + q_j (o_{j2} + o_{j5}) + r_j (o_{j1} + o_{j3}) + M (1 - \zeta_{i,j}) & \geq z_i \quad \forall i, j \in \{1 \ldots n\}; i \neq j \\
z_j + p_j (o_{j4} + o_{j6}) + q_j (o_{j2} + o_{j5}) + r_j (o_{j1} + o_{j3}) & \leq z_i + M (1 - \zeta_{i,j}) \quad \forall i, j \in \{1 \ldots n\}; i \neq j
\end{align}

The constraint (2) shows that each product has only one among its six possible orientations. Constraint (3) assures that there is at least one relative position between any two products so that there will be no intersection. Constraints (4) to (9) define the relative positions. For example, if product \( i \) is on the left side of product \( j \) (\( a_{ij} = 1 \)) then the x-coordinate of \( i \) plus its side that is parallel to the x-axis is not greater than the x-coordinate of \( j \). Constraints (10), (11) and (12) mean all products must be placed entirely inside the box. Constraint (13) means if the z-coordinate of a product is greater than zero then it must be supported by another product. Constraints (14) to (19) imply that if product \( i \) is supported by \( j \), its bottom face must be entirely in contact with top face of \( j \) (full base support condition). Constraints (20) to (21) represent the upper and lower bounds of box length, width, height and volume.

2.2. Linearization

As the model \textbf{ODP-1} has a non-linear objective function, it is difficult and requires lots of time to obtain the optimal solution. To linearize the objective function (1), logarithmic transformations and the piecewise function linearization technique presented in [4] and [3] are applied. The following additional parameters, variables and constraints are included to the linearized model \textbf{ODP-2}:

\textbf{Parameters}:
- \( m \) : Number of breakpoints for the piecewise function.
- \( x_k^\phi, y_k^\phi, z_k^\phi \) : Value of \( X, Y, Z \) at the breakpoint \( k \) on the axes \( x, y, \) and \( z \).

\textbf{Variables}:
- \( f_X, f_Y, f_Z \) : Piecewise function of \( \ln(X), \ln(Y), \ln(Z) \), respectively.
- \( \lambda^X_i, \lambda^Y_i, \lambda^Z_i \) : Continuous variables for piecewise functions of \( X, Y \) and \( Z \), respectively.
- \( u_k^X, u_k^Y, u_k^Z \) : Binary variables for SOS2 piecewise functions [3] of \( X, Y \) and \( Z \), respectively.

\textbf{Linearization constraints}:
\begin{align}
f_\phi &= \sum_{i=1}^{m} (\ln(a_{i}^\phi) \lambda_{i}^\phi) \quad \phi = \sum_{i=1}^{m} a_{i}^\phi \lambda_{i}^\phi \quad \sum_{i=1}^{m} \lambda_{i}^\phi = 1 \quad \forall \phi \in \{X, Y, Z\} \\
\sum_{i \in S^+ (k)} \lambda_{i}^\phi & \leq u_k^\phi \quad \sum_{i \in S^- (k)} \lambda_{i}^\phi \leq 1 - u_k^\phi \quad \lambda_{i}^\phi \in \{0; 1\} \quad u_k^\phi \in \{0; 1\} \quad \forall \phi \in \{X, Y, Z\}
\end{align}
The constraint (21) can be rewritten as follow:

\[
\ln \left( \sum_{i=1}^{n} (p_i q_i r_i) \right) \leq f_X + f_Y + f_Z \leq \ln \left( \sum_{i=1}^{n} p_i \times \left( \max_{i \in [1, n]} q_i \right) \times \left( \max_{i \in [1, n]} r_i \right) \right)
\]  

(24)

The linearized model ODP_2 is:

\[
\text{Minimize } \ f_X + f_Y + f_Z
\]

Subject to: (2) to (20) and (22) to (24)

3. Numerical experiments

In the following experiments, two sets of test instances will be solved to evaluate the impact of the FBS constraint on 3D-ODRPP. The first set includes ten literature problems (T1 to T10) derived from [3]. The second problem set is a log of 40 client orders drawn from the database of a packaging company in France. All problems are solved in ILOG Cplex 12.8.0 runs on an Intel Core i7 at 2.70 GHz computer with 32 GB of RAM.

Let Model_1 be the model proposed by Tsai et al. [3]. Table 2 shows box volume and filling rate (the ratio of product volume to box volume) of problems T1 to T10 given by the Model_1 (without FBS) and the ODP_2 (with FBS). Let M1 be the mean number of supporting products of the products that are not placed on the floor of the box, M2 be the mean percentage of the supported surface of the products that are not placed on the floor of the box. M1 and M2 are N/A when all products are placed on the bottom of the box. The solutions whose M1 less than 1 are not feasible, which means there is at least one product floats in the air without any support.

Table 2. Computational results of Model_1 (1) and the model ODP_2 (2) with m = 256.

| Ins. | No. of products | Box volume | Filling rate (%) | Model_1 M1 | Model_2 M2 | CPU time (s) |
|------|-----------------|------------|------------------|------------|------------|--------------|
|      |                 |            | (1)              | (1)        | (1)        | (1)          |
| T1   | 4               | 4368       | 82.78            | 82.78      | 0          | N/A          |
| T2   | 5               | 5040       | 86.33            | 86.33      | 1          | 100          |
| T3   | 6               | 5880       | 84.08            | 84.08      | 1          | 100          |
| T4   | 7               | 7040       | 87.27            | 85.05      | 1          | 90.91        |
| T5   | 8               | 360        | 81.67            | 81.67      | 1          | 100          |
| T6   | 9               | 480        | 74.58            | 74.58      | 0.75       | 100          |
| T7   | 4               | 217170     | 85.71            | 85.71      | 1          | 100          |
| T8   | 5               | 290700     | 87.64            | 87.64      | 1          | 35.30        |
| T9   | 6               | 372600     | 90.17            | 90.17      | 0.5        | 30.52        |
| T10  | 7               | 449450     | 94.07            | 92.22      | 1          | 89.29        |

| Mean |                 |            |                 |            |            | 85.43        |

* Best values are highlighted in bold.

Without product supporting constraint (Model_1), the average filling rate is 85.43%. When full base support constraint is enforced (ODP_2), this value is 85.02%. There are only two out of ten problems (T4 and T10) have different filling rates. It is possible to say that the impact of FBS constraint on filling rate is not enormous, with an average decrease of about 0.41% from ten test instances. In the other hand, as the Model_1 considers no product supporting constraint, it is evident that there are unsupported or not fully supported products in its solutions. The metric M1 on problem T1, T6 and T9 of Model_1 is smaller than 1, so solutions for these problems are not feasible. For the ODP_2, if a product is not placed on the floor of the box, it will always be supported by another product, so M1 is equal to 1. In terms of supporting rate, the mean M2 of Model_1 is only 68.98% while that of the ODP_2 is always 100%. The ODP_2 always give more stable solutions than the Model_1. To demonstrate the impact of FBS on product stability, let us consider problem T6. Figure 1 shows that the result given by Model_1 contains products that aren't placed in stable equilibrium positions (M1 or M2 of these products are under 1), such as the gray product lying on top of a much smaller one (green) has M2 equal to 25% and the pink one floating in the air has M1 equal to 0. In contrast, the solution given by the proposed model has all products well supported without overhanging nor unstable equilibrium placement (Figure 2).

Table 2 also shows the computational time of the Model_1 and the ODP_2. For the small-sized products (T1 to T6) the proposed model solves the problem in a shorter time for five out of six tests. For industrial products (T7 to T10), the Model_1 shows a better computational time. The computational time...
of both models depends on product size, heterogeneity and number of products. For the model \textit{ODP.2}, it can take up to more than thirteen minutes to solve the problem of seven industrial products (T10). For larger problems, CPU time will be much longer. Note that although T5 and T6 have more products than T10, the products of T5 and T6 are small-sized and homogeneous (all products are cube), therefore, the results are found in a shorter time than T10.

As introduced in Section 2.1, the proposed model uses fewer binary variables and constraints to define product orientation. It's can be seen that the model of Tsai et al. [3] needs \(9 \times n\) binary variables to define product orientations and \(6 \times n\) constraints to assure there is only one orientation for each product while the proposed model needs only \(6 \times n\) variables and \(n\) constraints for the same condition. Table 3 shows the number of variables and constraints of each model for problems T1 to T3, with \(m = 256\). It demonstrates that the proposed model needs \(3 \times n\) binary variables and \(5 \times n\) constraints fewer than that of Tsai et al. [3] for product orientations.

![Figure 1. Tsai et al. [3]](image1)

| Problem T1 (4 products) | Tsai et al. [3] | Proposed model |
|-------------------------|-----------------|----------------|
| CPU time (hh:mm:ss)    | 00:00:01        | 00:00:02       |
| No. of variables/constraints for product orientation | 36/24           | 24/4           |
| Box size (in cm)       | (28 \times 26 \times 6) | (28 \times 26 \times 6) |

| Problem T2 (5 products) | Tsai et al. [3] | Proposed model |
|-------------------------|-----------------|----------------|
| CPU time (hh:mm:ss)    | 00:00:07        | 00:00:05       |
| No. of variables/constraints for product orientation | 45/30           | 30/5           |
| Box size (in cm)       | (30 \times 28 \times 6) | (30 \times 28 \times 6) |

| Problem T3 (6 products) | Tsai et al. [3] | Proposed model |
|-------------------------|-----------------|----------------|
| CPU time (hh:mm:ss)    | 00:00:40        | 00:00:30       |
| No. of variables/constraints for product orientation | 54/36           | 36/6           |
| Box size (in cm)       | (35 \times 28 \times 6) | (35 \times 28 \times 6) |

Table 4 shows the results of the second set of test instances where problem size is between 2 and 5. The dimensional weight \(Dw\) of each order given by the company’s data is compared to that given by the proposed method. \(Dw\) (kg), also known as volumetric weight, is a pricing technique for commercial freight transport, which uses an estimated weight that is calculated as follows: \(Dw = \text{max}(\frac{V_{box}}{5000}, \mu)\) where \(\mu/(kg)\) is total product weight of the order, and \(V_{box} (cm^3)\) is the volume of the box. It is shown that if the company had implemented the proposed method, they could have saved 11.6\% (86.94 kg) of dimensional weight for the 40 given orders. In terms of mean box volume, the company’s resolution is 16.56\% greater than that of the proposed method.

4. Conclusions
This study has first proposed a new mathematical model for the 3D-ODRPP. Compared with existing models, the proposed model uses fewer decision variables for product orientations. The model presented in this paper also includes the FBS constraints to ensure product static equilibrium, which is one of the most important constraints of three-dimensional packing problems that guarantee the feasibility of solutions. Ten test instances drawn from [3] are solved to demonstrate the improvement of result quality given by the proposed model comparing to existing models.

However, FBS constraint in this study doesn’t cover some product supporting situations like multi-base support where a product can be supported by multiple products and product overhanging which is physically acceptable in many cases.
As suggestions for future work, it would be interesting to apply more effective product supporting strategies that allow product overhanging and multi-base support. To deal with bigger instances or shorter limits of computation time, approximation methods or packing heuristics shall be proposed in the next studies.

| References |
|-----------------|
| [1] C. Chen, S.-M. Lee, and Q. Shen. An analytical model for the container loading problem. European Journal of Operational Research, 80(1):68–76, 1995. |
| [2] J. F. Gonçalves and M. G. Resende. A parallel multi-population biased random-key genetic algorithm for a container loading problem. Computers & Operations Research, 39(2):179–190, 2012. |
| [3] J.-F. Tsai, P.-C.Wang, and M.-H. Lin. A global optimization approach for solving three-dimensional open dimension rectangular packing problems. Optimization, 64(12):2601–2618, 2015. |
| [4] J. P. Vielma and G. L. Nemhauser. Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. Mathematical Programming, 128(1-2):49–72, 2011. |
| [5] G. Wäschcher, H. Haßner, and H. Schumann. An improved typology of cutting and packing problems. European Journal of operational research, 183(3):1109–1130, 2007. |