A test of the guillotine restrictions determination for a Rectangular Three Dimensional Bin Packing Problem

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

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

Abstract: This paper is an extended version of the paper [9] regarding to the rectangular three dimensional bin packing problem, where a bin is loaded with a set of rectangular boxes, without overlapping but with possible gaps. One of the most popular restriction for the solution of the 3D-bin packing problem is the guillotine restriction. The guillotine restriction requires that the packing 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. Our objective is to find a method to verify if a 3D bin-packing pattern has the guillotine constrains or not. For this purpose we use a weighed graph representations of a solution of the problem, the generalisation of this kind of representation obtained by us for 2D cutting-stock problem in [10, 11, 12].

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

1 Introduction

The bin packing problem is one of the well-known combinatorial NP-hard problems in which box-shaped objects of different sizes must be placed into a finite number of bins in a way that minimizes the number of the bins used. In [4] H. Dyckhoff presented many kinds of bin packing problems, 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 holds special and important applications. So are 3D rectangle optimal packing, container packing and container loading optimization, pallet building and truck loading, air cargo load planning software, transportation software, warehouse management systems, package design software, other applications concerning the orthogonal 3D space arrangement, space optimization and volume utilization with rectangular shaped boxes, including odd shaped containers.

In the three-dimensional bin-packing problem each object and bin exists in three dimensions. These objects and bins represent triplets containing three values: width, length, and height. Each box should fit into a bin or bins most efficiently. 3D bin-packing may involve a single bin or multiple bins. The singular bin-packing problem involves only one bin with either definite or infinite volume. Like 2D bin-packing, each box must retain stay orthogonal, or maintain 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 constrains are: gravity, weight distribution or delivery time and so-called guillotine constraint. That means that the packing patterns should be such that the boxes can be obtained by sequential face-to-face cutting plane parallel to a face of the bin.

Many models and algorithms are developed for bin packing problem such as: formulation as a mixed integer program, which can solve the small sized instance to optimum [5], genetic algorithms [6], or approximation algorithms [7, 8]. While an approximation algorithm become a guide that attempts to place objects in the least amount of space and time, a mixed integer program gives solution as the position of the boxes in the bin. If a robot is used for packing the boxes in the bin it is not enough to have a packing pattern but it is necessary to have a plan for packing. This plan means that every new box is positioned in front of, right of, and above the packed boxes.In [13] it is presented a method to determine this kind of order for packing the boxes in the bin. Here we are looking for another problem, the problem of the guillotine constrains. While there are a lot of algorithms for packing patterns determination for the general three dimensional bin packing problem , exact algorithms or approximation algorithms, the general problem with guillotine restrictions is more difficult to solve. For example R.R. Amossen, D. Pisinger [1] solve a general packing problem, where in each step they test for satisfaction of guillotine constraints. Using some graph representations defined by us in [13], we present now another guillotine test for the threee dimensional bin packing patterns, by generaling the results obtained in [10, 11] for the two dimensional cutting-stock patterns. This test is based on theoretical results from [9].

2 Problem formulation

We consider a 3D bin-packing problem where a three dimensional rectangular bin $\mathcal{B}$, a container with length L, width W, height H is filled with k rectangular boxes $C_1, C_2, \ldots C_k$ without overlapping but with possible gaps. 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 $\mathcal{B}$, so that the faces of the boxes C_i be parallel with the faces of the container $\mathcal{B}$.*

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 [15, 14, 13] 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 [10, 11].

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

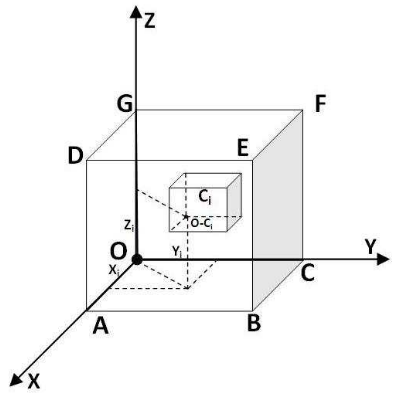

Figure 1: The position of box C_i in the bin

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 [13] 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 $\mathcal{B}$ (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.

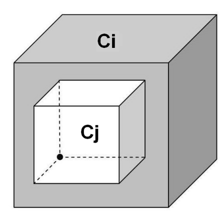

Figure 2: The adjacency of C_i with C_j in Ox direction

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.

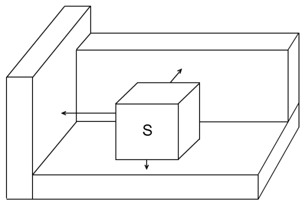

Figure 3: The moving directions

Starting from the three kinds of adjacency we have defined in [13] 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*

$$\begin{cases} \Gamma_{Ox}(C_i) \ni C_j \text{ only if } C_i \text{ is adjacent in} \\ \qquad\qquad \text{direction } Ox \text{ with } C_j \\ \Gamma_{Ox}(X) \ni C_i \text{ only if the North face of } C_i \\ \qquad\qquad \text{touches the } yOz \text{ plan} \\ Value(U, C_j) = w_j, \forall U \in C \cup R_X \text{ and } C_j \in C \end{cases}$$

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 [15, 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 $(1, 1.5, 2)$

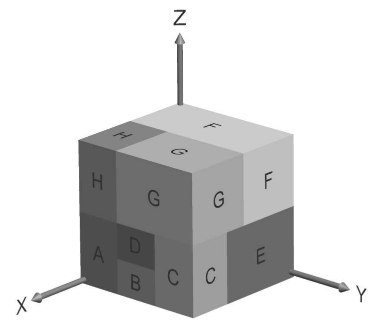

Figure 4: A pattern view from the top-right-front corner for Example 1.

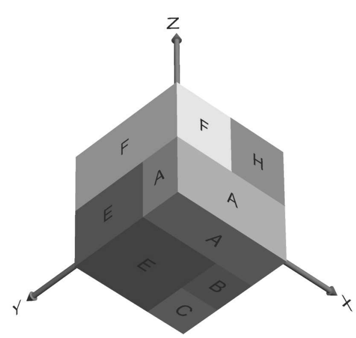

Figure 5: A pattern view from the bottom-left-back corner for Example 1.

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

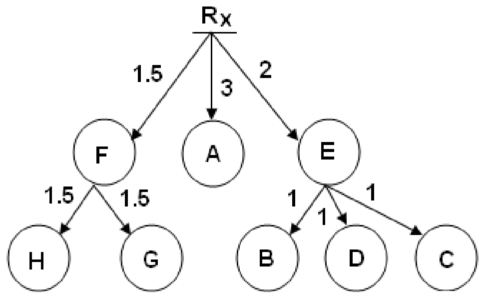

Figure 6: Graph G_{Ox} for Example 1.

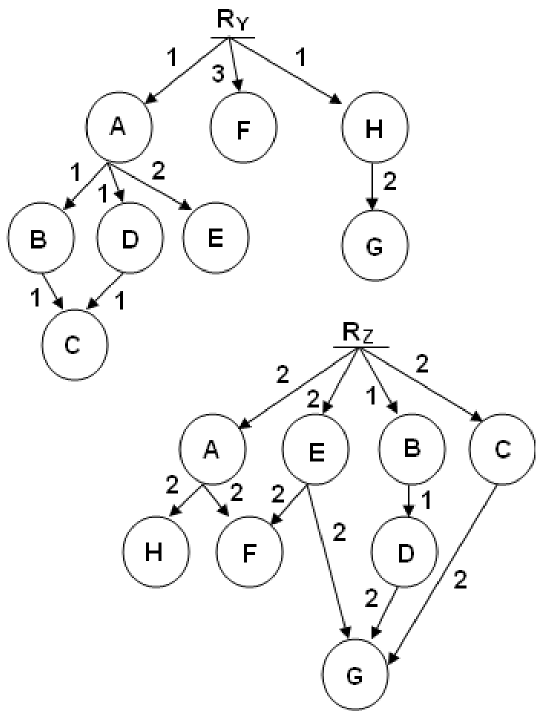

Figure 7: Graphs G_{Oy}, G_{Oz} for Example 1.

[16, 17] 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:

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

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 [12] and one for the pattern with gaps [10, 11].

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 [10, 11] 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 $Lpd(R_X, C_i)$ for the length of the path from R_X to C_i in the graph G_{Ox}. Similarly we will use the notations $Lpr(R_Y, C_i)$ for the length of the path from R_Y to C_i in the graph G_{Oy}, respectively $Lpr(R_Z, C_i)$ for the length of the path from R_Z to C_i in the graph G_{Oz}. We remark that $Lpd(R_X, C_i)$ represents the distance from the northern face of the bin *B* to the southern face of box C_i, $Lpr(R_Y, C_i)$ represents the distance from the western face of the bin *B* to the eastern face of box C_i and $Lpr(R_Z, C_i)$ represents the distance from the bottom face of the bin to the top face of bin 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. $Lpd(R_Z, C_i) \leq M$ *for every* $C_i \in B_1$;

2. $Lpd(R_Z, C_i) > M$ *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 $C_j \in C$ so that $(R_Z, C_j) \in \Gamma_z$ it follows that $C_j \in B_1$;*

3. *$Lpd(R_Z, C_i) \leq M$ for every $C_i \in B_1$;*

4. *If there is $C_j \in B_1$ so that $Lpd(R_Z, C_j) < M$ 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 $Lpd(R_Z, C_j) < M$ the item C_i is a direct successor of C_j and suppose that $C_i \in B_2$. It follows that $Lpd(R_Z, C_i) > M$ 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 $Lpd(R_Z, C_j) < M$ and so $C_j \in B_1$. From condition 4, because C_i is a direct successor of C_j, it follows that C_i must be in B_1 in contradiction with our hypothesis. That means that if the conditions 1-4 are fulfilled then there is a horizontal guillotine cutting plane in the bin-packing pattern. □

We obtain a similar result if we consider the weighed graphs of adjacency in the directions Ox or Oy.

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} or G_{Oz} attached to a bin packing pattern.

Output data: The s-pictural representation of the cutting pattern [16] 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).

- $succ(C_i|_G)$ 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.

The procedures ZCUT, YCUT (analogue XCUT) are presented below:

4.1 Example

Let us have the covering pattern from Figures 4 and 5, with the weighed graphs G_{Ox}, G_{Oy}, G_{Oz} from Figure 6 and Figure 7. By examination of the weighed graphs we observe that it is posible to make the first horizontal cut by a cutting plane perpendicular on Oz, procedure ZCUT. In Figure 8 it is presented this first horizontal cut of the packing pattern that separate the set of the boxes in two components, one set $\{A, E, B, C, D\}$ and the other set $\{H, F, G\}$. It can be seen that on the graph G_{Oz} we did the cut at distance 2. In the syntactic tree, these components are connected using the horizontal concatenation $\oplus$, an operation for a horizontal cutting plane, Figure 9.

The prefix polish notation for this syntactic tree from Figure 9 is: $\oplus$.

We continue to make horizontal, vertical N-S or W-E cuts for the left and right components from the syntactic tree until every components will contain

PROCEDURE PRORD$(G, C, L, W, H, ADD())$
begin
  ZCUT$(G_{Oz}, C, L, W, H, err, B_l, B_2, H_1, H_2)$;
  if $err = 0$ **then**
    if $|C| = 1$ **then ADD**(C)
    else ADD$(\oplus)$;
    PRORD$(G_{Ox}, B_1, L, W, H_1, ADD())$;
    PRORD$(G_{Ox}, B_2, L, W, H_2, ADD())$;
  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$(\ominus)$;
      PRORD$(G_{Oy}, B_1, L, W_1, H, ADD())$;
      PRORD$(G_{Oy}, B_2, L, W_2, H, ADD())$;
    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$(\oslash)$;
        PRORD$(G_{Oz}, B_1, L_1, W, H, ADD())$;
        PRORD$(G_{Oz}, B_2, L_2, W, H, ADD())$;
      end
      else No guillotine restrictions
    end
  end
end

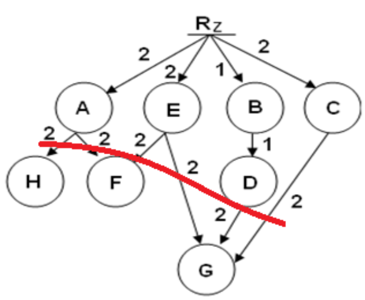

Figure 8: The first vertical cut of the bin-packing pattern

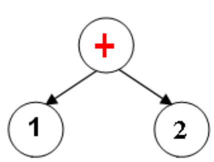

Figure 9: The first syntactic tree

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 $:= \bigcup\{C_i | C_i \in U, (R_{Oz}, C_i) \in \Gamma_{Oz}\}$, 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**
        $V := V \bigcup\{C_j|$ where C_j is unmarked$\}$;
        $P_j := P_i + h_j$;
        if $P_j > maxM$ **then**
          $maxM := P_j$;
        end
      end
    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

PROCEDURE
YCUT$(G_{Oy}, U, L, W, H, err, B_1, B_2, L_1, L_2)$ **begin**
  $err = 0$; **SUBGRAPH**$(G_{Oy}, G'_{Oy}, U, R_{Oy})$;
  V $:= \bigcup\{C_i | C_i \in U, (R_{Oy}, C_i) \in \Gamma_{Oy}\}$, where all the elements are unmarked
  maxM $:= max\{l_i | C_i \in V\}$
  $P_i := \{l_i | C_i \in V\}$ **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'_{Oy}) **do**
        $V := V \bigcup\{C_j|$ where C_j is unmarked$\}$;
        $P_j := P_i + l_j$;
        if $P_j > maxM$ **then**
          $maxM := P_j$;
        end
      end
    end
  end
  maxM $:= max\{Lpd(R_{Oy}, C_i) | C_i \in V\}$
  if $maxM = L$ **then**
    $err = 1$;
  end
  else
    $L_1 := maxM$; $L_2 := L - maxM$; $B_1 := V$;
    $B_2 := U - V$;
  end
end

only one item from the covering model, considering the extracted subgraphs.

Now considering the component 1, we have the three subgraphs $R1_X(A, E, B, D, C)$, $R1_Y(A, E, B, D, C)$ and $R1_Z(A, E, B, D, C)$ from Figure 10 and Figure 11, obtained by extracting only the nodes from first set, $\{A, E, B, D, C\}$.

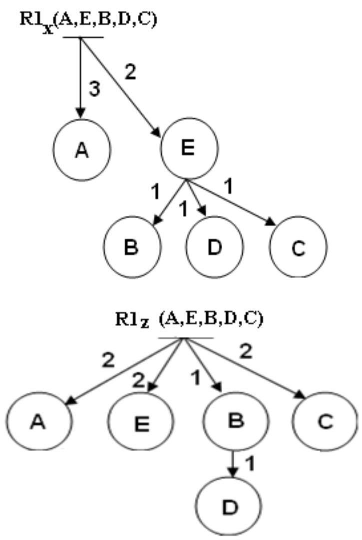

Figure 10: The subgraphs $R1_X(A, E, B, D, C)$ and $R1_Z(A, E, B, D, C)$ derived from the first set

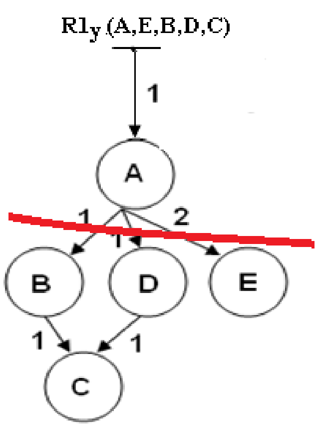

Figure 11: The subgraph $R1_Y(A, E, B, D, C)$ derived from the first set

Applying the procedure YCUT we observe that we can make a vertical W-E cut on $R1_Y(A, E, B, C, D)$, by a cutting plane perpendicular on Oy, at distance 1 from top of the subgraph, Figure 11. We extract two sets again, one containing just $\{A\}$ and another $\{E, B, C, D\}$, named component 3. In the syntactic tree from Figure 12 we see that the operation between box A and this new component, 3, is $\oslash$ the notation for a cutting plane perpendicular on O_y.

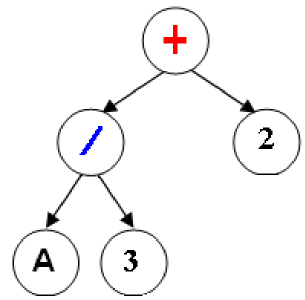

Figure 12: The syntactic tree. Step 2

The prefix polish notation for this syntactic tree from Figure 12 is: $\oplus \oslash A$.

We continue our algorithm using component 3, by extracting the three subgraphs $R3_Y(E, B, C, D)$, $R3_Z(E, B, C, D)$ and $R3_X(E, B, C, D)$, see Figure 13 and Figure 14.

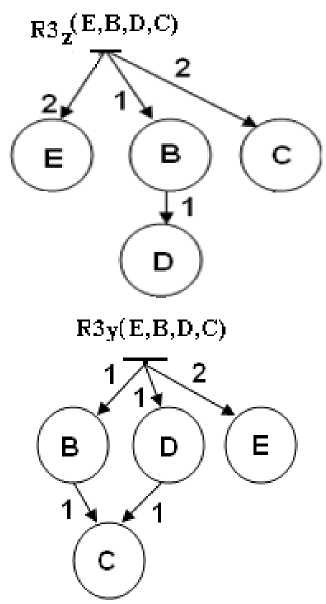

Figure 13: The subgraphs $R3_Y(E, B, D, C)$ and $R3_Z(E, B, D, C)$ derived from the third set

Now we use the procedure XCUT that makes a vertical N-S cut on $R_3X(A, E, B, C, D)$, at distance 2 from top of the subgraph, see Figure 14. We extract two sets, one containing just $\{E\}$ and another $\{B, C, D\}$, named component 4. In the syntactic tree from Figure 15 we have the operation $\ominus$ for a vertical cutting plane perpendicular on O_x between E and component 4.

The prefix polish notation for this syntactic tree from Figure 15 is: $\oplus \oslash A \ominus E$.

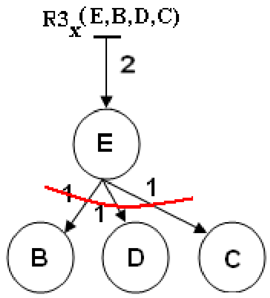

Figure 14: The subgraph $R3_X(E, B, D, C)$ derived from the third set

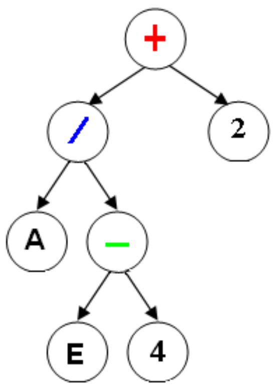

Figure 15: The syntactic tree. Step 3

Using component 4, we extract the three subgraphs $R4_X(B, C, D)$, $R4_Z(B, C, D)$ and $R4_Y(B, C, D)$, see Figure 16 and Figure 17.

We can make a cut on $R_4Y(B, C, D)$, at distance 1 from top of the subgraph, by a cutting plane perpendicular on Oy, see Figure 17. We extract two sets, one containing just $\{C\}$ and another $\{B, D\}$, named component 5. In the syntactic tree from Figure 18 we have the operation for a vertical cutting plane perpendicular on O_y between C and component 5.

Making the same steps with component 5, we obtained a vertical cutting plane perpendicular on O_z and the syntactic tree from Figure 19.

The prefix polish notation for this syntactic tree from Figure 19 is: $\oplus \oslash A \ominus E \oslash \oplus BDC$.

Let's turn to the set number 2, that one composed from $\{H, F, G\}$. We have the subgraphs from Figure 20 and Figure 21 and we've done a cut on $R2_X(F, H, G)$, at the distance 1.5, by a cutting plane perpendicular on Ox. The 2 sets are: $\{F\}$ and $\{H, G\}$.

The syntactic tree from Figure 22 has the component 6 connected with F thru a W-E cut.

Using component 6, we have the last three subgraphs $R6_X(H, G)$, $R6_Z(H, G)$ and $R6_Y(H, G)$.

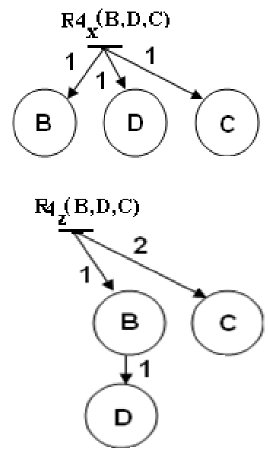

Figure 16: The subgraphs $R4_X(B, C, D)$ and $R4_Z(B, C, D)$ derived from the fourth set

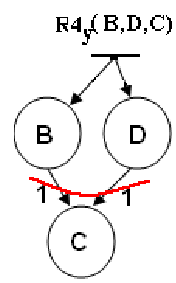

Figure 17: The subgraph $R4_Y(B, C, D)$ derived from the fourth set

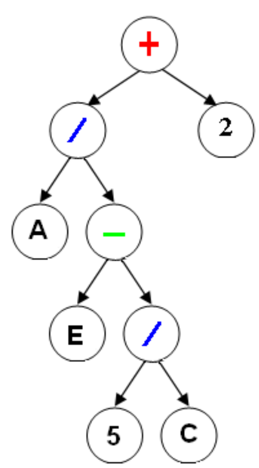

Figure 18: The syntactic tree. Step 4

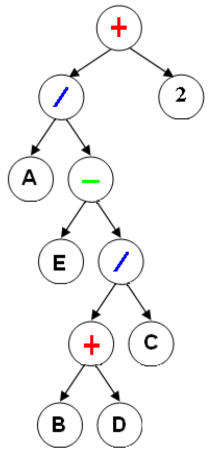

Figure 19: The syntactic tree. Step 5

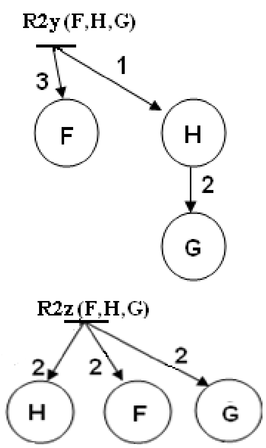

Figure 20: The subgraphs $R2_Y(F,H,G)$ and $R2_Z(F,H,G)$ derived from the second set

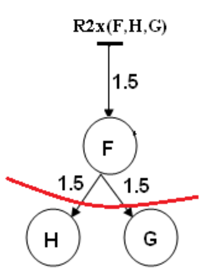

Figure 21: The subgraph $R2_X(F,H,G)$ derived from the second set

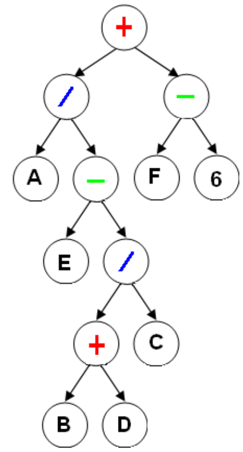

Figure 22: The syntactic tree. Step 6

The final syntactic tree is in Figure 23.

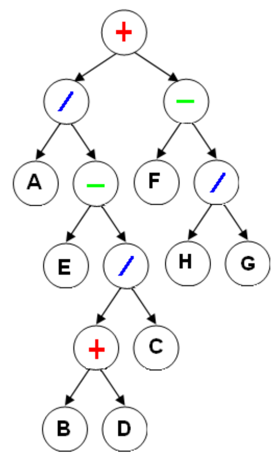

Figure 23: The final syntactic tree

This syntactic tree corresponds to the prefix polish notation:

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

exactly the one that we considered in previous section.

4.2 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, 3], where k is the number of the packed 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 $O(k^2)$.

5 Conclusions

The three dimensional bin-packing problem holds importance to many fields. Shipping and moving industries, architecture, engineering and design are all areas where three dimensional bin-packing could apply. Industry uses bin-packing for everything from scheduling television programming to stacking cargo in a semi-truck to designing automobiles and airplanes. 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 [15, 14] or to find out an order of packing for loading a container[13].

We remark that we can apply this algorithm for guillotine determination 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:

[1] R.R. Amossen, D. Pisinger, Multi-dimensional Bin Packing Problems with Guillotine Constraints, *Computers & OR* 37, No. 11, 1999-2006 (2010).

[2] E. Ciurea and L. Ciupala, *Algoritmi - Introducere în algoritmica fluxurilor în reţele*, Matrix ROM Bucuresti,2006.

[3] T.H. Cormen, C.E. Leiserson and R.R. Rievst, *Introduction to algorithms*, MIT Press, 1990.

[4] H. Dyckhoff, A typology of cutting and packing problems, *European Journal of Operational Research* 44 (1990) 145-159.

[5] Z. Jin and T. Ito, The three Dimensional Bin Packing Problem and its Practical Algorithm, *JSME International Journal*, Series , Vol. 46, No.1, 2003.

[6] J.E. Lewis, R.K. Ragade, A. Kumar and W.E. Biles, A distributed chromosome genetic algorithm for bin-packing, *14th International Conference on Flexible Automation and Intelligent Manufacturing, Robotics and Computer-Integrated Manufacturing*, Vol. 21, Issues 4-5, August-October 2005, Pp. 486-495.

[7] A. Lodi, S. Martello and D. Vigo, TSpack A Unified Tabu Search Code for Multi-Dimensional Bin Packing Problems, *Annals of Operations Research*,Vol. 131, No. 1-4, 2004, pp.203-213,

[8] W.F. Maarouf, A.M. Barbar and M.J. Owayjan, A New Heuristic Algorithm for the 3D Bin Packing Problem, *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, Springer, 2008, pp. 342-345.

[9] D. Marinescu, A. Băicoianu and D. Simian, The determination of the guillotine restrictions for a Rectangular Three Dimensional Bin Packing Pattern, *Recent Researches in ComputerScience, Proc. of the 15-th WSEAS International Conference on Computers CSCC*, Corfu, 2011, Vol. 1, pp. 491-496.

[10] D. Marinescu and A. Băicoianu, The determination of the guillotine restrictions for a rectangular cutting-stock pattern, *Latest Trends on Computers, Proc. of the 14-th WSEAS International Conference on Computers CSCC*, ISSN: 1792-4251, Corfu, 2010, Vol. 1, pp. 121-126.

[11] D. Marinescu and A. Băicoianu, An algorithm for the guillotine restrictions verification in a rectangular cutting-stock pattern, *WSEAS Transactions on Computers*, Volume 9 Issue 10, Oct. 2010, pp. 1160-1169.

[12] D. Marinescu and A. Băicoianu, An Algorithm for the guillotine restrictions verification in a rectangular Covering Model, *WSEAS Transactions on Computers*, ISSN: 1109-2750, Issue 8, Vol. 8, Aug. 2009, pp. 1306-1316.

[13] D. Marinescu, P. Iacob and A. Băicoianu, A topological order for a rectangular three dimensional bin packing problem, *New aspects of Computers, Proc. of the 12-th WSEAS International Conference of COMPUTERS*, Herakion, Crete, 2008, Part. I, pp. 285-290.

[14] D. Marinescu, A representation problem for a rectangular cutting - stock model, *Foundations of Computing and Decision Sciences*, Vol. 32, 2007, No. 3, pp. 239-250.

[15] D. Marinescu, Properties of the matrices attached to a rectangular cutting-stock model (French), *Buletin of the Transilvania University of Braşov*, seria C, Vol XXXIV,1992, pp 41-48.

[16] D. Marinescu, A s-picture language for a cutting-stock model with quillotine restrictions, *Buletin of the Transilvania University of Brasov* - seria C, Vol XXXIII 1991 pp 39-45.

[17] D. Marinescu, A pictural language for a cutting-stock model with guillotine restrictions, *Proc. of the First National Colocgue for Languages, Logic and Mathematical Linguistic*, Braşov, 1986, pp. 117 - 125.