The determination of the guillotine restrictions for a Rectangular Three Dimensional Bin Packing Pattern

DANIELA MARINESCU
Transilvania University of Brașov
Department of Computer Science
Iuliu Maniu 50, 500091, Brașov
ROMANIA
mdaniela@unitbv.ro

ALEXANDRA BĂICOIANU
Transilvania University of Brașov
Department of Computer Science
Iuliu Maniu 50, 500091, Brașov
ROMANIA
a.baicoianu@unitbv.ro

DANA SIMIAN
University Lucian Blaga of Sibiu
Department of Computer Science
Dr. Ion Ratiu 5-7, 550002, Sibiu
ROMANIA
dana.simian@ulbsibiu.ro

Abstract: This paper is regarding to the rectangular three dimensional bin packing problem, where a bin is loaded with a set of rectangular boxes, without overlapping. One of the most popular restriction for the solution of the 3D-bin packing problem is the guillotine restriction. That means that the packing patterns are so that the boxes can be obtained by sequential face-to-face cutting plane parallel to a face of the bin. Our objective is to find a method for verifying if a solution of the 3D-bin packing problem has the guillotine constrains or not. For this purpose we use a weighed graph representation of a solution of the problem, the generalization of this kind of representation obtained by us for 2D-cutting stock problem in [9, 10, 11].

Key–Words: 3D-bin packing problem, guillotine constraints, bin-packing pattern representation

1 Introduction

From the computational complexity theory point of view, the bin packing problem is a combinatorial NP-hard problem. In it, box-shaped objects of different sizes must be placed into a finite number of bins in a way that minimizes the number of bins used. In [4] many kinds of bin packing problems were presented, one dimensional, two dimensional and three dimensional with many kinds of constrains depending on technological restrictions. Of course the difficulty of the problem is increasing in three-dimensional bin-packing problem comparing to the difficulty of fewer dimensional bin-packing problems, but keeps special and important applications.

In the three-dimensional bin-packing problem each bin represents triplets containing three values: width, length, and height. Each box should fit into a bin or bins most efficiently. Like 2D bin-packing, each box must stay orthogonal, or maintains its orientation in the packing. Like all bin-packing problems, extra constraints may be added to the problem to create a more real-world-like problem. Such constraints are: gravity, weight distribution or delivery time and so-called guillotine constraint. The guillotine constraint requires that the patterns should be such that the boxes can be obtained recursively by cutting the bin in two smaller bins, until each bin will contains only one box and no box has been intersected by a cut.

While for the general three dimensional bin packing problem there are a lot of algorithms for packing patterns determination, exact algorithms [5] genetic algorithms [6], or approximation algorithms [7, 8], the general problem with guillotine restrictions is more difficult to solve. In [1] the authors solve a general packing problem, where in each step they test for satisfaction of guillotine constraints. Using some graph representations, defined by us in [12], we present now another guillotine test for the three dimensional bin packing patterns, by generalizing the results obtained in [9, 10] for the two dimensional cutting-stock patterns.

2 Problem formulation

We consider a three dimensional rectangular bin \( B \), a container with length \( L \), width \( W \), height \( H \). The bin \( B \) is filled with \( k \) rectangular boxes, \( C_1, C_2, \ldots, C_k \) without overlapping. Every box \( C_i \) has length \( l_i \), width \( w_i \), height \( h_i \).

**Definition 1** A rectangular 3D-bin packing pattern is an arrangement of the \( k \) rectangular boxes \( C_i \) in the container \( B \), so that the faces of the boxes \( C_i \) be parallel with the faces of the container \( B \).

**Definition 2** A rectangular bin packing pattern has guillotine restrictions if the bin can be recursively separated in two new bins by a cutting plane which is parallel with a face of the original bin, until each bin contains only one box.
We presented in [13, 14, 12] a representation for a bin packing pattern by means of some graphs of adjacency. Now we complete the graphs of adjacency by adding a value for each arc of these graphs like in [9, 10].

We consider a bin $OABCDEFG$ and a coordinate system $xOyz$ so that the corner $O$ is the origin of the coordinate system like in Figure 1.

The following notations are used:
- $ABCO$ is the bottom face of the bin
- $GDEF$ is the top face of the bin
- $OADG$ is the West face of the bin
- $OCFG$ is the North face of the bin
- $EBCF$ is the East face of the bin
- $ABDE$ is the South face of the bin
- $O - C_i$ is the O-corner of the box $C_i$ of coordinates $x_i, y_i, z_i$

![Figure 1: The position of box $C_i$ in the bin](image)

We mention that in Figure 1, $[AB]$ is the length $L$, $[BC]$ is the width $W$ and $[AD]$ is the height $H$. We will use the adjacency relations [12] to express the connections between two boxes $C_i$ and $C_j$ from the bin packing pattern.

**Definition 3** The box $C_i$ is adjacent in $Ox$ direction with the box $C_j$ in the bin packing pattern of $B$ (Figure 2), if the South face of $C_i$ and the North face of $C_j$ have at least three non-collinear common points.

Similarly we can define the adjacency relations in the direction $Oy$ and $Oz$.

Starting from the three kinds of adjacency we have defined in [12] three kind of graphs: $G_{Ox}$ the graph of adjacency in direction $Ox$, $G_{Oy}$ the graph of adjacency in direction $Oy$ and $G_{Oz}$ the graph of adjacency in direction $Oz$.

Now we complete these graphs by adding the values for every arc from the graphs which represent a bin packing pattern with respect to the restrictions from Remark 4.

**Definition 5** The weighed graph of adjacency in $Ox$ direction for the bin-packing pattern is $G_{Ox} = (C \cup R_X, \Gamma_{Ox})$, where the vertices are the boxes from $C = C_1, C_2, ..., C_k$, $R_X$ represents the face $GOCF$ situated on the $yOz$ plane, and

$$\Gamma_{Ox}(C_i) \ni C_j \text{ only if } C_i \text{ is adjacent in direction } Ox \text{ with } C_j$$

$$\Gamma_{Ox}(X) \ni C_i \text{ only if the North face of } C_i \text{ touches the } yOz \text{ plan}$$

$$\text{Value}(U, C_j) = w_j, \forall U \in C \cup R_X \text{ and } C_j \in C$$

**Remark 4** In the following we consider only the bin-packing pattern where the boxes are not situated above, to the East directions and to the North directions of an empty space. That means every box $C_j$ is adjacent with at least three boxes: one situated down, one to the West and one to the South, or $C_j$ is situated on the down face, respectively West face, or on the South face of the bin. Otherwise we will push the box $S$ downwards, either towards the South or the West directions, like in Figure 3 until $S$ will satisfy these conditions.

![Figure 3: The moving directions](image)
Similarly we can define a graph of adjacency in $Oy$ direction and another of adjacency in $Oz$ direction, using $w_j$ respectively $h_j$ for the every value of an incoming arc of $C_j$.

From the Remark 4 and from [13, 14] it follows that all of the three weighed graphs of adjacency are strongly quasi connected.

**Example 1.** We consider a bin-packing pattern described in the Figures 4 and 5 where the bin has the dimensions $L = 3$, $W = 3$, $H = 4$ and the boxes are of the dimensions $(l_i, w_i, h_i)$ like in the following:

- the box $A$ of dimension $(1, 3, 2)$
- the box $B$ of dimension $(1, 1, 1)$
- the box $C$ of dimension $(1, 1, 2)$
- the box $D$ of dimension $(1, 1, 1)$
- the box $E$ of dimension $(2, 2, 2)$
- the box $F$ of dimension $(3, 1.5, 2)$
- the box $G$ of dimension $(2, 1.5, 2)$
- the box $H$ of dimension $(1, 1.5, 2)$

Figure 4: A pattern view from the top-right-front corner for Example 1

Then the $G_{Ox}$, $G_{Oy}$ and $G_{Oz}$ are the weighed graphs from Figures 6 and 7.

We observe that the bin-packing pattern from Figures 4 and 5 has guillotine restrictions. Similar with [15, 16] it follows that it is possible to represent a bin-packing pattern with guillotine restrictions using a Polish expression with three operations:

1. $\oplus$ - the vertical concatenation, an operation for a horizontal cutting plane;
2. $\ominus$ - the W-E concatenation, an operation for a vertical cutting plane perpendicular on $Ox$;
3. $\oslash$ - the N-S concatenation, an operation for a vertical cutting plane perpendicular on $Oy$.

For example, the cutting pattern from Figure 4 will be described by the following Polish expression:

$$\ominus \oslash A \ominus E \ominus BDC \ominus F \ominus HG$$

Figure 5: A pattern view from the bottom-left-back corner for Example 1

Figure 6: Graph $G_{Ox}$ for Example 1.

3. **Cuts determination**

In the previous papers of us we presented two methods for cuts determination in case of a 2D-cutting pattern without overlapping: one for pattern without gaps [11] and one for the pattern with gaps [9, 10].

Now we consider the 3D-bin packing pattern without overlapping but with possible gaps, which respect the conditions from Remark 4.
Following the way described in [9, 10] we intend to find a connection between guillotine restrictions and the three weighed graphs of adjacency, $G_{Ox}$, $G_{Oy}$ and $G_{Oz}$.

First we will use the notation $\text{Lpd}(R_X, C_i)$ for the length of the path from $R_X$ to $C_i$ in the graph $G_{Ox}$. Similarly we will use the notations $\text{Lpr}(R_Y, C_i)$ for the length of the path from $R_Y$ to $C_i$ in the graph $G_{Oy}$, respectively $\text{Lpr}(R_Z, C_i)$ for the length of the path from $R_Z$ to $C_i$ in the graph $G_{Oz}$. We remark that $\text{Lpd}(R_X, C_i)$ represents the distance from the northern face of the bin $B$ to the southern face of box $C_i$, $\text{Lpr}(R_Y, C_i)$ represents the distance from the western face of the bin $B$ to the eastern face of box $C_i$ and $\text{Lpr}(R_Z, C_i)$ represents the distance from the bottom face of the bin to the top face of box $C_i$.

**Remark 6** If a cutting-stock pattern has a horizontal guillotine cutting plane (perpendicular on $Oz$) situated at a distance $M$ from the down face of the bin $B$ then the set of the items, $C$, can be separated in two subsets $B_1$, the set of the items situated below this cutting plane, and $B_2$, the set of the items situated above this plane. Of course in the weighed graph $G_{Oz}$ we have:

1. $\text{Lpd}(R_Z, C_i) \leq M$ for every $C_i \in B_1$;
2. $\text{Lpd}(R_Z, C_i) > M$ for every $C_i \in B_2$.

We obtain a similar result if the cutting-stock pattern has a vertical cutting plane perpendicular on $Ox$ or a vertical cutting plane perpendicular on $Oy$.

The two conditions from the above remark are necessary but are not sufficient, because it is possible the cutting plane to intersect some items from the set $B_2$. In the following we present necessary and sufficient conditions for a guillotine cut.

**Theorem 7** Let a 3D bin packing pattern with possible gaps and the weighed graph $G_{Oz}$ attached to this pattern. The bin packing pattern has a horizontal guillotine cutting plane situated at the distance $M$ from the downwards face of the bin if and only if it is possible to separate the sets of the items, $C$, in two subsets, $B_1$ and $B_2$ so that:

1. $C = B_1 \cup B_2$, $B_1 \cap B_2 = \emptyset$;
2. For every $C_i \in C$ so that $(R_Z, C_j) \in \Gamma_z$ it follows that $C_j \in B_1$;
3. $\text{Lpd}(R_Z, C_i) \leq M$ for every $C_i \in B_1$;
4. If there is $C_j \in B_1$ so that $\text{Lpd}(R_Z, C_j) < M$ then all direct descendant of $C_j$ will be in $B_1$.

**Proof:**

i. Suppose that the bin packing pattern has a horizontal guillotine cutting plane and let the weighed graph $G_{Oz}$ attached to the pattern. That means the sets of items $C$ can be separated in two subsets, $B_1$, the set of the vertices situated above the cutting plane, and $B_2$, the set of the vertices situated below the cutting plane. From the Remark 6 it follows that the conditions 1, 2 and 3 are fulfilled.

Suppose that the condition 4 is not fulfilled. That means there are two items $C_j \in B_1$ and $C_i \in B_2$ so that $\text{Lpd}(R_Z, C_j) < M$ the item $C_i$ is a direct successor of $C_j$ and suppose that $C_i \in B_2$. It follows that $\text{Lpd}(R_Z, C_i) > M$ and a horizontal cutting plane situated on the distance $M$ from the downwards face of the bin will intersects the box $C_i$. It means that without the condition 4 it is impossible to separate the set of the items by a horizontal cutting plane. So our supposition that the condition 4 is not fulfilled is false.

ii. Suppose all the conditions 1-4 are fulfilled but it is not possible to have a horizontal cutting plane at the distance $M$ in the cutting-stock pattern. It follows that there is at least item $C_i \in B_2$ which is intersected by such a cut. It means that the distance from the bottom face of the bin to the bottom face of the box $C_i$ is less than $M$ and the distance from the downwards face of the bin to the top face of the box $C_i$ is greater than $M$.

But from the Remark 6 it follows that the bottom face of the box $C_i$ is identical with the top face of some box $C_j$, situated downwards $C_i$. That means $(C_j, C_i) \in \Gamma_z$ and $\text{Lpd}(R_Z, C_j) < M$ and so $C_j \in$...
4 Verification test for guillotine restrictions

The results from the previous theorem suggest an algorithm for verification of the guillotine restrictions, in case of a bin-packing pattern with gaps but without overlapping.

**Input data:** The weighed graphs $G_{Ox}$ or $G_{Oy}$ attached to a bin packing pattern.

**Output data:** The s-pictural representation of the cutting pattern \[15\] like a formula in a Polish prefixed form.

**Method:** Using a depth-first search method, the algorithm constructs the syntactic tree for the Polish expression representation of the cutting pattern, starting from the root to the leaves (procedure PRORD). For every vertex of the tree it verifies if it is possible to make a guillotine cut by a cutting plane perpendicular on $Oz$ (procedure ZCUT) or perpendicular on $Ox$ (procedure XCUT) or perpendicular on $Oy$ (procedure YCUT), using an algorithm for decomposition of a set $C$ of boxes in two subsets, $B_1$ and $B_2$.

We will use the following notations:
- $G_{Ox}', G_{Oy}', G_{Oz}'$ are the subgraphs of $G_{Ox}|U$, respectively $G_{Oy}|U$ and $G_{Oz}|U$ where we can add, if it is necessary, the root $R_X (R_Y, R_Z)$ and the arcs starting from $R_X (R_Y, R_Z)$.
- $\text{succ}(C_i[G])$ is the set of successors of the box $C_i$ in the graph $G$.

The method ADD() is used for addition of the next member in the Polish prefixed form.

4.0.1 Correctness and Complexity

The correctness of the algorithm follows from the Theorem 7, that makes the connection between a guillotine cut and the decomposition of a graph in two subgraphs.

The procedure PREORD() represents a preorder traversal of a graph, so the complexity is $O(k^2)$, where $k$ is the number of the cutting boxes. Also, in the procedure ZCUT, respectively XCUT and YCUT we traverse a subgraph of the initial graph. So, the complexity of the algorithm is $O(k^2)$.

**PROCEDURE PRORD** ($G, C, L, W, H, ADD()$)

begin

\[ ZCUT(G_{Oz}, C, L, W, H, err, B_1, B_2, H_1, H_2); \]

if $err = 0$ then

if $|C| = 1$ then ADD($C$)
else ADD($C$);
PRORD($G_{Oy}, B_1, L, W, H, ADD()$);
PRORD($G_{Oz}, B_2, L, W, H, ADD()$);

end
else

\[ XCUT(G_{Ox}, C, L, W, H, err, B_1, B_2, W_1, W_2); \]

if $err = 0$ then

if $|C| = 1$ then ADD($C$)
else ADD($C$);
PRORD($G_{Oy}, B_1, L, W, H, ADD()$);
PRORD($G_{Oz}, B_2, L, W, H, ADD()$);

end
else

\[ YCUT(G_{Oy}, C, L, W, H, err, B_1, B_2, L_1, L_2); \]

if $err = 0$ then

if $|C| = 1$ then ADD($C$)
else ADD($C$);
PRORD($G_{Oz}, B_1, L_1, W, H, ADD()$);
PRORD($G_{Oy}, B_2, L_2, W, H, ADD()$);

end
else

No guillotine restrictions

end

end

**PROCEDURE ZCUT** ($G_{Oz}, U, L, W, H, err, B_1, B_2, H_1, H_2$) begin

$err = 0$; SUBGRAPH($G_{Oz}, G_{Oz}', U, R_{Oz}$):

$V := \{C_i| C_i \in U, (R_{Oz}, C_i) \in \Gamma_{Oz}\}$, where all the elements are unmarked

$maxM := \max\{h_i| C_i \in V\}$

$P_i := \{h_i| C_i \in V\}$ while $\exists C_i \in V$ unmarked do

mark $C_i$;

if $P_i < maxM$ then

for $C_j$ in succ($C_i$ in the graph $G_{Oz}'$) do

$V := V \cup \{C_j\}$ where $C_j$ is unmarked;

$P_j := P_i + h_j$;

if $P_j > maxM$ then

$maxM := P_j$;

end

end

$maxM := \max\{Lpd(R_{Oz}, C_i)| C_i \in V\}$

if $maxM = H$ then

$err = 1$;
end
else

$H_1 := maxM; H_2 := H - maxM$;

$B_1 := V; B_2 := U - V$;

end

end
5 Conclusions

Many of the applications of the three dimensional bin-packing problem need packing patterns with guillotine restrictions. So a way of solving this is to use some algorithms for packing patterns determination and to use our algorithm for verifying if the patterns have guillotine restrictions or not. This guillotine test can be used also in a constraint programming approach for solving the packing problem. The test for guillotine restrictions presented in this paper is based to a representation of the bin packing pattern by three weighed graphs of adjacency. These graphs, introduced first for the two-dimensional cutting stock problem, was very useful to prove some properties of a cutting or covering pattern [13, 14] or to find out an order of packing for loading a container[12].

We remark that we can apply this algorithm also in case of a cutting-stock pattern without gaps and, of course, in the case of covering pattern with or without gaps.

Acknowledgements: The research was supported in the case of the first author by the Grant PNII no. 22134/2008.

References: