Symmetry-breaking constraints for packing identical rectangles within polyhedra*

R. Andrade [†]   E. G. Birgin[†]

August 9, 2011

Abstract

Two problems related to packing identical rectangles within a polyhedron are tackled in the present work. Rectangles are allowed to differ only by horizontal or vertical translations and possibly ninety-degree rotations. The first considered problem consists in packing as many identical rectangles as possible within a given polyhedron, while the second problem consists in finding the smallest polyhedron of a given type that accommodates a fixed number of identical rectangles. Both problems are modeled as mixed integer programming problems. Symmetry-breaking constraints that facilitate the solution of the MIP models are introduced. Numerical results are presented.

Key words: Packing of rectangles, MIP models, symmetry-breaking constraints, algorithms.

1   Introduction

Several recent papers deal with the problem of packing as many identical rectangles as possible within an arbitrary convex region, allowing the rectangles to differ only by horizontal or vertical translations and possibly ninety-degree rotations [2, 5, 7, 9, 14]. See also [8, 10, 11]. When the convex region is a polyhedra – as opposed to a region with curved boundaries – the problem can be modeled as a mixed integer programming (MIP) problem and solved to optimality (see [1, 3, 4, 6] for the particular case of packing rectangles within rectangles). When all the rectangles are identical, it is possible to define equivalent solutions in which the roles of a pair of rectangles R_i and R_j are reversed. The existence of a multiplicity of equivalent solutions can slow down a branch-and-bound algorithm, as it expands the size of the search tree. As a result, several authors consider "symmetry-breaking constraints" that eliminate many of these equivalent solutions (see, for example, [12, 13]).

The contribution of this paper is to explore, with the help of numerical examples, the effect of three kinds of symmetry-breaking constraints: one related to the rectangles rotations and two related to how the rectangles are spatially ordered. We illustrate the impact of the symmetry-breaking constraints in the context of two problems: (i) packing as many identical rectangles within an equilateral triangle; and (ii) finding the smallest equilateral triangle within which a given set of rectangles can be packed. In both problems, the rectangles edges must be aligned with the Cartesian axes; that is, they can only be translated or rotated by ninety degrees. One slightly surprising result is that adding symmetry-breaking constraints is not always beneficial. In particular, if one is trying to determine whether k rectangles can be packed in a triangle, and if it turns out that the answer is "yes", then adding the symmetry-breaking constraints actually slows down the MIP. The main advantage of the symmetry-breaking constraints comes when proving the optimality of a solution or the infeasibility of a problem.

The rest of the paper is organized as follows. Section 2 describes the models of both problems. In Section 3, the symmetry-breaking constraints are presented. Numerical experiments are shown in Section 4 and some conclusions are provided in Section 5.

2 Mixed integer programming models

Consider a set of $R_1, \ldots, R_k$ rectangles with height h and width w, centered at the origin of the Cartesian two-dimensional space and with their sides parallel to the axes. We would like to pack them within a given polyhedron Ω allowing translations and ninety-degree rotations. To model the ninety-degree rotations we introduce binary variables

$$r_i \in \{0, 1\}, i = 1, \ldots, k, \tag{1}$$

and define

$$h_i \equiv (1 - r_i)h + r_i w \ \text{ and } \ w_i \equiv (1 - r_i)w + r_i h, \ \ i = 1, \ldots, k,$$

in such a way that $r_i = 0$ means that $h_i = h$ and $w_i = w$, i.e. that rectangle R_i preserves its original orientation. On the other hand, $r_i = 1$ means that $h_i = w$ and $w_i = h$ implying that rectangle R_i received a ninety-degree rotation with respect to its original orientation.

Let $(c_i^x, c_i^y)^T \in \mathbb{R}^2$ be the center coordinates of rectangle R_i with height h_i and width w_i, for all $i = 1, \ldots, k$. It is easy to see that, for every pair (i, j) with $j > i$, rectangles R_i and R_j do not overlap if and only if

$$|c_i^x - c_j^x| \geq \frac{1}{2}(w_i + w_j) \ \text{ or } \ |c_i^y - c_j^y| \geq \frac{1}{2}(h_i + h_j), \text{ for all } i = 1, \ldots, k, j = i+1, \ldots, k. \tag{2}$$

Accommodating the rectangles within the polyhedral object Ω can be modeled as

$$v_i^\ell \in \Omega, \ \ell \in D, \ i = 1, \ldots, k, \tag{3}$$

where $D = \{NW, NE, SE, SW\}$ and

$$v_i^{NW} \equiv (c_i^x - \frac{1}{2}w_i, c_i^y + \frac{1}{2}h_i)^T, \quad v_i^{NE} \equiv (c_i^x + \frac{1}{2}w_i, c_i^y + \frac{1}{2}h_i)^T,$$

$$v_i^{SE} \equiv (c_i^x + \frac{1}{2}w_i, c_i^y - \frac{1}{2}h_i)^T, \quad v_i^{SW} \equiv (c_i^x - \frac{1}{2}w_i, c_i^y - \frac{1}{2}h_i)^T,$$

define the four vertices of rectangle R_i for $i = 1, \ldots, k$.

Constraints (1,2,3) define a linear generalized disjunctive programing (LGDP) [16] (feasibility) model[1], called $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ from now on, for the problem of placing k non-overlapping rectangles with height h and width w, allowed to differ only by horizontal or vertical translations and possibly ninety-degree rotations, within Ω. The model has k binary variables $(r_i, i = 1, \ldots, k)$ and $2k$ continuous variables $(c_i^x$ and c_i^y, $i = 1, \ldots, k)$. Note that h_i, w_i, v_i^ℓ, $\ell \in D$, $i = 1, \ldots, k$, are not variables of the model but auxiliary values used to simplify the presentation. In (2) there are $k(k-1)/2$ disjunctive constraints constituted by four linear constraints each, while the number of constraints in (3) depends on the description of Ω. For example, if Ω were describing a triangle, there would be $12k$ additional linear constraints (three linear constraints per each of the four vertices of the k rectangular items). For solving the original problem of packing as many rectangles as possible, problem $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ can be solved for increasing values of $k = 1, 2, \ldots$ until the first infeasible problem $\mathcal{P}_{LGDP}^{k',[\Omega,h,w]}$ is detected. In this case, $N = k' - 1$ is the maximum number of rectangles that can be placed in Ω and the solution for $\mathcal{P}_{LGDP}^{N,[\Omega,h,w]}$ shows how to place them within Ω.

According to [16] (pp. 1892–1893), the most common alternatives to reformulate the non-overlapping disjunctive constraints in (2) as mixed integer linear constraints are the big-M and the convex-hull reformulations. Defining a pair of binary variables q_{ij} and q_{ji} for every pair (i, j) with $j > i$, and using sufficiently large constants $M_w > 0$ and $M_h > 0$, the big-M reformulation of (2) can be written as

$$
\begin{aligned}
c_i^x - c_j^x &\geq \frac{1}{2}(w_i + w_j) - M_w q_{ij} - M_w q_{ji}, \\
c_j^x - c_i^x &\geq \frac{1}{2}(w_i + w_j) - M_w(1 - q_{ij}) - M_w(1 - q_{ji}), \\
c_i^y - c_j^y &\geq \frac{1}{2}(h_i + h_j) - M_h(1 - q_{ij}) - M_h q_{ji}, \\
c_j^y - c_i^y &\geq \frac{1}{2}(h_i + h_j) - M_h q_{ij} - M_h(1 - q_{ji}), \\
q_{ij}, q_{ji} &\in \{0,1\}, \text{for all } i = 1, \ldots, k, j = i + 1, \ldots, k.
\end{aligned}
\tag{4}
$$

In (4), variables q_{ij} and q_{ji} are used to enforce at least one of four constraints, and can also be interpreted as capturing the relative position between rectangles R_i and R_j. Roughly speaking, for rectangles R_i and R_j not to overlap, there are four possibilities: rectangle R_j can be to the left of, to the right of, below, or above rectangle R_i; and those possibilities are related to (i) $q_{ij} = q_{ji} = 0$, (ii) $q_{ij} = q_{ji} = 1$, (iii) $q_{ij} = 1$ and $q_{ji} = 0$, and (iv) $q_{ij} = 0$ and $q_{ji} = 1$, respectively. See Figure 1.

Defining sixteen continuous variables $d_{i,[i,j,m]}^x$, $d_{i,[i,j,m]}^y$, $d_{j,[i,j,m]}^x$, $d_{j,[i,j,m]}^y$, $m = 1, \ldots, 4$, and four binary variables $s_{[i,j,m]}$, $m = 1, \ldots, 4$, for every pair (i, j) with $j > i$, the convex-hull

---
[1]We are, in fact, considering that $|x| \geq c$ can be written as $x \geq c$ or $x \leq -c$.
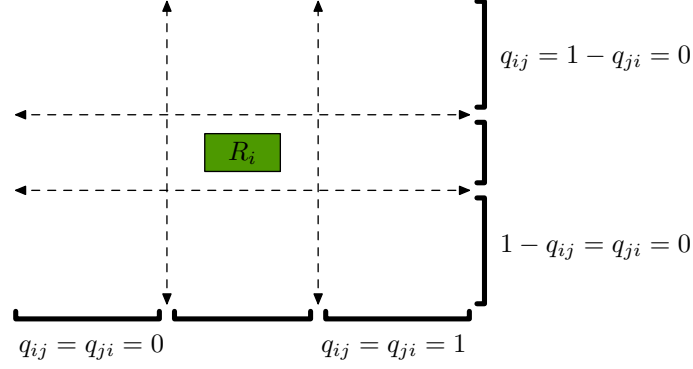
Figure 1: Relation between the values of binary variables q_{ij} and q_{ji} in (4) and the relative position between rectangles R_i and R_j.

reformulation of (2) can be written as

$$
\begin{aligned}
&\sum_{m=1}^{4} s_{[i,j,m]} = 1, \quad s_{[i,j,m]} \in \{0,1\}, \quad m = 1, \ldots, 4, \\
&c_u^x = \sum_{m=1}^{4} d_{u,[i,j,m]}^x, \quad c_u^y = \sum_{m=1}^{4} d_{u,[i,j,m]}^y, \quad u = i, j, \\
&0 \leq d_{u,[i,j,m]}^x \leq \hat{c}^x\, s_{[i,j,m]}, \quad 0 \leq d_{u,[i,j,m]}^y \leq \hat{c}^y\, s_{[i,j,m]}, \quad u = i, j, \quad m = 1, \ldots, 4, \\
&d_{i,[i,j,1]}^x - d_{j,[i,j,1]}^x \geq \frac{1}{2}(w_i + w_j)\, s_{[i,j,1]}, \\
&d_{j,[i,j,2]}^x - d_{i,[i,j,2]}^x \geq \frac{1}{2}(w_i + w_j)\, s_{[i,j,2]}, \\
&d_{i,[i,j,3]}^y - d_{j,[i,j,3]}^y \geq \frac{1}{2}(h_i + h_j)\, s_{[i,j,3]}, \\
&d_{j,[i,j,4]}^y - d_{i,[i,j,4]}^y \geq \frac{1}{2}(h_i + h_j)\, s_{[i,j,4]}, \\
&\text{for all } i = 1, \ldots, k, j = i+1, \ldots, k.
\end{aligned}
\tag{5}
$$

In (5), it is assumed that $0 \leq c_i^x \leq \hat{c}^x$ and $0 \leq c_i^y \leq \hat{c}^y$, for $i = 1, \ldots, k$. A sufficient condition to have those bounds is that

$$
\Omega \subseteq [0,0] \times [\hat{c}^x, \hat{c}^y] \subset \mathbb{R}^2.
\tag{6}
$$

We will assume that (6) holds and that constants $\hat{c}^x$ and $\hat{c}^y$ are known from now on. If, for example, Ω were an equilateral triangle of side L, it would be enough to locate it with its bottom-left corner at the origin of the Cartesian axes and to consider $\hat{c}^x \equiv L$ and $\hat{c}^y \equiv (\sqrt{3}/2)L$.

Constraints (1,3,4) and (1,3,5) define MIP (feasibility) models, named $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$, respectively, equivalent to $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$. In $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$ there are two additional binary variables $(q_{ij}, q_{ji}, j = i+1, \ldots, k)$ for each pair of rectangles, for a total of $k + 2[k(k-1)/2] = k^2$ binary variables and $2k$ continuous variables. The number of constraints in (4) is $4k(k-1)/2$. In

$\mathcal{P}_{CH}^{k,[\Omega,h,w]}$ there are sixteen additional continuous variables ($d^x_{i,[i,j,m]}$, $d^y_{i,[i,j,m]}$, $d^x_{j,[i,j,m]}$, $d^y_{j,[i,j,m]}$, $m = 1, \ldots, 4$) for each pair of rectangles, for a total of $2k + 16[k(k-1)/2]$ continuous variables. There are also four additional binary variables ($s_{[i,j,m]}$, $m = 1, \ldots, 4$) for each pair of rectangles, for a total of $k + 4[k(k-1)/2] = 2k^2 - k$ binary variables. The number of constraints in (5) is $41k(k-1)/2$.

Assume now that Ω is a polyhedron given by $\Omega \equiv \{x \in \mathbb{R}^2 \mid Ax \leq b\}$ and that a fixed number k of identical rectangles is given. The problem of finding the "smallest polyhedron of the same type of Ω" that accommodates the k rectangles can be modeled as

$$
\begin{aligned}
&\text{Minimize} \quad L \\
&\text{s.t.} \quad\quad Av^\ell_i \leq Lb, \; \ell \in D, \; i = 1, \ldots, k, \\
&\quad\quad\quad\; \text{plus constraints (1,2).}
\end{aligned}
\tag{7}
$$

Big-M and convex-hull MIP reformulations of (7) can be obtained by replacing (2) by (4) and (5), respectively. We name the LGDP model (7) and its big-M and convex-hull MIP reformulations as $\mathcal{S}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{S}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{S}_{CH}^{k,[\Omega,h,w]}$, respectively.

3 Symmetry-breaking constraints

Given a solution ($c^x_i, c^y_i, r_i, i = 1, \ldots, k$) for problem $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ and choosing any pair (i,j) with $j > i$, it is possible to define an equivalent or symmetric solution in which the roles of rectangles R_i and R_j are interchanged. From now on, we will use the terms *equivalent solutions* or *symmetric solutions* to refer to solutions that can be obtained one from the other by simple relabeling the packed rectangles. It is easy to see that each solution belongs to a family composed by $k!$ equivalent solutions corresponding to all possible ways of numbering the packed rectangles. It is well known that the presence of symmetric solutions may impair the performance of a branch-and-bound method (see, for example, [15] and the references therein). Basically, a traditional branch-and-bound method may have difficulties pruning tree nodes associated with equivalent subproblems and, in consequence, a prohibitive amount of enumeration may be required.

The simplest set of constraints that may be added to eliminate equivalent solutions, preserving at least a representative within the feasible set, is

$$
r_i \leq r_{i+1}, \; i = 1, \ldots, k-1.
\tag{8}
$$

If there is a solution with p non-rotated rectangles and $k - p$ rotated rectangles, the set of constraints (8) clearly says that the non-rotated rectangles must be rectangles $R_1, \ldots, R_p$ while the rotated rectangles must be $R_{p+1}, \ldots, R_k$.

However, when considering model $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ with the additional constraints (8), symmetric solutions that correspond to interchanging places of any pair of rectangles with the same orientation still exist. To eliminate symmetric solutions of this type, we would like to add constraints that impose a lexicographical (total) order within the rectangles with the same orientation. These constraints can be written as

$$
r_i = r_{i+1} \; \Rightarrow \; (c^x_{i+1} > c^x_i \; \text{ or } \; (c^x_{i+1} = c^x_i \text{ and } c^y_{i+1} \geq c^y_i)), \; i = 1, \ldots, k-1,
\tag{9}
$$

but the strict inequality in (9) and the continuity of variables $c_i^x, c_i^y, i = 1, \dots, k$, prevent us from expressing the lexicographical order constraints as mixed integer linear constraints. Replacing the strict inequality with "less than or equal to" reduces (9) to

$$r_i = r_{i+1} \;\Rightarrow\; c_{i+1}^x \geq c_i^x, \; i = 1, \dots, k-1,$$

that turns out to be a preorder (reflexive and transitive binary relation) among the rectangles with the same orientation. In fact, for any $\alpha, \beta \in \mathbb{R}$,

$$r_i = r_{i+1} \;\Rightarrow\; \alpha c_{i+1}^x + \beta c_{i+1}^y \geq \alpha c_i^x + \beta c_i^y, \; i = 1, \dots, k-1,$$

or, equivalently,

$$\alpha c_{i+1}^x + \beta c_{i+1}^y \geq \alpha c_i^x + \beta c_i^y - M_o(r_{i+1} - r_i), \; i = 1, \dots, k-1, \tag{10}$$

where $M_o > 0$ is a sufficiently large constant, defines a preorder among the rectangles with the same orientation. (In the context of a facility layout problem, similar constraints aiming to reduce the number of different orderings of rectangles were also considered in [17].) Figure 2 illustrates the effect of the symmetry-breaking constraints (8) and (10).
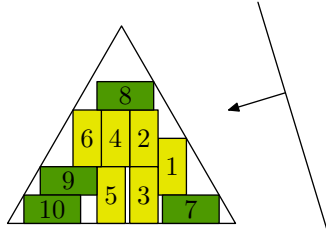
Figure 2: This figure corresponds to the graphical representation of an instance with 10 rectangular items. The line represents constraint (10) with $\alpha \approx -0.827$ and $\beta \approx -0.251$. Note that when the line is dragged in the direction indicated by the arrow, rectangles with the same orientation are numbered in increasing order as the line touches their centers. This illustrates the preorder imposed by constraint (10). In addition, non-rotated rectangles, which correspond to $r_i = 0$, are numbered before the rotated ($r_i = 1$) rectangles as dictated by constraint (8).

We may arrive to a different approach for eliminating symmetric solutions by replacing the non-overlapping disjunctive constraints (2) by

$$c_j^x - c_i^x \geq \frac{1}{2}(w_i + w_j) \;\; \text{or} \;\; c_j^y - c_i^y \geq \frac{1}{2}(h_i + h_j), \; \text{for all } i = 1, \dots, k, j = i+1, \dots, k. \tag{11}$$

Constraints (11) say, simultaneously, that, for every pair (i, j) with $j > i$, R_j and R_i must not overlap and that R_j must be situated to the right of or above R_i. As well as (2), disjunctive constraints (11) can be rewritten as a set of MIP constraints by considering the big-M or the convex-hull reformulations. Defining a binary variable q_{ij} for every pair (i, j) with $j > i$, and

6

using sufficiently large constants $M_w > 0$ and $M_h > 0$, the big-M reformulation of (11) can be written as

$$c_j^x - c_i^x \geq \frac{1}{2}(w_i + w_j) - M_w q_{ij},$$

$$c_j^y - c_i^y \geq \frac{1}{2}(h_i + h_j) - M_h(1 - q_{ij}), \qquad (12)$$

$$q_{ij} \in \{0, 1\}, \text{for all } i = 1, \ldots, k, j = i + 1, \ldots, k.$$

In (12), q_{ij} is used to enforce at least one of two constraints, and, as well as in (4), can be interpreted as capturing the relative position between rectangles R_i and R_j, $q_{ij} = 0$ meaning that R_j is to the right of R_i, and $q_{ij} = 1$ meaning that R_j is above R_i. Defining eight continuous variables $d_{i,[i,j,m]}^x$, $d_{i,[i,j,m]}^y$, $d_{j,[i,j,m]}^x$, $d_{j,[i,j,m]}^y$, $m = 1, 2$, and two binary variables $s_{[i,j,1]}$ and $s_{[i,j,2]}$, for every pair (i, j) with $j > i$, the convex-hull reformulation of (11) can be written as

$$s_{[i,j,1]} + s_{[i,j,2]} = 1, \quad s_{[i,j,1]}, s_{[i,j,2]} \in \{0, 1\},$$

$$c_u^x = d_{u,[i,j,1]}^x + d_{u,[i,j,2]}^x, \quad c_u^y = d_{u,[i,j,1]}^y + d_{u,[i,j,2]}^y, \quad u = i, j,$$

$$0 \leq d_{u,[i,j,1]}^x \leq \hat{c}^x s_{[i,j,1]}, \quad 0 \leq d_{u,[i,j,1]}^y \leq \hat{c}^y s_{[i,j,1]}, \quad u = i, j,$$

$$0 \leq d_{u,[i,j,2]}^x \leq \hat{c}^x s_{[i,j,2]}, \quad 0 \leq d_{u,[i,j,2]}^y \leq \hat{c}^y s_{[i,j,2]}, \quad u = i, j, \qquad (13)$$

$$d_{j,[i,j,1]}^x - d_{i,[i,j,1]}^x \geq \frac{1}{2}(w_i + w_j) \, s_{[i,j,1]},$$

$$d_{j,[i,j,2]}^y - d_{i,[i,j,2]}^y \geq \frac{1}{2}(h_i + h_j) \, s_{[i,j,2]},$$

$$\text{for all } i = 1, \ldots, k, j = i + 1, \ldots, k.$$

Summing up, adding constraints (8,10) to models $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$, we arrive to equivalent models that will be named $\mathcal{Q}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{Q}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{Q}_{CH}^{k,[\Omega,h,w]}$, respectively. On the other hand, replacing (2), (4), and (5), by (11), (12), and (13), in $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$, we also arrive to equivalent models that will be named $\mathcal{R}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{R}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{R}_{CH}^{k,[\Omega,h,w]}$, respectively. The "$\mathcal{Q}$" and "$\mathcal{R}$" models are equivalent to their corresponding "$\mathcal{P}$" models but with less representatives of each family of equivalent or symmetric solutions. Figure 3 summarizes the main characteristics of each model and the relations among them. To see that at least a representative of each family of symmetric solutions for a "$\mathcal{P}$" model is a feasible solution for the corresponding "$\mathcal{Q}$" and "$\mathcal{R}$" models, it is enough to note that, given a solution for a "$\mathcal{P}$" model, it is possible to relabel the rectangles in order to satisfy the additional constraints that are present in the corresponding "$\mathcal{Q}$" and "$\mathcal{R}$" models.

In an analogous way, the symmetry-breaking constraints can also be easily incorporated in $\mathcal{S}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{S}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{S}_{CH}^{k,[\Omega,h,w]}$. We name $\mathcal{T}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{T}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{T}_{CH}^{k,[\Omega,h,w]}$, the models that result of adding symmetry-breaking constraints (8,10); while we name $\mathcal{U}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{U}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{U}_{CH}^{k,[\Omega,h,w]}$, the models that result of replacing (2), (4), and (5), by (11), (12), and (13), in $\mathcal{S}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{S}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{S}_{CH}^{k,[\Omega,h,w]}$, respectively.
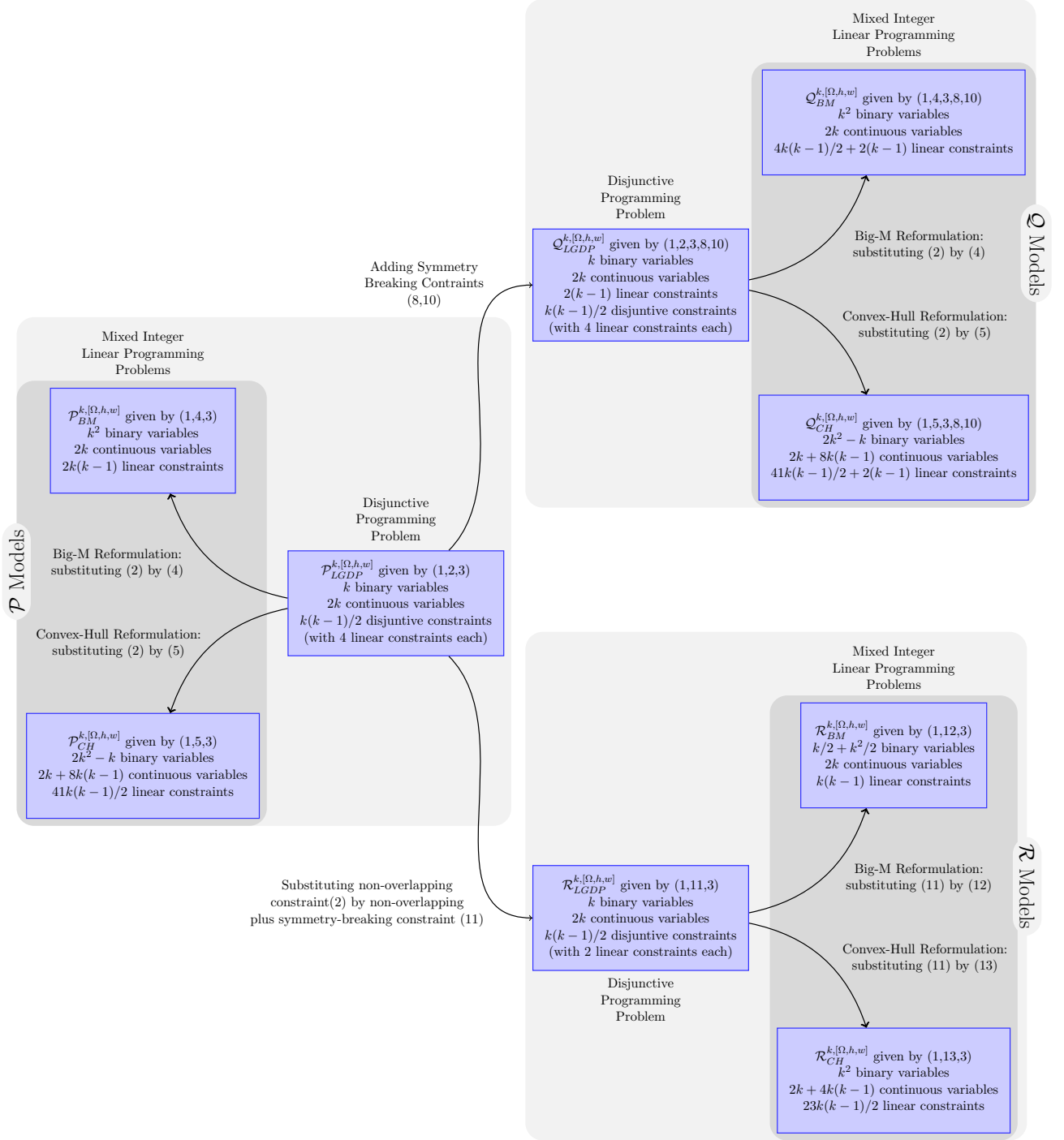
Figure 3: Description of the disjunctive programming and mixed integer linear programming feasibility models for the problem of placing k non-overlapping identical rectangles with height h and width w, allowed to differ only by horizontal or vertical translations and possibly ninety-degree rotations, within Ω. Numbers of constraints do not include the number of placing constraints (3), that depends on the description of Ω.

4 Numerical experiments

In the numerical experiments we considered the case of equilateral triangular objects and identical rectangular items with height $h = 1$ and width $w = 0.5$. Equilateral triangles were arbitrarily chosen as illustrative examples. There is no additional difficulty in considering a generic polyhedral object Ω, other than dealing with the additional linear constraints needed to define it. Note that, as detailed in (3), if the polyhedral object is described by p linear inequalities, each item requires only $4p$ linear constraints to enforce its vertices (and, in consequence, the whole item) to be accommodated within the object. All the models considered in the numerical experiments were implemented in C/C++ (gcc version 4.4.3) and solved with CPLEX (IBM ILOG AMPL/CPLEX 12.1.0) using its default parameters. It means that a solution is reported as optimal by the solver when

$$\text{absolute gap} = \text{best feasible solution} - \text{best lower bound} \leq 10^{-6}$$

or

$$\text{relative gap} = \frac{|\text{ best feasible solution} - \text{best lower bound }|}{1e{-}10 + |\text{ best feasible solution }|} \leq 10^{-4}.$$

In order to test the proposed models, values for constants α, β, and M_o in (10), M_w and M_h in (4,12), and $\hat{c}^x$ and $\hat{c}^y$ in (5,13) must be given. For α and β we considered random numbers with uniform distribution within the interval $[-1, 1]$. A sufficient condition for the correctness of the symmetry-breaking constraints (10) is $M_o \geq L_{ub} + \frac{\sqrt{3}}{2}L_{ub}$, where L_{ub} is an upper bound on the (possible unknown) side L of the triangular object. A sufficient condition for the correctness of the overlapping constraints (4,12) is $M_w \geq L_{ub}$ and $M_h \geq \frac{\sqrt{3}}{2}L_{ub}$. Valid values for $\hat{c}^x$ and $\hat{c}^y$ in (5,13) are $\hat{c}^x = L_{ub}$ and $\hat{c}^y = \frac{\sqrt{3}}{2}L_{ub}$. When the triangle is given and the problem consists on computing the largest number of identical rectangles that can be packed within the triangle, L is known and we set $L_{ub} = L$. When the problem consists on computing the smallest equilateral triangle that packs a given fixed number of identical rectangles, we proceed as follows. A square that contains the given number N of identical rectangular items is computed, and the side of the smallest equilateral triangle that circumscribe the square is taken as L_{ub}. As a result, in the case in which L is unknown, the considered values of L_{ub} where $L_{ub} = 2.16$ for $N \in \{1, 2\}$, $L_{ub} = 4.31$ for $N \in \{2, 3, \ldots, 8\}$, and $L_{ub} = 6.47$ for $N \in \{9, 10, \ldots, 16\}$. Note that providing those values is important as it is well know that they interfere in the performance of the considered MIP solver.

4.1 Minimizing the object dimension

In this subsection we deal with the problem of finding the smallest equilateral triangle within which a given set of N identical rectangular items can be packed. (Recall that we are dealing with rectangular items with height $h = 1$ and width $w = 0.5$.)

In a first set of experiments, in order to have a starting point for comparison, we tried to solve instances of $\mathcal{S}_{LGDP}^{k,[\Omega,h,w]}$ (LGDP model (7)), $\mathcal{S}_{BM}^{k,[\Omega,h,w]}$ (big-M MIP reformulation), and $\mathcal{S}_{CH}^{k,[\Omega,h,w]}$ (convex-hull MIP reformulation), that do not include any symmetry-breaking constraint, increasing the number of items $N \in \{1, 2, \ldots\}$. Table 1 shows the results. In the table, N is the

number of items being packed. For the side of the smallest equilateral triangle that packs the N items, the table shows the best lower bound as well as the best feasible solution found by the solver. Relative and absolute gaps are also provided. The right-hand side of the table displays the effort measurements: "MIP" is the number of MIP iterations, "B&B nodes" is the number of nodes in the branch-and-bound tree, and "CPU Time" is the CPU time in seconds. In the table, it can be seen that, with a CPU time limit of 6 hours, we were not able to solve instances with $N > 8$.

	Fixed N	Instance data					Effort measurements		
		Triangle side		MIP GAP			MIP	B&B nodes	CPU Time
		Best lower bound	Best feasible solution	Relative	Absolute				
LGDP model (7)	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00		0	0	0.00
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00		38	4	0.01
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00		215	53	0.01
	4	2.732050807568878	2.732050807568878	0.00e+00	0.00e+00		1,212	315	0.04
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00		41,003	12,476	1.46
	6	3.231852460094526	3.232050807568877	6.14e−05	1.98e−04		910,795	232,853	40.26
	7	3.443060783087007	3.443375672338170	9.14e−05	3.15e−04		27,536,014	6,626,931	1,466.85
	8	3.597717089329720	3.598076211353316	9.98e−05	3.59e−04		153,247,823	36,600,213	11,186.81
	9	3.215704772343326	3.809401068449104	1.56e−01	5.94e−01		−	−	>21,600.00
	10	3.016083139658178	4.020725924957480	2.50e−01	1.00e+00		−	−	>21,600.00
Big-M MIP reformulation	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00		0	0	0.00
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00		30	11	0.01
	3	2.443375672974065	2.443375672974065	0.00e+00	0.00e+00		149	80	0.01
	4	2.732050807568878	2.732050807568878	0.00e+00	0.00e+00		1,315	725	0.07
	5	3.020725942163690	3.020725942163690	0.00e+00	0.00e+00		28,832	17,763	1.08
	6	3.232050807568878	3.232050807568878	0.00e+00	0.00e+00		457,542	283,835	17.45
	7	3.443375672974065	3.443375672974065	0.00e+00	0.00e+00		27,097,399	14,284,652	1,032.67
	8	3.443375672974065	3.598076205466564	4.30e−02	1.55e−01		−	−	>21,600.00
	9	3.154700538379261	3.809401070449102	1.72e−01	6.55e−01		−	−	>21,600.00
	10	3.065384140902228	4.020725935431639	2.38e−01	9.55e−01		−	−	>21,600.00
Convex-hull MIP reform.	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00		0	0	0.00
	2	2.154700518676984	2.154700529884166	5.20e−09	1.12e−08		78	16	0.02
	3	2.443375667541825	2.443375667541825	0.00e+00	0.00e+00		2,123	259	0.07
	4	2.732031213559349	2.732050717264398	7.14e−06	1.95e−05		33,886	3,841	1.16
	5	3.020486516477845	3.020725894730525	7.92e−05	2.39e−04		1,592,352	138,176	60.18
	6	3.231727579601285	3.232050397095424	9.99e−05	3.23e−04		45,750,048	3,594,324	2,019.11
	7	3.154699924971216	3.443373062238137	8.38e−02	2.89e−01		−	−	>21,600.00
	8	2.809401076681599	3.598076080375179	2.19e−01	7.89e−01		−	−	>21,600.00
	9	2.618740741071466	3.866023671044396	3.23e−01	1.25e+00		−	−	>21,600.00
	10	2.427524391209028	4.020724784525934	3.96e−01	1.59e+00		−	−	>21,600.00

Table 1: Instances of LGDP model $\mathcal{S}_{LGDP}^{k,[\Omega,h,w]}$ and its MIP reformulations $\mathcal{S}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{S}_{CH}^{k,[\Omega,h,w]}$ with increasing number of items $N \in \{1, 2, \dots\}$. Recall that these models correspond to pure models with no symmetry-breaking constraints.

In a second set of experiments, we were interested in testing the efficiency of the introduced symmetry-breaking constraints (8,10). Hence, we considered instances of problems $\mathcal{T}_{LGDP}^{k,[\Omega,h,w]}$ (LGDP model (7) plus symmetry-breaking constraints (8,10)), $\mathcal{T}_{BM}^{k,[\Omega,h,w]}$ (big-M MIP reformulation), and $\mathcal{T}_{CH}^{k,[\Omega,h,w]}$ (convex-hull MIP reformulation). Regarding constraint (8), it is easy to see that there are only $k + 1$ possible values for variables r_i, $i = 1, \dots, k$, such that (8) holds.

The $k+1$ possibilities correspond to

$$r_i = \left\{ \begin{array}{ll} 0, & \text{if } i \leq p, \\ 1, & \text{otherwise,} \end{array} \right. \quad i = 1, \ldots, k, \tag{14}$$

for $p = 0, \ldots, k$. Therefore, there are two possible ways of considering constraint (8) in connection with problem (7): (i) $k+1$ problems of type (7) may be solved fixing r_i, $i = 1, \ldots, k$, as suggested in (14) for $p = 0, \ldots, k$, or (ii) constraint (8) can be explicitly incorporated into (7) solving (7,8). Tables 2 and 3 show options (i) and (ii), respectively, with the addition of constraint (10). In Table 2, we can see that all instances of $\mathcal{T}_{LGDP}^{k,[\Omega,h,w]}$ and $\mathcal{T}_{BM}^{k,[\Omega,h,w]}$ with N up to 10 were solved to optimality, while instances of $\mathcal{T}_{CH}^{k,[\Omega,h,w]}$ were solved to optimality only with N up to 8^2. Anyway, in the three models, the symmetry-breaking constraints (8,10) helped to reduce the computational effort needed to solve the instances. In particular, note that the CPU time used to solve the instance with $N = 8$, the largest one solved without the help of the symmetry-breaking constraints, was reduced from 11,186.81 seconds (see Table 1) to 16.05 seconds (see Table 2). The comparison between Tables 2 and 3 shows that it seems to be profitable to use strategy (i) instead of strategy (ii) to deal with constraints (8). Tables 1–3 suggest that the solver deals more efficiently with the original LGDP models than with any of their MIP reformulations, while the big-M reformulations can be solved with less effort than the convex-hull reformulations.

To complete the numerical experiments of the present subsection, we tested the efficiency of replacing, in the LGDP model (7), the non-overlapping constraints (2) by the non-overlapping plus symmetry-breaking constraints (11). Therefore, we tried to solve instances of the LGDP model $\mathcal{U}_{LGDP}^{k,[\Omega,h,w]}$ and its MIP reformulations $\mathcal{U}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{U}_{CH}^{k,[\Omega,h,w]}$, increasing the number of items $N \in \{1, 2, \ldots\}$. Table 4 shows the results. Figures in the table show that replacing the non-overlapping constraints (2) by the non-overlapping plus symmetry-breaking constraints (11) is much more effective than adding the symmetry breaking constraints (8,10) (see Tables 2 and 3), as instances with larger values of N were solved to optimality in the former case. Figure 4 shows a graphical representation of the solutions described in Table 4 for $\mathcal{U}_{BM}^{k,[\Omega,h,w]}$ varying the number of rectangles from 1 to 15.

4.2 Packing as many rectangles as possible

In this subsection we deal with the problem of, given an equilateral triangle of fixed dimensions, find the largest number N of identical rectangular items that can be packed within it. (Recall that rectangular items have height $h = 1$ and width $w = 0.5$.)

As detailed in Section 2, the strategy for packing as many rectangular items as possible consists of solving $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ (the feasibility LGDP model (1–3)), or any of its MIP reformulations $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$, for increasing values of k until a problem with k' items is proven to be infeasible. In practice, a time limit is imposed for this strategy. If, within the time limit, the infeasible problem is reached and solved (i.e. proven to be infeasible), then it is clear that

---

[2]It is worth noting that, for $N = 9$, the small reported relative gap corresponds to the solution for the case $r_1 = 0$ and $r_i = 1$, $i > 1$, while there was not enough time to consider most of the other feasible combinations of fixed values for the rotation variables r_i within the time limit.

		Instance data				Effort measurements		
	Fixed N	Triangle side		MIP GAP		MIP	B&B nodes	CPU Time
		Best lower bound	Best feasible solution	Relative	Absolute			
LGDP model (7,8,10)	1	1.577350269189625	1.577350269189625	0.00e+00	0.00e+00	0	0	0.01
	2	2.154700538379168	2.154700538379168	0.00e+00	0.00e+00	0	0	0.00
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00	143	16	0.02
	4	2.732050807568876	2.732050807568876	0.00e+00	0.00e+00	957	170	0.06
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00	7,787	1,879	0.28
	6	3.232050807568879	3.232050807568879	0.00e+00	0.00e+00	40,757	7,211	1.41
	7	3.443375672974065	3.443375672974064	2.58e−16	8.88e−16	90,126	17,075	3.31
	8	3.598076211353316	3.598076211353316	0.00e+00	0.00e+00	423,122	67,662	16.05
	9	3.809162326070802	3.809401076758502	6.27e−05	2.39e−04	14,817,741	2,466,016	590.69
	10	4.020399584377308	4.020725942163673	8.12e−05	3.26e−04	36,412,075	4,894,328	1,680.93
Big-M MIP reformulation	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.01
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00	0	0	0.00
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00	166	76	0.02
	4	2.732050807568878	2.732050807568878	0.00e+00	0.00e+00	921	463	0.04
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00	9,112	3,693	0.36
	6	3.232050807568878	3.232050807568878	0.00e+00	0.00e+00	34,669	13,086	1.25
	7	3.443375672974062	3.443375672974062	0.00e+00	0.00e+00	195,958	96,956	7.26
	8	3.598076211353316	3.598076211353316	0.00e+00	0.00e+00	1,242,459	600,426	51.12
	9	3.809252482546021	3.809401076758503	3.90e−05	1.49e−04	22,397,280	8,929,210	872.38
	10	4.020335713626388	4.020725942163690	9.71e−05	3.90e−04	191,614,100	81,788,232	10,443.75
Convex-hull MIP reform.	1	1.577350269189625	1.577350269189625	0.00e+00	0.00e+00	0	0	0.01
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00	59	0	0.00
	3	2.443375672974065	2.443375672974065	0.00e+00	0.00e+00	733	76	0.05
	4	2.732050807568877	2.732050807568877	0.00e+00	0.00e+00	4,883	686	0.18
	5	3.020725942163687	3.020725942163687	0.00e+00	0.00e+00	69,208	6,219	2.81
	6	3.232050807568884	3.232050807568884	0.00e+00	0.00e+00	357,205	28,210	14.42
	7	3.443375672973989	3.443375672973989	0.00e+00	0.00e+00	1,948,856	184,532	83.96
	8	3.597775882542360	3.598076211353317	8.35e−05	3.00e−04	12,018,350	1,126,677	581.89
	9	3.886402767942472	3.886751345948120	8.97e−05	3.49e−04	−	−	>21,600.00
	10	3.892320595579104	4.154700528069849	6.32e−02	2.62e−01	−	−	>21,600.00

Table 2: Instances of LGDP model $\mathcal{T}_{LGDP}^{k,[\Omega,h,w]}$ and its MIP reformulations $\mathcal{T}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{T}_{CH}^{k,[\Omega,h,w]}$ with increasing number of items $N \in \{1, 2, \dots\}$. The symmetry-breaking constraint (8) was considered by solving $k + 1$ instances of model (7,10), fixing the values of r_i according to (14) for $p = 0, \dots, k$.

the optimal solution is given by the one obtained for the problem with $N = k' - 1$ items. On the other hand, if the strategy is halted due to the time limit, two different possibilities exist regarding the last solved problem with, say, k'' items (assuming that $k'' \geq 1$): (a) k'' represents a suboptimal solution and a packing with at least $k'' + 1$ items exists, but there was not enough time to solve it; or (b) k'' is the maximum number of items that can be packed, but there was not enough time to prove that the instance with $k'' + 1$ is infeasible.

In the first set of experiments of this subsection, we considered ten different instances of $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$ (LGDP feasibility model (1–3)), $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$ (big-M reformulation), and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$ (convex-hull reformulation). In those ten instances, the immutable triangle side was fixed according to the best feasible solutions presented in Tables 1–4. In this way, a lower bound on the number of items that can be packed within each triangle is known. Table 5 shows some figures related to the solution of those problems. In the table, the CPU Time is presented in detail. The first column, named "Total" represents the total CPU time used to solve the problem or

		Instance data				Effort measurements		
	Fixed N	Triangle side		MIP GAP		MIP	B&B nodes	CPU Time
		Best lower bound	Best feasible solution	Relative	Absolute			
LGDP model (7,8,10)	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700538379252	2.154700538379252	0.00e+00	0.00e+00	5	0	0.00
	3	2.443375672974065	2.443375672974065	0.00e+00	0.00e+00	148	37	0.02
	4	2.732050807568881	2.732050807568881	0.00e+00	0.00e+00	781	195	0.03
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00	9,173	1,574	0.43
	6	3.232050807568879	3.232050807568879	0.00e+00	0.00e+00	29,822	5,752	1.48
	7	3.443375672974064	3.443375672974064	0.00e+00	0.00e+00	138,598	22,211	6.93
	8	3.597794231919681	3.598076211353336	7.84e−05	2.82e−04	536,140	83,428	29.21
	9	3.809064591545733	3.809401076758504	8.83e−05	3.36e−04	8,346,530	1,109,399	488.32
	10	4.020338430165712	4.020725942163692	9.64e−05	3.88e−04	71,062,703	10,825,791	5,238.60
Big-M MIP reformulation	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00	23	5	0.00
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00	119	54	0.01
	4	2.732050807568878	2.732050807568878	0.00e+00	0.00e+00	754	315	0.04
	5	3.020725942163691	3.020725942163690	4.41e−16	1.33e−15	5,564	2,231	0.23
	6	3.232050807568877	3.232050807568877	0.00e+00	0.00e+00	32,150	128,80	1.22
	7	3.443375672974065	3.443375672974065	0.00e+00	0.00e+00	131,457	62,731	5.99
	8	3.598076211353317	3.598076211353317	0.00e+00	0.00e+00	852,407	391,708	42.24
	9	3.809031272878700	3.809401076758504	9.71e−05	3.70e−04	37,824,870	12,474,040	2,935.49
	10	4.020324767277819	4.020725939008989	9.98e−05	4.01e−04	165,798,740	58,305,517	15,722.49
Convex-hull MIP reform.	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700534174552	2.154700534174552	0.00e+00	0.00e+00	63	15	0.01
	3	2.443374238386405	2.443375672363309	5.87e−07	1.43e−06	1,282	166	0.07
	4	2.732045702433151	2.732050806969059	1.87e−06	5.10e−06	12,153	1,227	0.43
	5	3.020505520742559	3.020725922875648	7.30e−05	2.20e−04	157,030	13,592	5.60
	6	3.231730502831837	3.232050797184711	9.91e−05	3.20e−04	1,936,538	139,754	76.34
	7	3.443031366455914	3.443375648663827	1.00e−04	3.44e−04	25,129,693	1,802,254	1,222.66
	8	3.597716312942202	3.598076064926102	1.00e−04	3.60e−04	−	−	>21,600.00
	9	3.207584396989211	3.886749550355905	1.75e−01	6.79e−01	−	−	>21,600.00
	10	3.105662430987886	4.020725345200177	2.28e−01	9.15e−01	−	−	>21,600.00

Table 3: Instances of LGDP model $\mathcal{T}_{LGDP}^{k,[\Omega,h,w]}$ and its MIP reformulations $\mathcal{T}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{T}_{CH}^{k,[\Omega,h,w]}$ with increasing number of items $N \in \{1, 2, \dots\}$. The symmetry-breaking constraint (8) was considered explicitly in the model.

elapsed until the time limit was reached. The three remaining columns represent: (i) the CPU time used to solve the instances with $1, \dots, N-1$ items; (ii) the CPU time used to solve the largest feasible instance with N items; and (iii) the CPU time used to solve the last, infeasible instance, with $N+1$ items. Note that only the first seven smaller instances were solved within the time limit (6 hours). In the remaining three cases, the optimal solution was in fact achieved, but the time limit was reached before proving the infeasibility of the last instance. Note that the time used to find the optimal solution is always very small and that the major part of the computation time is used to prove that the solution found is optimal. This observation justifies the sequential increase of N in contrast with a maybe more intuitive bisection approach.

Tables 6 and 7 show the solutions obtained by solving the same ten instances (same immutable triangles sides) of $\mathcal{Q}_{LGDP}^{k,[\Omega,h,w]}$ (LGDP model (1–3,8,10)), $\mathcal{Q}_{BM}^{k,[\Omega,h,w]}$ (big-M reformulation), and $\mathcal{Q}_{CH}^{k,[\Omega,h,w]}$ (convex-hull reformulation); and $\mathcal{R}_{LGDP}^{k,[\Omega,h,w]}$ (LGDP model (1,11,8)), $\mathcal{R}_{BM}^{k,[\Omega,h,w]}$ (big-M reformulation), and $\mathcal{R}_{CH}^{k,[\Omega,h,w]}$ (convex-hull reformulation), respectively. Figure 5 shows a

		Instance data				Effort measurements		
	Fixed N	Triangle side		MIP GAP		MIP	B&B nodes	CPU Time
		Best lower bound	Best feasible solution	Relative	Absolute			
LGDP model (7) replacing (2) by (11)	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00	5	0	0.00
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00	46	7	0.02
	4	2.732050807568879	2.732050807568879	0.00e+00	0.00e+00	94	10	0.00
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00	286	61	0.02
	6	3.232050807568878	3.232050807568878	0.00e+00	0.00e+00	738	173	0.08
	7	3.443375672974065	3.443375672974065	0.00e+00	0.00e+00	3,193	715	0.15
	8	3.598076211353317	3.598076211353317	0.00e+00	0.00e+00	16,115	2,847	0.75
	9	3.809401076758504	3.809401076758504	0.00e+00	0.00e+00	71,257	11,960	3.46
	10	4.020509285973630	4.020725942163690	5.39e−05	2.17e−04	203,721	41,891	12.03
	11	4.175304993828159	4.175426480542942	2.91e−05	1.21e−04	927,056	110,081	43.67
	12	4.386375364750972	4.386751345948129	8.57e−05	3.76e−04	3,244,798	736,824	156.11
	13	4.463659545857176	4.464101615137761	9.90e−05	4.42e−04	20,780,447	2,483,065	1,356.96
	14	4.597626257023785	4.598076211353318	9.79e−05	4.50e−04	84,549,515	12,186,953	7,119.75
	15	4.419443416399242	4.752776740536415	7.01e−02	3.33e−01	−	−	>21,600.00
	16	4.357521665581659	4.964101605673653	1.22e−01	6.07e−01	−	−	>21,600.00
Big-M MIP reformulation	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700538379251	2.154700538379251	0.00e+00	0.00e+00	14	0	0.00
	3	2.443375672974064	2.443375672974064	0.00e+00	0.00e+00	49	7	0.01
	4	2.732050807568878	2.732050807568878	0.00e+00	0.00e+00	105	13	0.00
	5	3.020725942163691	3.020725942163691	0.00e+00	0.00e+00	416	91	0.02
	6	3.232050807568878	3.232050807568878	0.00e+00	0.00e+00	1,619	373	0.06
	7	3.443216646206837	3.443375672974065	4.62e−05	1.59e−04	7,003	1,462	0.24
	8	3.598076211353316	3.598076211353316	0.00e+00	0.00e+00	14,739	2,622	0.54
	9	3.809401076758503	3.809401076758503	0.00e+00	0.00e+00	51,510	5,636	1.54
	10	4.020725942163691	4.020725942163691	0.00e+00	0.00e+00	486,468	77,816	16.25
	11	4.175088319554543	4.175426480542939	8.10e−05	3.38e−04	857,439	103,487	32.44
	12	4.386326500032685	4.386751345948128	9.68e−05	4.25e−04	6,235,708	695,202	244.93
	13	4.463655413028338	4.464101615137753	1.00e−04	4.46e−04	27,231,140	3,427,145	1,327.47
	14	4.597616595058138	4.598076211353316	1.00e−04	4.60e−04	80,702,664	5,283,372	3,402.18
	15	4.752303388377661	4.752776749732567	9.96e−05	4.73e−04	165,176,634	26,793,091	9,019.73
	16	4.318527513204312	4.964101607673653	1.30e−01	6.46e−01	−	−	>21,600.00
Convex-hull MIP reformulation	1	1.577350269189626	1.577350269189626	0.00e+00	0.00e+00	0	0	0.00
	2	2.154700538070694	2.154700538381463	1.44e−10	3.11e−10	40	7	0.02
	3	2.443375672645347	2.443375672645347	0.00e+00	0.00e+00	201	33	0.01
	4	2.732050806483621	2.732050806483621	0.00e+00	0.00e+00	1,019	139	0.07
	5	3.020725579211940	3.020725942196470	1.20e−07	3.63e−07	3,408	506	0.26
	6	3.231850013972345	3.232050807511775	6.21e−05	2.01e−04	12,039	1,458	0.61
	7	3.443202829554564	3.443375666133137	5.02e−05	1.73e−04	46,810	5,859	3.18
	8	3.597886815976836	3.598076209758018	5.26e−05	1.89e−04	123,158	13,064	8.20
	9	3.809272419509341	3.809401078416926	3.38e−05	1.29e−04	662,582	73,940	51.17
	10	4.020327421884938	4.020725794811397	9.91e−05	3.98e−04	4,930,428	54,196	445.95
	11	4.175027994557123	4.175426437952257	9.54e−05	3.98e−04	18,041,399	1,720,300	1,797.97
	12	4.366025218173216	4.386751328238087	4.72e−03	2.07e−02	−	−	>21,600.00
	13	4.166912549242940	4.464101448383035	6.66e−02	2.97e−01	−	−	>21,600.00
	14	3.963950046260996	4.598075130802665	1.38e−01	6.34e−01	−	−	>21,600.00
	15	3.787499813498732	4.752776715788993	2.03e−01	9.65e−01	−	−	>21,600.00
	16	3.845092857113551	4.964101312861107	2.25e−01	1.12e+00	−	−	>21,600.00

Table 4: Instances of LGDP model $\mathcal{U}_{LGDP}^{k,[\Omega,h,w]}$ and its MIP reformulations $\mathcal{U}_{BM}^{k,[\Omega,h,w]}$ and $\mathcal{U}_{CH}^{k,[\Omega,h,w]}$ with increasing number of items $N \in \{1, 2, \dots\}$.

graphical representation of the solutions described in Table 7 for $\mathcal{R}_{BM}^{k,[\Omega,h,w]}$ varying the number of rectangles from 1 to 14. It is easy to see that the symmetry-breaking constraints help a lot to
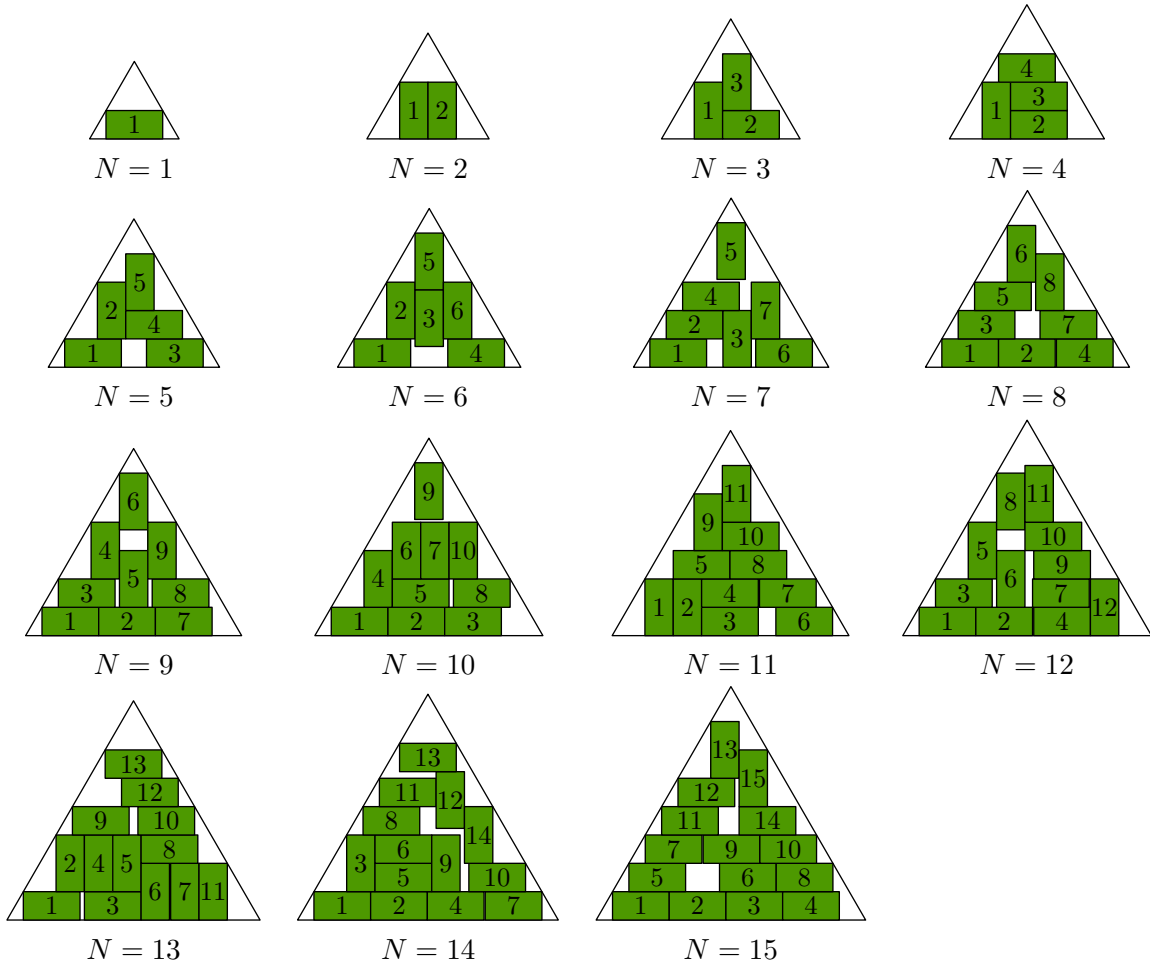
Figure 4: Graphical representation of the solutions to the problem of minimizing the size of an equilateral triangle that contains $N \in \{1, 2, \ldots, 15\}$ identical rectangles with height $h = 1$ and width $w = 0.5$. The figures correspond to the solutions to $\mathcal{U}_{BM}^{k,[\Omega,h,w]}$ described in Table 4. Note that they satisfy constraints in (11) or their reformulations (12) and (13).

drastically reduce the effort required to solve the instances. In fact, with their help, the optimal solutions of the ten instances were found and proven to be optimal and even larger instances were also solved to optimality. The comparison between the times displayed in Tables 5–7 is elucidating. When an instance with a fixed value k of items is feasible, it seems that eliminating symmetric solutions with the help of the symmetry-breaking constraints is not profitable. Note that CPU times for solving the instances with N items in Table 5 (which correspond to not using symmetry breaking constraints) are smaller than the ones displayed in Tables 6 and 7, which considers the symmetry-breaking constraints. The explanation for that observation seems to be simple. If a feasibility problem is feasible, the larger the number of feasibly (symmetric or not) feasible solutions, the faster a method (which aims to find a feasible solution) will find

Instance data			Effort measurements					
	Fixed triangle side	Best N	MIP	B&B nodes	CPU Time			
					Total	1 to $N-1$	N	$N+1$
LGDP model (1–3)	1.577350269189625	1	0	0	0.01	−	0.00	0.01
	2.154700529884166	2	112	15	0.01	0.00	0.00	0.01
	2.443375667541825	3	444	149	0.04	0.01	0.00	0.03
	2.732050717264398	4	17,882	3,270	0.63	0.01	0.00	0.62
	3.020725894730525	5	337,480	54,092	10.31	0.02	0.00	10.29
	3.232050397095424	6	1,702,397	288,180	64.57	0.02	0.02	64.53
	3.443373062238137	7	310,606,602	41,339,182	20,824.26	0.04	0.08	20,824.14
	3.598076064926102	8	−	−	>21,600.00	0.06	0.03	−
	3.809401068449104	9	−	−	>21,600.00	0.08	4.57	−
	4.020724784525934	10	−	−	>21,600.00	0.09	0.44	−
Big-M reformulation	1.577350269189625	1	0	0	0.00	−	0.00	0.00
	2.154700529884166	2	61	23	0.01	0.00	0.00	0.01
	2.443375667541825	3	484	213	0.02	0.00	0.00	0.02
	2.732050717264398	4	10,844	4,847	0.31	0.00	0.01	0.30
	3.020725894730525	5	340,644	133,09 5	9.21	0.00	0.01	9.20
	3.232050397095424	6	5,891,214	2,488,436	193.02	0.01	0.01	193.00
	3.443373062238137	7	346,797,622	132,731,792	12,484.64	0.02	0.02	12,484.60
	3.598076064926102	8	−	−	>21,600.00	0.04	0.36	−
	3.809401068449104	9	−	−	>21,600.00	0.05	0.57	−
	4.020724784525934	10	−	−	>21,600.00	0.05	0.22	−
Convex-hull reform.	1.577350269189625	1	51	1	0.01	−	0.00	0.01
	2.154700529884166	2	839	94	0.04	0.00	0.01	0.03
	2.443375667541825	3	11,087	1,016	0.30	0.01	0.01	0.28
	2.732050717264398	4	294,957	16,712	8.76	0.02	0.03	8.71
	3.020725894730525	5	10,026,367	632,425	301.85	0.03	0.13	301.69
	3.232050397095424	6	−	−	>21,600.00	0.07	1.53	−
	3.443373062238137	7	−	−	>21,600.00	0.13	2.77	−
	3.598076064926102	8	−	−	>21,600.00	1.31	5.91	−
	3.809401068449104	9	−	−	>21,600.00	0.60	871.80	−
	4.020724784525934	10	−	−	>21,600.00	0.93	221.04	−

Table 5: Instances of $\mathcal{P}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{P}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{P}_{CH}^{k,[\Omega,h,w]}$ setting the fixed side of the equilateral triangles as the best feasible solutions depicted in Tables 1–4.

one of them and stop declaring success. The opposite situation occurs when we compare the CPU times used to solve the infeasible problem with $N+1$ items in Tables 5–7. As instances are infeasible, the whole tree must be visited to prove the infeasibility. In this case, eliminating symmetric solutions makes the search tree smaller converting the task of proving infeasibility into an easier job. All in all, considering the symmetry-breaking constraints is profitable. Moreover, the best of both scenarios might be useful in an heuristic combined approach to rapidly find a probably optimal solution and then proving its optimality.

5 Conclusions

LGDP and MIP models for two packing problems were presented and symmetry-breaking constraints were introduced. The symmetry-breaking constraints do help, as expected, with the one exception of checking for feasibility/infeasibility when the problem is feasible. The main benefit of adding symmetry-breaking constraints, in terms of reduced computation time, comes when proving optimality or infeasibility. The non-rotated-first and the preorder symmetry-

	Instance data			Effort measurements				
						CPU Time		
	Fixed triangle side	Best N	MIP	B&B nodes	Total	1 to $N-1$	N	$N+1$
LGDP model (1,2,3,8,10)	1.577350269189625	1	0	0	0.01	–	0.00	0.01
	2.154700529884166	2	0	0	0.01	0.00	0.00	0.01
	2.443375667541825	3	0	0	0.03	0.00	0.01	0.02
	2.732050717264398	4	272	49	0.06	0.00	0.01	0.05
	3.020725894730525	5	10,604	1,575	0.39	0.01	0.07	0.31
	3.232050397095424	6	53,965	7,822	2.20	0.02	0.83	1.35
	3.443373062238137	7	151,158	21,720	5.14	0.05	0.86	4.50
	3.598076064926102	8	3,486,287	288,006	107.50	0.07	0.60	106.83
	3.809401068449104	9	20,550,301	1,819,760	666.90	0.10	196.57	469.23
	4.020724784525934	10	244,497,247	18,368,674	9,090.28	3.11	442.74	8,644.43
Big-M reformulation	1.577350269189625	1	0	0	0.00	–	0.00	0.00
	2.154700529884166	2	0	0	0.00	0.00	0.00	0.00
	2.443375667541825	3	222	64	0.04	0.00	0.01	0.03
	2.732050717264398	4	1,374	626	0.08	0.00	0.01	0.08
	3.020725894730525	5	20,295	6,363	0.66	0.00	0.11	0.55
	3.232050397095424	6	71,477	22,973	2.26	0.02	0.78	1.46
	3.443373062238137	7	348,952	104,069	10.32	0.02	1.06	9.24
	3.598076064926102	8	3,198,793	743,105	78.42	0.12	1.21	77.09
	3.809401068449104	9	20,967,920	4,103,489	591.68	0.31	159.70	431.67
	4.020724784525934	10	504,665,327	92,696,919	15,443.09	0.14	213.67	15,229.28
Convex-hull reform.	1.577350269189625	1	0	0	0.01	–	0.00	0.01
	2.154700529884166	2	141	0	0.02	0.00	0.00	0.02
	2.443375667541825	3	865	16	0.08	0.01	0.01	0.06
	2.732050717264398	4	12,495	712	0.47	0.00	0.00	0.47
	3.020725894730525	5	189,844	6,809	7.46	0.01	0.58	6.87
	3.232050397095424	6	888,823	31,810	34.58	0.04	11.32	23.22
	3.443373062238137	7	9,305,812	302,078	468.18	0.50	18.58	449.10
	3.598076064926102	8	110,802,449	2,211,931	9,489.72	0.91	27.68	9,461.13
	3.809401068449104	9	–	–	>21,600.00	3.41	–	–
	4.020724784525934	10	–	–	>21,600.00	1.68	–	–

Table 6: Instances of $\mathcal{Q}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{Q}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{Q}_{CH}^{k,[\Omega,h,w]}$ setting the fixed side of the equilateral triangles as the best feasible solutions depicted in Tables 1–4.

breaking constraints help, but the up-right ordering symmetry-breaking constraints seem to be the more useful ones as they simplify the non-overlapping constraints making the whole model much easier to be solved, at least for the problems considered. While the original LGDP models appear to be simpler for the solver in the minimization case, the big-M MIP reformulations seem to be more adequate in the feasibility case. In any case, the original LGDP models and their big-M MIP reformulations are more competitive than the corresponding convex-hull MIP reformulations.

References

[1] J. E. Beasley, An exact two-dimensional non-guillotine cutting tree-search procedure, *Operations Research* 33, pp. 49–64, 1985.

	Instance data		Effort measurements					
						CPU Time		
	Fixed triangle side	Best N	MIP	B&B nodes	Total	1 to $N-1$	N	$N+1$
LGDP model (1,11,3)	1.577350269189625	1	0	0	0.00	−	0.00	0.00
	2.154700529884166	2	3	0	0.00	0.00	0.00	0.00
	2.443375667541825	3	9	0	0.01	0.00	0.01	0.00
	2.732050717264398	4	159	15	0.02	0.00	0.01	0.01
	3.020725894730525	5	427	73	0.05	0.01	0.01	0.03
	3.232050397095424	6	634	157	0.07	0.01	0.02	0.04
	3.443373062238137	7	8,295	1,308	0.35	0.01	0.03	0.31
	3.598076064926102	8	14,626	2,915	0.60	0.04	0.03	0.53
	3.809401068449104	9	43,669	9,693	1.71	0.04	0.11	1.56
	4.020724784525934	10	611,197	64,388	20.51	0.06	0.03	20.42
	4.175426437952257	11	2,190,175	228,068	81.77	0.06	0.07	81.64
	4.386751328238087	12	5,299,011	987,112	212.68	0.08	0.14	212.46
	4.464101448383035	13	34,795,370	3,248,546	1,540.14	0.12	0.05	1,539.97
	4.598075130802665	14	205,848,392	13,957,106	10,299.73	0.17	0.54	10,299.01
	4.752776715788993	15	−	−	>21,600.00	0.22	0.07	−
Big-M reformulation	1.577350269189625	1	0	0	0.01	−	0.00	0.01
	2.154700529884166	2	19	0	0.00	0.00	0.00	0.00
	2.443375667541825	3	58	0	0.01	0.00	0.00	0.01
	2.732050717264398	4	126	18	0.03	0.02	0.00	0.01
	3.020725894730525	5	447	70	0.03	0.01	0.00	0.02
	3.232050397095424	6	805	175	0.04	0.01	0.00	0.04
	3.443373062238137	7	5,135	1,252	0.20	0.02	0.01	0.17
	3.598076064926102	8	29,736	3,209	0.79	0.03	0.02	0.74
	3.809401068449104	9	79,232	10,090	2.25	0.03	0.04	2.18
	4.020724784525934	10	519,162	52,811	14.35	0.03	0.02	14.30
	4.175426437952257	11	1,783,841	157,266	50.15	0.05	0.04	50.06
	4.386751328238087	12	10,102,510	1,249,609	358.40	0.04	0.11	358.25
	4.464101448383035	13	34,466,909	2,878,251	1,276.59	0.07	0.06	1,276.46
	4.598075130802665	14	122,262,705	16,673,839	3,636.51	0.10	0.04	3,636.37
	4.752776715788993	15	−	−	>21,600.00	0.13	0.03	−
Convex-hull reform.	1.577350269189625	1	27	0	0.01	−	0.00	0.01
	2.154700529884166	2	118	12	0.02	0.00	0.00	0.02
	2.443375667541825	3	401	44	0.06	0.01	0.01	0.04
	2.732050717264398	4	1,018	121	0.10	0.01	0.01	0.08
	3.020725894730525	5	6,824	721	0.32	0.02	0.05	0.25
	3.232050397095424	6	29,177	2,696	1.24	0.03	0.36	0.85
	3.443373062238137	7	119,510	8,104	5.46	0.06	0.39	5.01
	3.598076064926102	8	396,333	21,332	19.86	0.11	0.85	18.90
	3.809401068449104	9	2,136,966	88,392	120.17	0.15	1.84	118.18
	4.020724784525934	10	15,166,636	482,411	1,046.81	0.28	0.74	1,045.79
	4.175426437952257	11	46,304,928	1,438,015	4,317.87	0.46	2.01	4,315.40
	4.386751328238087	12	−	−	>21,600.00	0.71	17.03	−
	4.464101448383035	13	−	−	>21,600.00	0.80	9.98	−
	4.598075130802665	14	−	−	>21,600.00	1.36	48.78	−
	4.752776715788993	15	−	−	>21,600.00	6.80	37.14	−

Table 7: Instances of $\mathcal{R}_{LGDP}^{k,[\Omega,h,w]}$, $\mathcal{R}_{BM}^{k,[\Omega,h,w]}$, and $\mathcal{R}_{CH}^{k,[\Omega,h,w]}$ setting the fixed side of the equilateral triangles as the best feasible solutions depicted in Tables 1–4.

[2] E. G. Birgin and R. D. Lobato, Orthogonal packing of identical rectangles within isotropic convex regions, *Computers & Industrial Engineering* 59, pp. 595–602, 2010.

[3] E. G. Birgin, R. D. Lobato and R. Morabito, An effective recursive partitioning approach for the packing of identical rectangles in a rectangle, *Journal of the Operational Research*
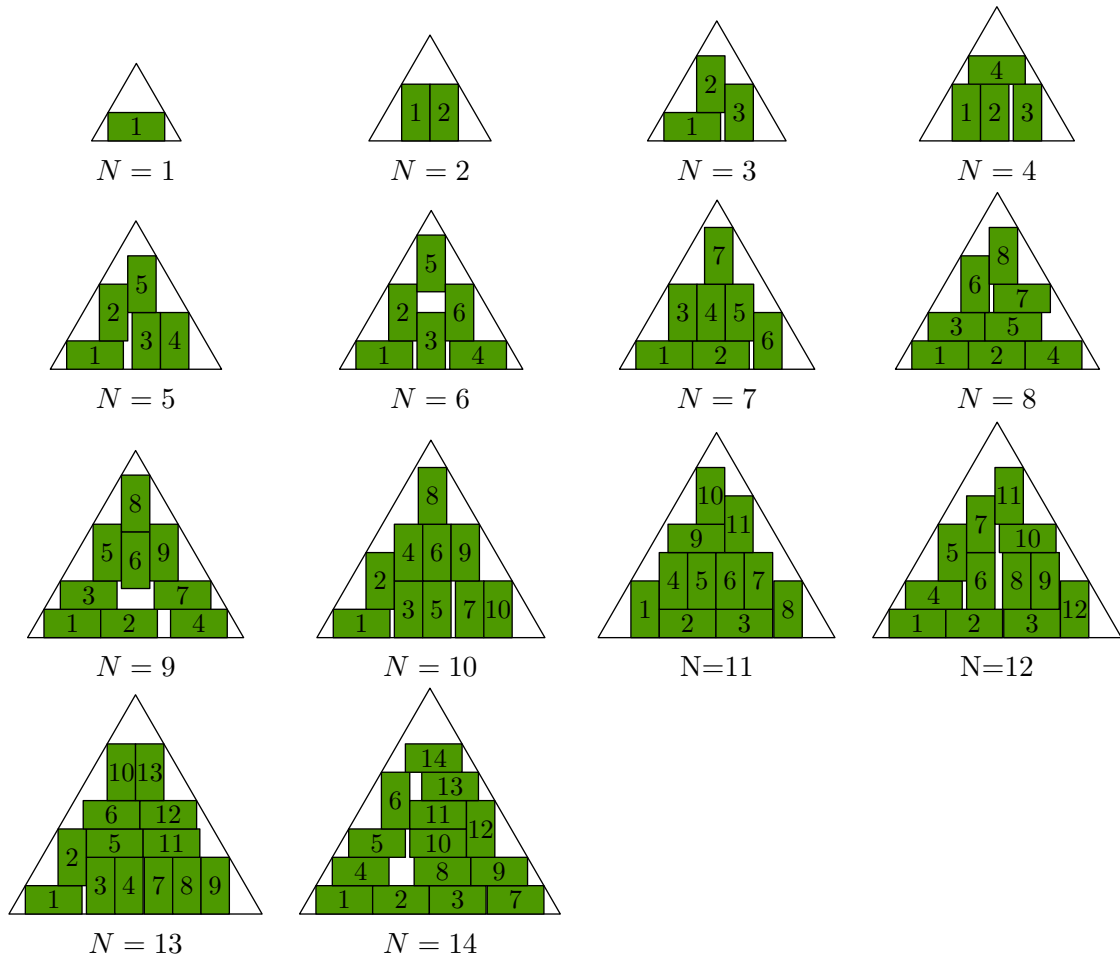
Figure 5: Graphical representation of the solutions to the problem of packing as many identical rectangles as possible with height $h = 1$ and width $w = 0.5$ within an equilateral triangles of fixed size. The figures correspond to the solutions to $\mathcal{R}_{BM}^{k,[\Omega,h,w]}$ described in Table 7. Note that they satisfy constraints in (11) or their reformulations (12) and (13).

Society 61, pp. 306–320, 2010.

[4] E. G. Birgin, R. D. Lobato and R. Morabito, Generating unconstrained two-dimensional non-guillotine cutting patterns by a recursive partitioning algorithm, *Journal of the Operational Research Society*, doi:10.1057/jors.2011.6.

[5] E. G. Birgin, J. M. Martínez, W. F. Mascarenhas and D. P. Ronconi, Method of Sentinels for Packing Items within Arbitrary Convex Regions, *Journal of the Operational Research Society* 57, pp. 735–746, 2006.

[6] E. G. Birgin, R. Morabito and F. H. Nishihara A note on an L-approach for solving the

manufacturer's pallet loading problem, *Journal of the Operational Research Society* 56, pp. 1448–1451, 2005.

[7] E. G. Birgin, J. M. Martínez, F. H. Nishihara and D. P. Ronconi, Orthogonal packing of rectangular items within arbitrary convex regions by nonlinear optimization, *Computers & Operations Research* 33, pp. 3535–3548, 2006.

[8] F. Cao, D.-Z. Du, B. Gao, P.-J. Wan, and P.M. Pardalos, Minimax Problems in Combinatorial Optimization, in *Minimax and Applications*, D.-Z. Du and P.M. Pardalos (Eds.), Kluwer Academic Publishers, 1995, pp. 262–285.

[9] A. Cassioli and M. Locatelli, A heuristic approach for packing identical rectangles in convex regions, *Computers & Operations Research* 38, pp. 1342–1350, 2011.

[10] C. A. Floudas and P. M. Pardalos (Eds.), *Encyclopedia of Optimization*, 2nd ed., Springer, 2009.

[11] C. A. J. Hurkens, A. Lodi, S. Martello, M. Monaci, and G. J. Woeginger, Complexity and approximation of an area packing problem, *Optimization Letters*, to appear (DOI: 10.1007/s11590-010-0246-2).

[12] J. Kallrath, Cutting circles and polygons from area-minimizing rectangles, *Journal of Global Optimization* 43, pp. 299–328, 2009.

[13] A. Lodi and M. Monaci, Integer linear programming models for 2-staged two-dimensional Knapsack problems, *Mathematical Programming* 94, pp. 257–278, 2003.

[14] W. F. Mascarenhas and E. G. Birgin, Using sentinels to detect intersections of convex and nonconvex polygons, *Computational & Applied Mathematics* 29, pp. 247–267, 2010.

[15] J. Ostrowski, J. Linderoth, F. Rossi and S. Smirglio, Solving Steiner triple covering problems, *Otima* 83, pp. 1–7, 2010.

[16] N. W. Sawaya and I. E. Grossmann, A cutting plane method for solv- ing linear generalized disjunctive programming problems, *Computers & Chemical Engineering* 29, pp. 1891–1913, 2005.

[17] W. Xie and N. V. Sahinidis, A branch-and-bound algorithm for the continuous facility layout problem, *Computers & Chemical Engineering* 32, pp. 1016–1028, 2008.