ILP Models for the 2-staged Cutting Stock

Andrea Lodi and Michele Monaci

Dipartimento di Elettronica, Informatica e Sistemistica
University of Bologna
Viale Risorgimento, 2 - 40136 - Bologna, Italy
Email: {alodi,mmonaci}@deis.unibo.it

ABSTRACT: We are given a unique rectangular stock of material $S$, with height $H$ and width $W$, and a list of $m$ rectangular shapes to be cut from $S$. Each shape’s type $i$ ($i = 1, \ldots, m$) is characterized by a height $h_i$, a width $w_i$, a profit $p_i$, and an upper bound $ub_i$ indicating the maximum number of items of type $i$ which can be cut. We refer to Two-Dimensional Cutting stock (TDC) as the problem of determining a cutting pattern of $S$ maximizing the sum of the profits of the cut items. In particular, we consider the classical variant of TDC in which the maximum number of cuts allowed to obtain each item is fixed to 2, and we refer to this problem as 2-staged TDC (2TDC). For 2TDC we present two new Integer Linear Programming models, we discuss their properties, and we reinforce them by the addition of linear inequalities which avoid symmetries. Finally, both models are computationally tested on a large set of instances from the literature.

KEYWORDS: Packing, Cutting, Integer Linear Programming

1. INTRODUCTION

The problem of cutting a given set of small rectangles (items) from large identical stock rectangles of material has been regarded as a prototypical problem in the field of Cutting & Packing ever since the well-known seminal work of (Gilmore and Gomory, 1965). In this paper the authors discussed a large variety of multi-dimensional cutting problems, moving from the definition of the Two-Dimensional Cutting Stock Problem in which the objective function is to minimize the number of large rectangles used. The column generation approach proposed by (Gilmore and Gomory, 1961) was extended to the two dimensional case in (Gilmore and Gomory, 1965), and each slave problem is as follows: a profit is associated to each item, and a unique large rectangle has to be cut so as to obtain a subset of the items whose sum of the profits is a maximum. The authors referred to the latter problem as the Cutting Knapsack Problem to emphasize that a unique stock rectangle and a set of profits are considered. However, the terminology has been often confused referring to both problems in the same way, while now most of the authors refer to former problem as Two-Dimensional Bin Packing (see (Lodi et al., 1999) for a recent survey), and to the latter as Two-Dimensional Cutting stock (TDC).

More formally, in TDC we are given a unique rectangular stock of material $S$, with height $H$ and width $W$, and a list of $m$ rectangular shapes to be cut from $S$. Each shape’s type $i$ ($i = 1, \ldots, m$) is characterized by a height $h_i$, a width $w_i$, a profit $p_i$, and an upper bound $ub_i$ indicating the maximum number of items of type $i$ which can be cut. The problem calls for the determination of a cutting pattern of $S$ maximizing the sum of the profits of the cut items (see Figure 1).

TDC can be found in the literature in many variants deriving from additional requirements or extensions. One of the most common of these variants is determined by the requirement of producing cutting patterns of guillotine type, i.e., in which each item must be cut with a sequence of edge to edge cuts parallel to the edges of $S$ (see Figure 1(b)). A special case of this class of problems is the so called $d$-staged (Two-dimensional) Cutting Stock, in which the maximum number of guillotine cuts allowed to obtain each item is fixed to $d$. This latter class of problems was introduced by (Gilmore and Gomory, 1965), and has received considerable attention due to relevant real-world applications.

In this paper we consider the case of $d$-staged Cutting Stock with $d = 2$ (see Figure 2), and we denote it as 2-staged Cutting Stock (2TDC). This case has been addressed in the literature by several authors, and both exact and heuristic algorithms have been proposed for some different variants of it. In particular, the following problems have to be considered. (1) 2TDC is said to be Un-
constrained (U-2TDC) if there is no limit to the number of items of each type which can be cut off; otherwise, the problem is said to be Constrained (C-2TDC). (2) if a 90°-degree Rotation of the items is allowed, the problem is referred to as R-2TDC; otherwise, we consider the orientation of the items to be Fixed (F-2TDC). (3) if a third stage of cutting is allowed to separate an item from a waste area, we call this the non-exact case of 2TDC or 2TDC with trimming (see Figure 2(a)), otherwise, we have the exact case of 2TDC, or 2TDC without trimming (see, Figure 2(b)). (4) 2TDC is said to be unweighted if \( \bar{p}_i = \bar{h}_i \bar{w}_i \) \( (i = 1, \ldots, m) \); otherwise the problem is said to be weighted.

Figure 2. Examples of 2-staged patterns: non-exact (a) and exact (b) cases.

Extensive studies on 2TDC problems have been performed by (Hifi, 1999). In particular, (Morabito and Garcia, 1998) and (Hifi and Zissimopoulos, 1996) adapted and extended the original approach of Gilmore and Gomory to solve many of the variants of 2TDC discussed above. Recently, (Hifi and Roucairol, 2000) proposed both exact and heuristic algorithms for the specific case FC-2TDC.

The rest of the paper is organized as follows: in Section 2, we propose two Integer Linear Programming (ILP) models for 2TDC, considering the Fixed Constrained version of the problem. In Section 3, linear inequalities aimed at avoiding possible symmetries are introduced, and finally, in Section 4, the models are computationally tested on a set of instances from the literature and compared with (Hifi and Roucairol, 2000). In the extended version of this paper (Lodi and Monaci, 2000), the results on these instances are discussed in details, and the extension of the models to the other variants of 2TDC are presented along with additional computational results on each of them. In the following, without loss of generality, we assume that all input data are positive integers.

2. ILP MODELS FOR FC-2TDC

Many of the classical heuristic results on Two-Dimensional Packing Problems are obtained by considering the restriction of packing (cutting) the items into shelves, i.e., rows forming levels. More precisely, a shelf is a slice of the stock rectangle with width \( W \), and height coincident with the height of the tallest item cut off from it. In addition, each item cut off from the shelf has its bottom edge on a line, the base of the shelf, and the top of the shelf determines the base of a following shelf (see, e.g., the slice containing items 4, 5, 6, 7 in Figure 2(a)).

The following simple observation holds.

**Observation 1** Each feasible solution of 2TDC with trimming is composed of shelves, and, vice-versa, each item packed into a shelf can be cut off in at most two stages.

Recently (Lodi et al., 2000) and (Lodi, 2000) introduced new models for Two-Dimensional Packing problems in which the restriction of packing into shelves is explicitly considered. In the light of the previous observation, some of the results and the terminology introduced in the works above can be used for 2TDC. In particular, it is also true for 2TDC that for any optimal solution there exists an equivalent solution in which the first (leftmost) item cut in each shelf is the tallest item of the shelf (see again Figure 2). This allows us to consider only solutions which satisfy this condition, and this first item is said to initialize the shelf.

2.1. Model 1

For the first model we consider each item to be distinct, i.e., for each shape’s type \( i (i = 1, \ldots, m) \), we define \( u_{bi} \) identical items \( j \) such that \( h_j = \bar{h}_i \), \( w_j = \bar{w}_i \), and \( p_j = \bar{p}_i \). Let \( n = \sum_{i=1}^{m} u_{bi} \) indicate the overall number of items, and consider the items ordered in such a way that \( h_1 \geq h_2 \geq \ldots \geq h_n \). The model assumes that \( n \) potential shelves may be initialized, each one by its corresponding item. Then, the possible cutting of the \( n \) items from the potential shelves is described by the following binary variables:

\[
 x_{jk} = \begin{cases} 
 1 & \text{if item } j \text{ is cut from shelf } k \\
 0 & \text{otherwise} 
\end{cases} \quad (1) 
\]

where \( k = 1, \ldots, n \) and \( j = k, \ldots, n \).

The model is then as follows:

\[
 \text{Mod1(2TDC)} \quad \max \sum_{j=1}^{n} p_j \sum_{k=1}^{j} x_{jk} \quad (2) 
\]

subject to

\[
 \sum_{k=1}^{j} x_{jk} \leq 1 (j = 1, \ldots, n) \quad (3) 
\]

\[
 \sum_{j=k+1}^{n} w_j x_{jk} \leq (W - w_k) x_{kh} (k = 1, \ldots, n - 1) \quad (4) 
\]

\[
 \sum_{k=1}^{n} h_k x_{kh} \leq H \quad (5) 
\]

\[
 x_{jk} \in \{0, 1\} (k = 1, \ldots, n; \ j = k, \ldots, n) \quad (6) 
\]
The objective function (2) maximizes the sum of the profits of the cut items. Inequalities (3) guarantee that each item is cut at most once, and only from shelves whose height is at least equal to the height of the item. Inequalities (4) assure that the width constraint for each shelf is satisfied, whereas inequality (5) imposes the height constraint. Note that the meaning of each variable $x_{ik}$ ($k = 1, \ldots, n$) is twofold: $x_{ik} = 1$ implies that item $k$ is cut from shelf $i$, i.e., shelf $k$ is used and initialized by its corresponding item.

### 2.2. Model 2

In the second model the decomposition of the sets of shapes into single items is done only in terms of shelves, i.e., we consider the items with the same shape’s type together, whereas we separate them with respect to the initialization of the shelves. Hence, we need to define a mapping between shape’s types $i$ ($i = 1, \ldots, m$), and potential shelves $k$ ($k = 1, \ldots, n$). In fact, any item of type $i$ may be cut from shelves in the range $[1, \sum_{j=1}^{i} u_{b_j}]$, and we define $\alpha_i = \sum_{j=1}^{i} u_{b_j}$ ($i = 1, \ldots, m$) with $\alpha_0 = 0$. On the other hand, any shelf $k$ can be used to obtain items whose type is in the range $[\beta_k, m]$, with $\beta_k = \min\{r : 1 \leq r \leq m, \alpha_r \geq k\}$ ($k = 1, \ldots, n$). Hence, by assuming again $\tilde{h}_1 \geq \tilde{h}_2 \geq \ldots \geq \tilde{h}_m$, we have two separate sets of variables. The first set is composed of the following integer (nonbinary) variables:

$$x_{ik} = \begin{cases} 
\text{number of items of type } i \text{ cut from shelf } k, & \text{if } i \neq \beta_k \\
\text{number of additional items of type } i \text{ cut from shelf } k, & \text{if } i = \beta_k
\end{cases}$$

(7)

where $i = 1, \ldots, m; k \in [1, \alpha_i]$, and the term “additional” indicates that the item of type $i$ initializing shelf $k$ is separately considered (if the shelf corresponds to this type of items).

The second set involves the following binary variables:

$$q_k = \begin{cases} 
1 & \text{if shelf } k \text{ is used} \\
0 & \text{otherwise}
\end{cases} \quad (k = 1, \ldots, n)$$

(8)

The model is then as follows:

$$\text{Mod2(2TDC) } \max \sum_{i=1}^{m} p_i \left( \sum_{k=1}^{\alpha_i} x_{ik} + \sum_{k=\alpha_i+1}^{\alpha_i} q_k \right) \quad (9)$$

subject to

$$\sum_{k=1}^{\alpha_i} x_{ik} + \sum_{k=\alpha_i+1}^{\alpha_i} q_k \leq u_{b_i} (i = 1, \ldots, m) \quad (10)$$

$$\sum_{i=\beta_k}^{m} \bar{\overline{w}}_{i} x_{ik} \leq (W - \bar{\overline{w}}_{\beta_k})q_k (k = 1, \ldots, n) \quad (11)$$

$$\sum_{k=1}^{n} \tilde{h}_{\beta_k} q_k \leq H \quad (12)$$

$$0 \leq x_{ik} \leq u_{b_i} \quad \text{integer} \quad (i = 1, \ldots, m; k \in [1, \alpha_i]) \quad (13)$$

$$q_k \in \{0, 1\} (k = 1, \ldots, n) \quad (14)$$

The objective function (9) corresponds to the one of Model 1, so as inequalities (10), (11), and (12) which impose the cardinality constraints, the width constraints, and the height constraint, respectively.

It is quite easy to see that $q_k$ variables of Model 2 correspond to $x_{ik}$ variables ($k = 1, \ldots, n$) in Model 1, whereas each variable $x_{ik}$ of Model 2 represents the union of the set of $x_{ik}$ variables where $\ell = k + 1, \ldots, \alpha_i$. In other words, in Model 2 the items of a single type are considered together in a shelf, apart from possibly the one initializing the shelf.

### 2.3. Models’ Comparison

It is immediate that Mod1(2TDC) involves $n(n+1)/2$ binary variables and $2n$ constraints, whereas Mod2(2TDC) involves $n + m + 1$ constraints, $n$ binary variables, and $\sum_{i=1}^{m} \sum_{j=1}^{i} u_{b_j}$ integer variables. This means that the number of integer variables of Mod2(2TDC) depends on the structure of the instance, and, in particular, belongs to the range $[n, n(n+1)/2]$, where the lower bound of $n$ variables corresponds to the case in which all the items are identical ($m = 1$), whereas the upper bound is given by the case in which all items are different ($m = n$).

Mod1(2TDC) and Mod2(2TDC) are also different in terms of LP relaxations: each time an item must initialize a shelf, the LP relaxation of Mod2(2TDC) is allowed to split the item into one part initializing the shelf (q part) and some others which can be packed as “additional” parts (x parts) in the same shelf or in the following shelves of that type (if any). Hence, the profit of the item is possibly taken into account completely (see the objective function (9)), while the height of the corresponding shelf is only partially paid (see constraint (12)). This is not the case for Mod1(2TDC), since if an item initializes a shelf it cannot be packed as “additional” neither in it nor in the following shelves.

An example of this behavior is pointed out by the following very simple instance.

**Example 1:** $H = W = 10$, $m = 2$, $\tilde{h}_1 = 10$, $\bar{w}_1 = 1$, $\tilde{h}_2 = 1000$, $u_{b_1} = 1$, $\bar{w}_2 = 1$, $u_{b_2} = 100$. It is easy to see that the LP relaxation of Mod1(2TDC) gives an upper bound $u_{\text{Mod1}} = 1009$, whereas for Mod2(2TDC) $u_{\text{Mod2}} = 1090$. In fact the first model, in order to “gain” the entire profit of the only item of shape’s type 1, must open a shelf of height $h = 10$ which becomes the unique shelf and can accommodate 9 items of type 2. Instead, for the second model it is enough to open the shelf with $q_{11} = 0.1$ and take $x_{11} = 0.9$: the entire profit is gained, but the shelf is paid with height $h = \tilde{h}_{11} = 1$, and it is possible to open 9 other shelves and fill them completely.

The discussed problem can be eliminated by strengthening the bound on the $x_{ik}$ variables for which $i = \beta_k$:...
this can be achieved by considering, as explicitly done in Mod1(2TDC), the correspondence between each item with a shelf. Then, we obtain the following valid inequalities:

\[ x_{ik} \leq u_{bi} - (k - \alpha_{i-1}) \]

\[ \forall i = 1, \ldots, m, \ k \in [\alpha_{i-1} + 1, \alpha_i] \quad (15) \]

which will be indicated in the following as **Bound Inequalities** (BIs). These inequalities reconstruct the mapping items - shelves by stating that for each shape’s type, the \( k \)-th of the \( u_{bi} \) potential shelves can accommodate at most \( u_{bi} - k \) additional items of type \( i \).

It is easy to see that BIs strengthen the second formulation, so that the LP relaxation of Mod2(2TDC) is equal, for the instance presented in Example 1, to the upper bound given by Mod1(2TDC). However, these inequalities are enough only for the case in which the cardinality of shape’s type 1 in the example is at most 2. If we consider \( u_{b1} = 3 \) in Example 1, then we obtain \( u_{Mod1} = 3007 \), while \( u_{Mod2+} = 3070 \) (corresponding to the feasible solution \( q_1 = 0.1, x_{11} = 0.9, q_2 = 0.2, x_{12} = 1.8 \) plus 7 shelves full of items of shape’s type 2), where \( u_{Mod2+} \) indicates the upper bound obtained by adding BIs to Mod2(2TDC).

A further improvement of Mod2(2TDC) can be obtained by lifting BIs as follows. For any shape’s type \( i \), the set of additional items of type \( i \) which can be cut from the shelf \( k \) \( (k \in [\alpha_{i-1} + 1, \alpha_i], \ i.e., \ i = \beta_k) \) cannot include items \( \alpha_{i-1} + 1, \ldots, k \). The following inequalities, say **Lifted Bound Inequalities** (LBIs),

\[ \sum_{s=1}^{\alpha_i} x_{is} \leq u_{bi} - (k - \alpha_{i-1}) \]

\[ \forall i = 1, \ldots, m, \ \forall k \in [\alpha_{i-1} + 1, \alpha_i] \quad (16) \]

are then valid for Mod2(2TDC), and clearly dominate the corresponding BIs by the lifting operation.

By denoting with Mod2^+(2TDC) the second model with the addition of LBIs (which replace the BIs), the following theorem holds.

**Theorem 1** *Mod1(2TDC) and Mod2^+(2TDC) are equivalent in terms of continuous relaxation.*

**Proof.** It is enough to show that to each solution of the continuous relaxation of Mod1(2TDC) (\( M_{1c} \) in the following) corresponds a solution of the continuous relaxation of Mod2^+(2TDC) (\( M_{2c} \) in the following) with the same value, and vice-versa.

Mod1(2TDC) \( \rightarrow \) Mod2^+(2TDC). Given a feasible solution of \( M_{1c} \), (with \( x \) variables denoted as \( x^* \)) a corresponding solution of \( M_{2c} \) is built as follows:

\[ \text{for } k := 1 \text{ to } n \quad q_k = x^*_{kk}; \]

\[ \text{for } i := 1 \text{ to } m \]

\[ \text{for } k := 1 \text{ to } \alpha_i \]

\[ \text{if } (i \neq \beta_k) \text{ then } x_{ik} = \sum_{j=\alpha_{i-1}+1}^{\alpha_i} x^*_j \]

\[ \text{else } x_{ik} = \sum_{j=k+1}^{\alpha_i} x^*_j \]

and this solution, with exactly the same value, immediately satisfies LBIs.

Mod2^+(2TDC) \( \rightarrow \) Mod1(2TDC). Given a feasible solution of \( M_{2c} \) (with \( x \) and \( q \) variables denoted as \( x^* \) and \( q^* \)) a corresponding solution of \( M_{1c} \) is built as follows:

\[ \text{for } k := 1 \text{ to } n \]

\[ x_{kk} = q^*_k; \]

\[ \text{for } j := k+1 \text{ to } n \]

\[ z_j = x^*_j \]

\[ \text{for } j := k+1 \text{ to } n \]

\[ x_{jk} = \min \{1, z_{\beta_j}\} \]

\[ z_{\beta_j} = z_{\beta_j} - x_{jk} \]

The solution obtained with the algorithm above is feasible for \( M_{1c} \) since LBIs assure that, given a shelf \( k \) such that \( i = \beta_k \) \( (k = 1, \ldots, n) \), no item \( j \) of shape’s type \( i \) can be cut off from \( k \) if \( j < k \), and it has the same value.

A final remark concerns the addition of LBIs to the second model in order to strengthen the bound. This strengthening is not only theoretical, but it strongly improves the model in practice. Hence, in the rest of the paper we consider Mod2^+(2TDC).

### 3. AVOIDING SYMMETRIES BY LINEAR INEQUALITIES

Symmetries are one of the main problems when dealing with exact methods for cutting problems. Some of these symmetries are avoided by the assumptions discussed at the beginning of Section 2.: we do not explicitly distinguish the position of an item in a shelf (neither the one of a shelf in the stock), and the fact that \( n \) potential shelves are considered (each of them with a prefixed height) introduces specific and effective cutting rules.

In addition, some other symmetries can be avoided in both models by using two sets of linear inequalities; for simplicity, we first present these inequalities for Mod1(2TDC), and successively adapt them to Mod2^+(2TDC).

The first set of inequalities is aimed at cutting off from the search space the equivalent solutions in which a shelf \( k \) is used instead of a shelf \( \ell \), both corresponding to identical items, and with \( k > \ell \). We call these inequalities **Ordering Inequalities** (OIs), and their correctness is proved by the following theorem.
Theorem 2 For any shape’s type \( i = 1, \ldots, m \), defining the \( w_b \) identical items \( j, j+1, \ldots, j+w_i - 1 \) (with \( j = \sum_{s=1}^{i-1} u_{bs} + 1 \)), linear inequalities
\[
x_{kh} \geq x_{k+1,h+1} \quad (k = j, \ldots, j + w_i - 2) \quad (17)
\]

**Proof:** Suppose that in an optimal solution of 2TDC one of the inequalities (17), say \( x_{kh} \geq x_{k+1,h+1} \), is violated. This implies that item \( k \) has been cut off from one of the shelves \( 1, \ldots, k-1 \), say \( s \), whereas item \( k+1 \) initializes its shelf. However, since \( k \) and \( k+1 \) are identical, it is always possible to cut item \( k+1 \) from \( s \), initialize shelf \( k \) with its corresponding item, thus obtaining from shelf \( k \) the items which were cut from shelf \( k+1 \). Hence, we obtained an equivalent optimal solution for which inequality \( x_{kh} \geq x_{k+1,h+1} \) is satisfied.

The second set of inequalities concerns the number of additional items of a given shape’s type, say \( i \), which can be cut from shelves, say \( k \) and \( \ell \), such that \( \beta_k = \beta_\ell = i \). In particular, we want to eliminate from the search space those solutions in which the number of these items cut from shelf \( \ell \) is greater than that cut from shelf \( k \) if \( k < \ell \). We refer to these inequalities as **Extended Ordering Inequalities** (EOIs), and their correctness is proved by the following theorem.

Theorem 3 For any shape’s type \( i = 1, \ldots, m \), defining the \( u_{b_i} \) identical items \( j, j+1, \ldots, j+u_{b_i} - 1 \) (with \( j = \sum_{s=1}^{i-1} u_{bs} + 1 \)), linear inequalities
\[
\sum_{s=k+1}^{k+u_{b_i}-1} x_{sk} \geq \sum_{s=k+2}^{k+u_{b_i}} x_{s,k+1} \quad (k = j, \ldots, j+u_{b_i} - 2) \quad (18)
\]

**Proof:** The proof of Theorem 2 can be easily adapted. Suppose that in an optimal solution of 2TDC one of the inequalities (18), say the one corresponding to shelves \( k \) and \( \ell \) (\( k < \ell \)), is violated by \( t \). Since the shelves are initialized by identical items, it is always allowed to move for example \( t \) items (of the same type as the ones initializing the shelves) from \( \ell \) to \( k \) (thus satisfying the violated inequality), and, possibly, some items of different type from \( k \) to \( \ell \) in order to satisfy constraints (4). It is easy to see that in a feasible solution satisfying inequalities (18) \( k \) and \( \ell \) have been in some sense swapped.

Corresponding **Ordering Inequalities** (OIs) can be derived from the second model: we obtain respectively
\[
q_k \geq q_{k+1}
\]

(\( i = 1, \ldots, m ; \ k = \alpha_{i-1} + 1, \ldots, \alpha_i - 1 \) (19) and
\[
x_{i,k} \geq x_{i,k+1}
\]

(\( i = 1, \ldots, m ; \ k = \alpha_{i-1} + 1, \ldots, \alpha_i - 1 \) (20) whose correctness is immediate from Theorems 2 and 3 above.

Both **Ordering Inequalities** and **Extended Ordering Inequalities** can be added to the models in the construction phase \((O(n) + O(n)\) inequalities in both cases), and their effectiveness is computationally tested in the following section.

4. **COMPUTATIONAL EXPERIMENTS**

We tested the models introduced in Section 2, on a set of instances from the literature by using the standard Branch-and-Bound (without specific tuning of the parameters) of the commercial ILP solver Cplex version 6.5.3. The code runs on a Digital Alpha 533 Mhz.

In particular, we considered the set of 38 FC-2TDC instances\(^1\) solved in (Hiﬁ and Roucairol, 2000), composed of 14 weighted instances and 24 unweighted ones.

The computational experiments\(^2\), reported in Table 1, are aimed at showing the effectiveness of both the linear inequalities added to avoid symmetries, and the models with respect to the best exact algorithm presented in the literature by (Hiﬁ and Roucairol, 2000). In Table 1 we report computational results comparing the algorithm above (HR in the table) with Mod1(2TDC) and Mod2\(^+\)(2TDC) in their basic versions and with the addition of OIs and EOIs introduced in Section 3. The results are given separately for weighted and unweighted instances (the cardinality of each set is indicated as \( N \)), and, for algorithm HR and for each model, both in basic and strengthened versions, the average computing time (in seconds) and the number of unsolved instances (uns. in the table) within the time limit of 1,800 seconds are reported\(^3\). We need to note that the literature of 2TDC distinguishes two cases: (i) the case in which the first cut is horizontal, i.e., the case we referred to in the entire paper, and (ii) the case in which the first cut is vertical. In the latter case, we can simply extend the terminology of shelf patterns by referring to “shelves” also as “columns” forming levels (see the discussion at the beginning of Section 2). Hence, each of the instances is solved twice, and the average values are computed over 76 runs instead of 38.

The results in Table 1 show that the behavior of our models is satisfactory with respect to algorithm HR for both

\(^1\)Available at ftp://panoramix.univ-paris1.fr/pub/CERMSEM/hifi/

\(^2\)The extended version of this section is reported in (Lodi and Monaci, 2000).

\(^3\)Computing times for HR are on a UltraSparc10 250 Mhz with a time limit of 7,200 seconds.
Table 1. FC-2TDC: effectiveness of OIs and EOIs for Mod1(2TDC) and Mod2⁺(2TDC) and comparison with results from (Hiﬁ and Roucairol, 2000).

<table>
<thead>
<tr>
<th>Instances type</th>
<th>N</th>
<th>HR (Hiﬁ and Roucairol, 2000)</th>
<th>Mod1(2TDC)</th>
<th>Mod2⁺(2TDC)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>time</td>
<td>unsolved</td>
<td>without OIs,EOIs</td>
</tr>
<tr>
<td>weighted</td>
<td>14</td>
<td>658.14</td>
<td>2</td>
<td>40.90</td>
</tr>
<tr>
<td>unweighted</td>
<td>24</td>
<td>26.23</td>
<td>-</td>
<td>307.83</td>
</tr>
</tbody>
</table>

weighted and unweighted instances. Moreover, for both the classes of instances, the addition of linear inequalities in order to avoid symmetries in the models is very effective by allowing the solution of the entire set of instances. Note that, since we considered as solution time for the unsolved instances the time limit, the average computing times are an optimistic estimate, thus the already relevant speed up shown by the table could be even higher.

ACKNOWLEDGMENTS

We are grateful to Silvano Martello and Daniele Vigo for introducing us to the wide domain of Cutting & Packing. We acknowledge the support given to this project by the Consiglio Nazionale delle Ricerche (CNR), Italy.

REFERENCES


