RAIRO-Oper. Res. 52 (2018) 391–414
https://doi.org/10.1051/ro/2017019

RAIRO Operations Research
www.rairo-ro.org

A THEORETICAL AND EXPERIMENTAL STUDY OF FAST LOWER BOUNDS FOR THE TWO-DIMENSIONAL BIN PACKING PROBLEM

Mehdi Serairi[1] and Mohamed Haouari[2]

Abstract. We address the two-dimensional bin packing problem with fixed orientation. This problem requires packing a set of small rectangular items into a minimum number of standard two-dimensional bins. It is a notoriously intractable combinatorial optimization problem and has numerous applications in packing and cutting. The contribution of this paper is twofold. First, we propose a comprehensive theoretical analysis of lower bounds and we elucidate dominance relationships. We show that a previously presented dominance result is incorrect. Second, we present the results of an extensive computational study that was carried out, on a large set of 500 benchmark instances, to assess the empirical performance of the lower bounds. We found that the so-called Carlier-Clautiaux-Moukrim lower bounds exhibits an excellent relative performance and yields the tightest value for all of the benchmark instances.

Mathematics Subject Classification. 90-08.

Received March 24, 2016. Accepted March 16, 2017.

1. Introduction

The *two-dimensional bin packing problem* ($2BPP$) is defined as follows. Given a set J of n rectangular items where each item j ($j = 1, \dots, n$) has a width w_j and a height h_j, a set of n identical rectangular bins where each bin is characterized by a width W and a height H, the $2BPP$ requires packing, without overlapping, the set of items into a minimum number of bins. The version where items cannot be rotated is considered. This problem is $\mathcal{NP}$-hard since it is a generalization of the much-studied one-dimensional bin packing ($1BPP$). Indeed, the particular case where the inequality $w_j > \frac{W}{2}$ holds for all $j \in J$ trivially reduces to a one-dimensional bin packing problem. In fact, the BPP and its two dimensional variation do have the same search space. The $2BPP$ has a wealth of pertinence to a wide range of applied areas including wood, glass, and steel industries, to quote just a few.

So far, several authors have investigated the $2BPP$. Exact methods can be found in Martello and Vigo [34], Clautiaux *et al.* [12] and Pisinger and Sigurd [35]. Moreover, exact approaches for the single-bin variant, which is referred to as the *two-dimensional orthogonal packing problem* ($2OPP$), were proposed by Hadjiconstantinou and Christofides [24], Clautiaux *et al.* [13], and Fekete *et al.* (2006). In addition to the exact methods, several

Keywords. Two-dimensional bin packing, lower bounds, dual feasible functions, dominance results.

[1] Sorbonne universités, Université de technologie de Compiègne, CNRS, Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne Cedex, France. mehdi.serairi@hds.utc.fr

[2] Department of Mechanical and Industrial Engineering, College of Engineering, Qatar University, Doha, Qatar. mohamed.haouari@qu.edu.qa

heuristics and meta-heuristics were developed for the $2BPP$. We refer to the excellent survey papers of Lodi *et al.* [29, 30] for a comprehensive review of approximation algorithms and heuristic approaches that were proposed up to the late 1990s. Recently, Cui *et al.* [16] proposed a sequential heuristic procedure that was shown to outperform five published algorithms. Also, Hong *et al.* [25] proposed a hybrid simulated annealing algorithm for the the variable-sized problem variant.

Furthermore, several authors addressed numerous variants of the $2BPP$. We provide a concise description of the most relevant contributions in this area.

- *The two-dimensional strip packing problem*: This problem requires orthogonally packing a given set of rectangular items into a strip, by minimizing the overall height of the packing. Exact approaches for this problem were extensively studied by many authors including Martello *et al.* [31], Cintra *et al.* [11], Kenmochi *et al.* [26], Boschetti and Montaletti [5], and Côté *et al.* [15]. Recently, Wei et al. [36] addressed the special variant of this problem with guillotine-cut constraint. They proposed a heuristic approach that requires iteratively packing horizontal layers.
- *The two-dimensional knapsack packing problem*: In this case, it is assumed that a non-negative weight (profit) is associated with each rectangular item, and the problem is to orthogonally pack a maximum-profit subset of items into a single rectangular knapsack. Hadjiconstantinou and Christofides [24] proposed an exact algorithm for this problem, while Egeblad and Pisinger [18], and Bortfeldt and Winter[3] proposed heuristic approaches. Furthermore, Caprara *et al.*[7] proposed an approximation scheme.
- *The two-dimensional loading vehicle routing problem*: This rich vehicle routing problem requires distributing two-dimensional items to a set of scattered customers. It involves two main decisions: loading the items into the vehicles (that can be viewed as two-dimensional bins) and designing the vehicle routes. Several authors addressed this challenging problem including Gendreau *et al.* [23], Fuellerer *et al.* [22], Zachariadis *et al.*[38]), Iori and Martello [27], and Wei *et al.* [37].

Furthermore, several additional 2D packing problems have been investigated in the operations research literature (though receiving relatively much less attention). A non exhaustive list includes the problem that requires packing, with no overlapping, a set of rectangles into the smallest square (Martello and Monaci [32]), and the two-dimensional vector packing problem (Alves *et al.* [1]) where a set of items with two *independent* dimensions must be packed into two-dimensional bins with *independent* dimensions.

In this paper, we focus on lower bounds for the $2BPP$. With few exceptions, all the lower bounds that we shall discuss were not thoroughly investigated in the literature. The objective of this paper is twofold. First, we provide an updated comprehensive theoretical study of lower bounds for the $2BPP$ with an emphasis on polynomial bounds that can be efficiently computed. Toward this end, we propose a classification of the lower bounds and point out the relation between proposed lower bounds and dual feasible functions. Furthermore, we show that the dominance relation claimed by Carlier *et al.* [9] of their lower bounds is incorrect by providing a counterexample. Second, we provide the results of a comprehensive computational study that was carried out to assess the empirical performance of the lower bounds.

The remainder of this paper is organized as follows. Section 2 includes a detailed description and analysis of the lower bounds that were proposed so far. In Section 3, we analyze dominance relationships between $2BPP$ lower bounds. In Section 4, we report the results of a comprehensive computational study that was carried to assess the computational performance of the different lower bounds. Finally, some concluding remarks and directions for future research are provided in Section 5.

2. Lower bounds for the 2BPP

In this section, we provide an updated theoretical study of polynomial $2BPP$ lower bounds that were proposed so far. Before proceeding further we introduce the following notation that will be used throughout this paper.

- $L^d_{XY,q}$ refers to the q^{th} d-dimensional bin packing lower bound that was originally described in the paper whose authors' initials are X and Y, respectively.

- I_1 denotes an instance of the one-dimensional bin-packing problem that requires packing a set S of items, each item being characterized by a weight c_j, into a set of identical bins of capacity C.
- $L(I)$ denotes the value of the bound delivered by a lower bounding procedure $L(.)$ for an instance I.
- $L_0^1(I_1) \equiv \lceil \sum_{j\in S} c_j/C \rceil$ is the so-called continuous $1BPP$ bound.
- L_{MV}^1 refers to the lower bound proposed by Martello and Vigo [34]. This bound is based on the bounds of Martello and Toth [33] and Dell'Amico and Martello [17], respectively. For the sake of completeness, we provide a description of L_{MV}^1 along the following lines. Let I be an $1BPP$ instance, and p an integer such that $1 \le p \le \frac{C}{2}$. We define $S_1(p), S_2(p)$ and $S_3(p)$ as, respectively, the following subsets: $S_1(p) = \{j \in S : c_j > C - p\}, S_2(p) = \{j \in S \backslash S_1(p) : c_j > \frac{C}{2}\}$ and $S_3(p) = \{j \in S \backslash (S_1(p) \cup S_2(p)) : c_j \ge p\}$. Then, we define:

$$L_{MV}^1 \equiv \max\{L_\alpha, L_\beta\},$$

where

$$L_\alpha = \max_{1\le p\le \frac{C}{2}} \left\{ |S_1(p) \cup S_2(p)| + \max\left\{0, \left\lceil \frac{\sum_{j\in(S_2(p)\cup S_3(p))} c_j}{C} - |S_2(p)| \right\rceil \right\} \right\}$$

$$L_\beta = \max_{1\le p\le \frac{C}{2}} \left\{ |S_1(p) \cup S_2(p)| + max\left\{0, \left\lceil \frac{|S_3(p)| - \sum_{j\in S_2(p)} \left\lfloor \frac{C-c_j}{p} \right\rfloor}{\left\lfloor \frac{C}{p} \right\rfloor} \right\rceil \right\} \right\}$$

In the sequel, and for the sake of convenience, we shall partition the set of lower bounds that will be discussed into three classes depending on how both dimensions are handled. More precisely, we define the following classes:

- **Class 1.** These bounds consider only one dimension at one time. An $1BPP$-based lower bound is computed by considering just one specified dimension. Next, the same process is repeated by considering the second dimension.
- **Class 2.** These bounds consider both dimensions simultaneously.
- **Class 3.** These bounds are based on an appropriate transformation of the genuine heights and widths into new ones and then computing a lower bound using these modified dimensions.

Remark 2.1. In addition to the discussed polynomial lower bounds, non-polynomial lower bounding procedures were proposed as well. In particular, Pisinger and Sigurd [35] proposed a column-generation-based lower bound and Caprara and Monaci [8] proposed bi-linear programming-based lower bounds. Not surprisingly, these bounds require a significant computational burden and will be omitted in this survey. Actually, while the considered polynomial lower bounds needs few milliseconds, the computation times of the lower bounds of Pisinger and Sigurd [35] and Caprara and Monaci [8] are of the order of several seconds.

2.1. The continuous bound

To begin with, we introduce a simple $O(n)$ continuous bound that is defined as follows:

$$L_0^2 = \left\lceil \frac{\sum_{j\in J} h_j w_j}{HW} \right\rceil \tag{2.1}$$

Martello and Vigo [34] show that the worst-case performance ratio of L_0^2 is $\frac{1}{4}$.

2.2. Lower bounds of Class 1

2.2.1. $L^2_{MV,1}$ lower bound

A first bound proposed by Martello and Vigo [34] is described as follows. Define an $1BPP$ instance I^1_w that is constructed as follows: $S_w = \left\{j \in J : w_j > \frac{W}{2}\right\}$ is the set of items, the weight of an item j is $c_j = h_j$, and the bin capacity is $C = H$. Since in any feasible solution no two items from S_w can be packed side by side then an $1BPP$ lower bound $L(I^1_w)$ is also a valid lower bound for the original $2BPP$ instance. Similarly, by interchanging the roles of the widths and the heights, we can derive a symmetric bound $L(I^1_h)$. In their implementation, Martello and Vigo [34] used L^1_{MV} as the $1BPP$ lower bound. This yields the $O(n^2)$ lower bound:

$$L^2_{MV,1} = \max\left(L^1_{MV}(I^1_w), L^1_{MV}(I^1_h)\right) \tag{2.2}$$

Martello and Vigo [34] show that no dominance relation exists between L^2_0 and $L^2_{MV,1}$.

2.2.2. $L^2_{MV,2}$ and $L^2_{BM,1}$ lower bounds

The main idea of these two lower bounds is based on the fact that the large items cannot be packed side by side. Therefore a one-dimensional bin packing lower bound is invoked to estimate the number of bins that should be considered to pack the subset of large items. It follows that the remaining items should be packed into a bin with a large item or into new one.

Formally, let q be an integer in $[0, W/2]$, we consider the following two subsets:

$$J^w_1(q) = \{j \in J : w_j > W - q\}$$
$$J^w_2(q) = \{j \in J : q \le w_j \le W - q\}$$

Martello and Vigo [34] define $L^{2,w}_{MV,2}(q)$ as follows:

$$L^{2,w}_{MV,2}(q) = L^1_{MV}(I^1_w) + \max\left\{0, \left\lceil \frac{\sum_{j\in J^w_2(q)} h_j w_j - \left(HL^1_{MV}(I^1_w) - \sum_{j\in J^w_1(q)} h_j\right) W}{HW} \right\rceil\right\} \tag{2.3}$$

This yields the lower bound:

$$L^{2,w}_{MV,2} = \max_{1\le q\le \frac{W}{2}} L^{2,w}_{MV,2}(q). \tag{2.4}$$

Similarly, an analogous lower bound $L^{2,h}_{MV,2}$ can be obtained by exchanging the widths with the heights. Finally, $L^2_{MV,2}$ is given by:

$$L^2_{MV,2} = \max\left(L^{2,w}_{MV,2}, L^{2,h}_{MV,2}\right) \tag{2.5}$$

Martello and Vigo [34] show that $L^2_{MV,2}$ can be computed in $O(n^2)$ time. Indeed, in $L^{2,w}_{MV,2}$ instance I^1_w does not depend on the value of q, and at most n values of q that correspond to distinct values of w_j should be considered.

Moreover, since we have $L^2_{MV,2} \ge L^1_0(I^1_w(1)) \equiv L^2_0$ and $L^2_{MV,2} \ge \max(L^1_{MV}(I^1_w), L^1_{MV}(I^1_h)) \equiv L^2_{MV,1}$, $L^2_{MV,2}$ dominates both L^2_0 and $L^2_{MV,1}$.

Boschetti and Mingozzi [6] introduced a lower bound $L^2_{BM,1}$. Since in any feasible solution no item $j \in J_1(q)$ can be packed side by side with an item from $J_1(q) \cup J_2(q)$ therefore the width of item j can be increased to W. Consider an $1BPP$ instance $I^1_w(q)$ that is defined as follows:

$$S = J^w_1(q) \cup J^w_2(q),\ C = WH \text{ and } c_j = \begin{cases} Wh_j & \text{if} \quad j \in J^w_1(q) \\ w_j h_j & \text{if} \quad j \in J^w_2(q) \\ 0 & \text{otherwise} \end{cases} \tag{2.6}$$

Clearly, $L(I_w^1(q))$ is a valid lower bound for the original $2BPP$ instance. Thus the following lower bound is valid.

$$L_{BM,1}^{2,w} = \max_{1 \le q \le \frac{W}{2}} L_{BM,1}^{2,w}(q). \tag{2.7}$$

Here again, only the value of q corresponding to distinct values of w_j should be considered to compute $L_{BM,1}^{2,w}$. Therefore the considered one-dimensional lower bound is invoked n times.

The symmetric lower bound $L_{BM,1}^{2,h}$ is derived in a similar way. Finally, we get:

$$L_{BM,1}^{2} = \max\left(L_{BM,1}^{2,w}, L_{BM,1}^{2,h}\right) \tag{2.8}$$

In their implementation, Boschetti and Mingozzi [6] used L_{MV}^1 as the $1BPP$ lower bound. In this case they show that $L_{BM,1}^2$ can be computed in $O(n^3)$ time since $L_{BM,1}^2$ is invoked $2n$ times as explained above.

Actually, it is easy to see that $L_{BM,1}^2$ can be viewed as an improved variant of $L_{MV,2}^2$. Indeed, we can restate $L_{MV,2}^2$ in the following way:

$$L_{MV,2}^{2,w}(q) = \max\left\{ L_{MV}^1(I_w^1), \left\lceil \frac{\sum\limits_{j \in J_2^w(q)} h_j w_j + \sum\limits_{j \in J_1^w(q)} h_j W}{HW} \right\rceil \right\} \tag{2.9}$$

Hence,

$$L_{MV,2}^{2,w}(q) = \max(L_{MV}^1(I_w^1), L_0^1(I_w^1(q))) \tag{2.10}$$

Thus, $L_{MV,2}^{2,w}$ is given by

$$L_{MV,2}^{2,w} = \max(L_{MV}^1(I_w^1), \max_{0 \le q \le \frac{W}{2}} L_0^1(I_w^1(q))) \tag{2.11}$$

By replacing $L_0^1(.)$ by $L_{MV}^1(.)$, we obtain $L_{BM,1}^2$. Clearly, $L_{BM,1}^2$ dominates $L_{MV,2}^2$.

2.3. Lower bounds of Class 2

2.3.1. $L_{BM,2}^2$ lower bound

Given two integers p and q such that $1 \le p \le \frac{H}{2}$ and $1 \le q \le \frac{W}{2}$, define the following subsets

$$J_{\text{Large}}(p,q) = \{j \in J : w_j > W - q \text{ and } h_j > H - p\}$$

$$J_{\text{Tall}}(p,q) = \{j \in J \backslash J_{\text{Large}}(p,q) : w_j \ge q \text{ and } h_j > H - p\}$$

$$J_{\text{Wide}}(p,q) = \{j \in J \backslash J_{\text{Large}}(p,q) : w_j > W - q \text{ and } h_j \ge p\}$$

$$J_{\text{Small}}(p,q) = \{j \in J \backslash (J_{\text{Large}}(p,q) \cup J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q)) : w_j \ge q \text{ and } h_j \ge p\}$$

Obviously, each item of $J_{\text{Large}}(p,q)$ requires a separate bin. Hence, $|J_{\text{Large}}(p,q)|$ is a trivial valid lower bound. Moreover, there is no item from $j \in J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$ that can be packed together with an item from $J_{\text{Large}}(p,q)$. Consequently, the value $|J_{\text{Large}}(p,q)|$ can be tightened by computing a lower bound on the number of bins required for packing the items of $J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$. Boschetti and Mingozzi [6] introduced two methods for computing such a lower bound. These methods are based on transforming the $2BPP$ instance, composed by the items of $J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$ into an $1BPP$ instance.

First method. Let two items i and j such that $j \in J_{\text{Wide}}(p,q)$ and $i \in J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$. Since $w_i + w_j > W$ then in any feasible packing there exists no pair $\{i,j\}$ of items such that

$i \in J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$ and $j \in J_{\text{Wide}}(p,q)$ and i is packed side by side with item j. Therefore, the width of item j can be increased to W. Using similar arguments, we observe that the height of any item $k \in J_{\text{Tall}}(p,q)$ can be increased to H. Consider the $1BPP$ instance $I_1^{\alpha}(p,q)$ that is defined as follows:

$$S = J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$$

$$C = HW$$

$$c_j = \begin{cases} Hw_j & \text{if} \quad j \in J_{\text{Tall}}(p,q) \\ h_j W & \text{if} \quad j \in J_{\text{Wide}}(p,q) \\ h_j w_j & \text{if} \quad j \in J_{\text{Small}}(p,q) \end{cases} \tag{2.12}$$

Clearly, $L^1_{MV}(I_1^{\alpha}(p,q))$ is a valid lower bound on the minimal number of bins that are required for packing the items that belong to $J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$.

Second method. Let i and j be two items such that $i \in J_{\text{Wide}}(p,q)$ and $j \in J_{\text{Tall}}(p,q)$. Since $w_i + w_j > W$ and $h_i + h_j > H$, it follows that in a feasible solution items i and j can not be packed together in the same bin. Thus, a valid lower bound on the number of bins that are required for packing the items of $J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$ is the sum of the lower bound on the number of bins that are required for packing the items of $J_{\text{Wide}}(p,q)$ and the lower bound on the number of bins that are required for packing the items of $J_{\text{Tall}}(p,q)$. Furthermore, we observe that there are no two items from $J_{\text{Wide}}(p,q)$ (respectively, $J_{\text{Tall}}(p,q)$) that can be packed side by side (respectively, one above the other) in the same bin. Therefore, we should consider the following two $1BPP$ instances.

- $I_1^{\beta}(p,q) : S = J_{\text{Wide}}(p,q), c_j = h_j$, and $C = H$
- $I_1^{\gamma}(p,q) : S = J_{\text{Tall}}(p,q)$, $c_j = w_j$, and $C = W$

Consequently, a second valid lower bound on the minimal number of bins that are required for packing the items that belong to $J_{\text{Tall}}(p,q) \cup J_{\text{Wide}}(p,q) \cup J_{\text{Small}}(p,q)$ is:

$$L^1_{MV}(I_1^{\beta}(p,q)) + L^1_{MV}(I_1^{\gamma}(p,q)). \tag{2.13}$$

Finally, lower bound $L^2_{BM,2}$ is

$$L^2_{BM,2} = \max_{1 \le p \le \frac{H}{2}, 1 \le q \le \frac{W}{2}} L^2_{BM,2}(p,q) \tag{2.14}$$

where,

$$L^2_{BM,2}(p,q) = |J_{\text{Large}}(p,q)| + \max(L^1_{MV}(I_1^{\alpha}(p,q)), L^1_{MV}(I_1^{\beta}(p,q)) + L^1_{MV}(I_1^{\gamma}(p,q))) \tag{2.15}$$

The time complexity of $L^2_{BM,2}$ is $O(n^4)$. Boschetti and Mingozzi [6] show that $L^2_{BM,2}$ dominates $L^2_{BM,1}$.

2.3.2. $L^2_{MV,3}$, $L^2_{BM,3}$ and $L^2_{BM,4}$ lower bounds

These lower bounds take into account both dimensions simultaneously. Given two integers p and q such that $1 \le p \le \frac{H}{2}$ and $1 \le q \le \frac{W}{2}$, define the following subsets:

- $J_{\text{Large}}(p,q) = \{j \in J : w_j > W - q \text{ and } h_j > H - p\}$
- $J_{\text{Medium}}(p,q) = \left\{j \in J \backslash J_{\text{Large}}(p,q) : w_j > \frac{W}{2} \text{ and } h_j > \frac{H}{2}\right\}$

Given two items i and k such that

$$i, k \in J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q).$$

Then these two items cannot be packed together into the same bin. Thus, $|J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)|$ is a valid lower bound. Furthermore, these lower bounds consider a subset of the remaining items and compute a lower bound on the number of bins that are required to pack the items of this subset. Clearly, items from $J \backslash (J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q))$ can be packed into a bin that has been initialized with an item from $J_{\text{Medium}}(p,q)$ or into an empty bin.

Martello and Vigo lower bound $\mathbf{L^2_{MV,3}}$: Martello and Vigo [34] considered the subset:

$$J_{s1}(p,q) = \left\{ j \in J : q \le w_j \le \frac{W}{2} \text{ and } p \le h_j \le \frac{H}{2} \right\}$$

They considered that each item of $J_{s1}(p,q)$ is a piece of size $(p \times q)$. Thus $|J_{s1}(p,q)|$ pieces have to be packed into the $|J_{\text{Medium}}(p,q)|$ bins that have been initialized with an item from $J_{\text{Medium}}(p,q)$ or in an empty bin. The number of pieces $m(j,p,q)$ that can be packed into a bin that has been initialized with an item $j \in J_{\text{Medium}}(p,q)$ is given by:

$$m(j,p,q) = \left\lfloor \frac{H}{p} \right\rfloor \left\lfloor \frac{W - w_j}{q} \right\rfloor + \left\lfloor \frac{W}{q} \right\rfloor \left\lfloor \frac{H - h_j}{p} \right\rfloor - \left\lfloor \frac{H - h_j}{p} \right\rfloor \left\lfloor \frac{W - w_j}{q} \right\rfloor \tag{2.16}$$

Furthermore, the maximal number of pieces that can be packed into an empty bin is given by $\left\lfloor \frac{H}{p} \right\rfloor \left\lfloor \frac{W}{q} \right\rfloor$

Thus, lower bound $L^2_{MV,3}$ is given by

$$\begin{aligned} L^2_{MV,3} = \max_{1 \le p \le \frac{H}{2}, 1 \le q \le \frac{W}{2}} & |J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)| \\ & + \max\left(0, \left\lceil \frac{|J_{s1}(p,q)| - \sum_{j \in J_{\text{Medium}}(p,q)} m(j,p,q)}{\left\lfloor \frac{H}{p} \right\rfloor \left\lfloor \frac{W}{q} \right\rfloor} \right\rceil \right) \end{aligned} \tag{2.17}$$

Martello and Vigo [34] show that $L^2_{MV,3}$ can be computed in $O(n^3)$ time and no dominance relation exists between $L^2_{MV,2}$ and $L^2_{MV,3}$.

Boschetti and Mingozzi lower bound $\mathbf{L^2_{BM,4}}$: To calculate a lower bound on the number of bins required to pack the subset $J_s(p,q) = \{j \in J \backslash (J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)) : h_j \ge p, w_j \ge q\}$, $L^2_{BM,4}$ follows a similar approach to $L^2_{MV,3}$. Indeed each item in $J_s(p,q)$ is composed by a number a pieces of size $(p \times q)$, Boschetti and Mingozzi [6] determine a lower bound on the number of pieces that compose an item $j \in J_s(p,q)$. Recall that Martello and Vigo [34] used a trivial lower bound that is equal to 1 for each item in $J_{s1}(p,q)$. In addition to the subset $J_{s1}(p,q)$, Boschetti and Mingozzi [6] considered two additional subsets:

$$J_{s2}(p,q) = \left\{ j \in J : p \le h_j \le \frac{H}{2} \qquad \text{and } w_j > \frac{W}{2} \right\}$$

$$J_{s3}(p,q) = \left\{ j \in J : q \le w_j \le \frac{W}{2} \qquad \text{and } h_j > \frac{H}{2} \right\}$$

The lower bound on the number of pieces that compose an item $j \in J_s(p,q)$ is given by (2.18)

$$m'(j,p,q) = \begin{cases} \left\lfloor \frac{h_j}{p} \right\rfloor \left\lfloor \frac{w_j}{q} \right\rfloor & \text{if} \quad j \in J_{s1}(p,q) \\ \left(\left\lfloor \frac{W}{q} \right\rfloor - \left\lfloor \frac{W - w_j}{q} \right\rfloor \right) \left\lfloor \frac{h_j}{p} \right\rfloor & \text{if} \quad j \in J_{s2}(p,q) \\ \left(\left\lfloor \frac{H}{p} \right\rfloor - \left\lfloor \frac{H - h_j}{p} \right\rfloor \right) \left\lfloor \frac{w_j}{q} \right\rfloor & \text{if} \quad j \in J_{s3}(p,q) \end{cases} \tag{2.18}$$

In the same way as for $L^2_{MV,3}$, the number of pieces that can be packed into a bin that is initialized with an item j such that $j \in J_{\text{Medium}}(p,q)$ is given by (2.16) and the maximal number of pieces that can be packed into an empty bin is given by $\left\lfloor \frac{H}{p} \right\rfloor \left\lfloor \frac{W}{q} \right\rfloor$. Thus, lower bound $L^2_{BM,4}$ is given by:

$$L^2_{BM,4} = \max_{1\le p\le \frac{W}{2}, 1\le q\le \frac{H}{2}} |J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)| + \max\left(0, \left\lceil \frac{\sum_{j\in J_s(p,q)} m'(j,p,q) - \sum_{j\in J_{\text{Medium}}(p,q)} m(j,p,q)}{\left\lfloor \frac{H}{p} \right\rfloor \left\lfloor \frac{W}{q} \right\rfloor} \right\rceil\right) \tag{2.19}$$

Boschetti and Mingozzi [6] show that $L^2_{BM,4}$ dominates $L^2_{MV,3}$ and its complexity is $O(n^3)$.

Boschetti and Mingozzi lower bound $\mathbf{L^2_{BM,3}}$: In $L^2_{MV,3}$ and $L^2_{BM,4}$ the items are considered as a number of pieces of equal size, a lower bound on the number of pieces of each item and an upper bound on the number of pieces that can be packed into an empty bin and an initialized bin are computed. However for $L^2_{BM,3}$ the items are not considered as a number of pieces. Indeed, an upper bound on the number of items that can be packed together into an empty bin and a bin that was initialized with an item from $J_{\text{Medium}}(p,q)$ are calculated by computing a maximal number of items that can be fitted side by side and a maximal number of items that can be fitted one over other.

Denote by $M_W(w, J_s)$ the maximal number of items from S that can be packed side by side in a bin of width w. Also, by symmetry, $M_H(h, J_s)$ is defined by similar way by considering the height dimension. Then an upper bound on the number of items from $J_s(p,q)$ that can be fitted in an empty bin is given by

$$M_W(W, J_s(p,q)) \times M_H(H, J_s(p,q)) \tag{2.20}$$

Also, a bound on the number of items that can be fitted in a bin that has been initialized with an item from $J_{\text{Medium}}(p,q)$ is given by

$$\begin{aligned} m''(j,p,q) = {} & M_W(W - w_j, J_s(p,q)) \times M_H(H, J_s(p,q)) \\ & + M_W(W, J_s(p,q)) \times M_H(H - h_j, J_s(p,q)) \\ & - M_W(W - w_j, J_s(p,q)) \times M_H(H - h_j, J_s(p,q)) \end{aligned} \tag{2.21}$$

Thus, lower bound $L^2_{BM,3}$ is given by:

$$L^2_{BM,3} = \max_{1\le p\le \frac{W}{2}, 1\le q\le \frac{H}{2}} |J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)| + \max\left(0, \left\lceil \frac{|J_s(p,q)| - \sum_{j\in J_{\text{Medium}}(p,q)} m''(j,p,q)}{M_W(W, J_s(p,q)) \times M_H(H, J_s(p,q))} \right\rceil\right) \tag{2.22}$$

where, the subset $J_s(p,q)$ is defined as in $L^2_{BM,4}$.

Boschetti and Mingozzi [6] show that the complexity of $L^2_{BM,3}$ is $O(n^4)$. Furthermore, they show that $L^2_{BM,3}$ dominates $L^2_{MV,3}$, but no dominance relationship exists between $L^2_{BM,3}$ and $L^2_{BM,4}$.

It is noteworthy that only values of p and q that correspond to distinct values of h_j or $H - h_j$ and w_j or $W - w_j$, respectively, have to be considered to derive $L^2_{MV,3}$, $L^2_{BM,3}$ and $L^2_{BM,4}$. For each combination of (p,q) an $O(n)$-time procedure is needed to compute $m'(j,p,q), \forall j \in J_s(p,q)$ and $m(j,p,q)$, for all $j \in J_{\text{Medium}}(p,q)$. However,

$O(n^2)$ time is necessary to compute $m''(j,p,q)$, for all $j \in J_{\text{Medium}}(p,q)$. Actually, for each $j \in J_{\text{Medium}}(p,q)$ a particular knapsack, where the profit of each item is equal to one, is invoked to calculate M_W and M_H. The complexity time of this particular knapsack problem is $O(n)$ time. Therefore, an overall complexity time of $O(n^3)$, $O(n^3)$ and $O(n^4)$ is needed to achieve $L^2_{MV,3}$, $L^2_{BM,3}$ and $L^2_{BM,4}$, respectively.

2.4. Lower bounds of Class 3

All these lower bounds are based on the so-called *Dual Feasible Functions* (DFF). Before proceeding further, and for the sake of making the paper self-contained, we shall briefly introduce this class of functions that represent a powerful tool for deriving enhanced $2BPP$ lower bounds.

Definition 2.2. A function $f : [0,1] \to [0,1]$ is said to be *dual feasible* if for any finite set S of positive real numbers, we have the relation

$$\sum_{x\in S} x \leq 1 \Longrightarrow \sum_{x\in S} f(x) \leq 1$$

Definition 2.3. A function $f : \mathbb{N} \to \mathbb{N}$ is said to be *discrete dual feasible* if for any set $S \subset \mathbb{N}$, we have the relation:

$$\sum_{x\in S} x \leq C \Rightarrow \sum_{x\in S} f(x) \leq f(C)$$

The concept of dual feasible functions has been first introduced by Johnson [28] in the context of bin packing. During the last years, and following the paper by Fekete and Schepers [19], there has been a resurgence of interest in DFF as a new tool for deriving a class of tight lower bounds for bin packing problems. Furthermore, Carlier and Néron [10] introduced the so-called *redundant functions* in the context of cumulative scheduling and which might be viewed as discretized versions of DFF. We refer to Clautiaux *et al.* [14] for a survey of DFF.

So far, several DFF were used in the context of bin packing problems. Here, we provide a short review of the most preeminent ones.

Type 1.

Fekete and Schepers [20] introduced the following DFF: Let $k \in \mathbb{N}$ then:

$$f^k_{FS,0}(x) = \begin{cases} x & \text{if} \quad (k+1)x \in \mathbb{Z} \\ \lfloor (k+1)x \rfloor \dfrac{1}{k} & \text{otherwise} \end{cases} \tag{2.23}$$

Type 2.

Fekete and Schepers [20] presented $f_{FS,1}$ that can be considered as the dual feasible version of the lower bound of Martello and Toth [33].

Let $k \in \left[0, \frac{1}{2}\right]$

$$f^k_{FS,1}(x) = \begin{cases} 1 & \text{if} \quad x > 1-k \\ x & \text{if} \quad k \leq x \leq 1-k \\ 0 & \text{otherwise} \end{cases} \tag{2.24}$$

Carlier *et al.* [9] proposed the following discretized version of this function.

Let $k \in \left[0, \frac{C}{2}\right]$ then:

$$f^k_{CCM,0}(x) : [0,C] \to [0,C]$$

$$f^k_{CCM,0}(x) = \begin{cases} C & \text{if} \quad x > C - k \\ x & \text{if} \quad k \le x \le C - k \\ 0 & \text{otherwise} \end{cases} \tag{2.25}$$

Type 3.

Carlier *et al.* [9] introduced a new concept of *Data Dependent Dual Feasible Function* (DDFF) that can be defined as follows.

Definition 2.4. Let $\Upsilon = \{\upsilon_1, \upsilon_2, \ldots, \upsilon_n\}$ be a set of n integers, and C be an integer such that $C \ge \upsilon_i$, $\forall i \in \{1, \ldots, n\}$. A function $f : [0, C] \to [0, f(C)]$ associated with Υ and C is said to be a *DDFF* if for any subset $S \subset \Upsilon$, we have the relation:

$$\sum_{x \in S} x \le C \Rightarrow \sum_{x \in S} f(x) \le f(C)$$

Carlier *et al.* [9] proposed a DDFF that is inspired from $L^2_{BM,3}$ of Boschetti and Mingozzi [6]. At this point, it should be pointed that no equivalence exists between resulting lower bound of this function and $L^2_{BM,3}$.

Let $k \in \left[1, \frac{C}{2}\right]$ then:

$$f^k_{CCM,1}(x) : [0, C] \to [0, M_C(C, S)]$$

$$f^k_{CCM,1}(x) = \begin{cases} M_C(C, S) - M_C(C - x, S) & \text{if} \quad x > \frac{C}{2} \\ 1 & \text{if} \quad k \le x \le \frac{C}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2.26}$$

where $M_c(C, S)$ denotes the maximum number of items in a subset S that can be simultaneously packed into a (one-dimensional) knapsack of capacity C.

Type 4.

Fekete and Schepers [20] presented the following DFF:

Let $k \in \left(0, \frac{1}{2}\right]$

$$f^k_{FS,2}(x) = \begin{cases} 1 - \dfrac{\lfloor (1 - x)k^{-1} \rfloor}{\lfloor k^{-1} \rfloor} & \text{if} \quad x > \dfrac{1}{2} \\ \dfrac{1}{\lfloor k^{-1} \rfloor} & \text{if} \quad k \le x \le \dfrac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2.27}$$

Clautiaux *et al.* [14] showed that the following DFF is the discrete version of $f_{FS,2}$.

Let $k \in \left[1, \frac{C}{2}\right]$ then:

$$f^k_{MV,1}(x) : [0, C] \to \left[0, \left\lfloor \frac{C}{k} \right\rfloor\right]$$

$$f^k_{MV,1}(x) = \begin{cases} \left(\left\lfloor \dfrac{C}{k} \right\rfloor - \left\lfloor \dfrac{C - x}{k} \right\rfloor\right) & \text{if} \quad x > \dfrac{C}{2} \\ 1 & \text{if} \quad k \le x \le \dfrac{C}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2.28}$$

Furthermore, applying $f_{MV,1}$ on both dimensions turns out to yield $L^2_{MV,3}$ lower bound.

As we mentioned earlier in Section 2.3.2 $L^2_{BM,4}$ can be considered as an improved version of $L^2_{MV,3}$. Consider the function $f_{BM,1}$ introduced in Boschetti [4]:
Let $k \in \left[1, \frac{C}{2}\right]$ then:

$$f^k_{BM,1}(x) : [0, C] \to \left[0, \left\lfloor \frac{C}{k} \right\rfloor\right]$$

$$f^k_{BM,1}(x) = \begin{cases} \left(\left\lfloor \frac{C}{k} \right\rfloor - \left\lfloor \frac{C-x}{k} \right\rfloor\right) & \text{if} \quad x > \frac{C}{2} \\ \left\lfloor \frac{x}{k} \right\rfloor & \text{if} \quad x \le \frac{C}{2} \end{cases} \tag{2.29}$$

Interestingly, $L^2_{BM,4}$ is equivalent to the lower bound resulting by applying $f_{BM,1}$ on both dimensions.
Finally, Carlier *et al.* [9] extended the DFF $f_{BM,1}$ to obtain the following function:
Let $k \in \left[1, \frac{C}{2}\right]$ then:

$$f^k_{CCM,2}(x) : [0, C] \to \left[0, 2\left\lfloor \frac{C}{k} \right\rfloor\right]$$

$$f^k_{CCM,2}(x) = \begin{cases} 2\left(\left\lfloor \frac{C}{k} \right\rfloor - \left\lfloor \frac{C-x}{k} \right\rfloor\right) & \text{if} \quad x > \frac{C}{2} \\ \left\lfloor \frac{C}{k} \right\rfloor & \text{if} \quad x = \frac{C}{2} \\ 2\left\lfloor \frac{x}{k} \right\rfloor & \textit{otherwise} \end{cases} \tag{2.30}$$

It is noteworthy that function $f_{CCM,2}$ is a maximal DFF (MDFF). The concept of MDFF was first introduced by Carlier and Néron (2007).

Definition 2.5. A function f is a MDFF if and only if: (i) f is a DFF, (ii) $f(0) = 0$ and, (iii) f is nondecreasing, superadditive ($f(x) + f(y) \ge f(x+y)$), and symmetric ($f(x) + f(C-x) = f(C), \forall x \in [0, C]$).

Interestingly, $f_{CCM,2}$ can be obtained by applying Theorem 2.6 of Clautiaux *et al.* [14] on $f_{BM,1}$.

Theorem 2.6. *Let f be a superadditive and nondecreasing function defined from $[0, C]$ to $[0, f(C)]$, and such that $f(0) = 0$. The following function is a maximal DFF.*

$$g \; : \; [0, C] \to [0, 2f(C)] \tag{2.31}$$

$$g(x) = \begin{cases} 2f(C) - 2f(C-x), & \textit{if} \quad C \ge x > \frac{C}{2} \\ f(C), & \textit{if} \quad x = \frac{C}{2} \\ 2f(x), & \textit{if} \quad x < \frac{C}{2} \end{cases} \tag{2.32}$$

2.4.1. Fekete and Schepers lower bound

In order to derive a valid lower bound for $2BPP$, Fekete and Schepers (2.23), (2.24) and (2.27) and normalize the dimensions of the items as $w^{'}_j = \frac{w_j}{W}$ and $h^{'}_j = \frac{h_j}{H}$ and set the bin sizes to $W^{'} = 1$ and $H^{'} = 1$.

Let functions F_u $(u = 1 \dots 7)$ be defined as follows:

$$F_1 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^1_{FS,0}(w^{'}_j) f^{\epsilon}_{FS,1}(h^{'}_j) \tag{2.33}$$

$$F_2 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^{\epsilon}_{FS,1}(w^{'}_j) f^{1}_{FS,0}(h^{'}_j) \tag{2.34}$$

$$F_3 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^1_{FS,0}(w^{'}_j) f^{\epsilon}_{FS,2}(h^{'}_j) \tag{2.35}$$

$$F_4 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^{\epsilon}_{FS,2}(w^{'}_j) f^{1}_{FS,0}(h^{'}_j) \tag{2.36}$$

$$F_5 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} w^{'}_j f^{\epsilon}_{FS,1}(h^{'}_j) \tag{2.37}$$

$$F_6 = \max_{\epsilon \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^{\epsilon}_{FS,1}(w^{'}_j) h^{'}_j \tag{2.38}$$

$$F_7 = \max_{\epsilon, \epsilon' \in \left(0, \frac{1}{2}\right]} \sum_{j \in J} f^{\epsilon}_{FS,2}(w^{'}_j) f^{\epsilon^{'}}_{FS,2}(h^{'}_j) \tag{2.39}$$

Lower bound L^2_{FS} is given by

$$L^2_{FS} = \max_{1 \le u \le 7} F_u \tag{2.40}$$

Note that the time complexity of L^2_{FS} is $O(n^2)$.

2.4.2. Carlier et al. lower bounds $L^2_{CCM,1}$ and $L^2_{CCM,2}$

We start first with $L^2_{CCM,1}$. Carlier *et al.* [9] proposed a lower bound, hereafter referred to by $L^2_{CCM,1}$, which can be viewed as a modified version of $L^2_{BM,2}$. Indeed, the lower bound L^1_{MV} is replaced by L^1_{CCM} which is defined as follows:

$$L^1_{CCM} = \max_{u \in \{0,1,2\}} \max_{k \in \left[1, \frac{C}{2}\right]} \sum_{j \in S} f^k_{CCM,u}(c_j)$$

Since L^1_{CCM} dominates L^1_{MV} (see Carlier *et al.* [9]) then $L^2_{CCM,1}$ dominates $L^2_{BM,2}$. Note that $L^2_{CCM,1}$ has the same complexity as $L^2_{BM,2}$. It is noteworthy that since $L^2_{CCM,1}$ is defined in the spirit of $L^2_{BM,2}$ then is similarly classified in Class 2.

Carlier *et al.* [9] introduced lower bound $L^2_{CCM,2}$ based on the three discrete dual feasible functions (2.25), (2.26) and (2.30) this lower bound can be defined as follows:

$$F(u,v) = max_{k \in \left[0, \frac{W}{2}\right], l \in \left[0, \frac{H}{2}\right]} \left\lceil \frac{\sum_{j \in J} f^k_{CCM,u}(w_j) \times f^l_{CCM,v}(h_j)}{f^k_{CCM,u}(W) \times f^l_{CCM,v}(H)} \right\rceil \tag{2.41}$$

$$L^2_{CCM,2} = \max_{u \in \{0,1,2\}, v \in \{0,1,2\}} F(u,v) \tag{2.42}$$

Carlier *et al.* [9] claim that $L^2_{CCM,2}$ dominates both $L^2_{BM,3}$ and $L^2_{BM,4}$. However, in Section 3.1 we show that this result is not correct.

3. Dominance results

In this section, we analyze dominance relationships between $2BPP$ lower bounds. In the sequel, we shall denote by:

- $L^2_{MV} = \max(L^2_{MV,2}, L^2_{MV,3})$
- $L^2_{BM} = \max(L^2_{BM,2}, L^2_{BM,3}, L^2_{BM,4})$
- $L^2_{CCM} = \max(L^2_{CCM,1}, L^2_{CCM,2})$

It is noteworthy that $L^1_{MV,1}$ and $L^1_{BM,1}$ will not be further considered since they are dominated by $L^2_{MV,1}$ and $L^2_{BM,1}$, respectively.

3.1. About the dominance of L^2_{CCM}

Carlier *et al.* [9] claim that L^2_{CCM} dominates L^2_{BM}. Toward this end, they show that $L^2_{CCM,1}$ dominates $L^2_{BM,2}$ and that $L^2_{CCM,2}$ dominates both $L^2_{BM,3}$ and $L^2_{BM,4}$. However, no relationship exists between $L^2_{CCM,2}$ and $L^2_{BM,3}$. In fact, Carlier *et al.* [9] claim that $F(1,1)$ dominates $L^2_{BM,3}$. Let us denote by $F^{p,q}(1,1)$ the lower bound resulting by applying the DDFF $f^p_{CCM,1}$ and $f^q_{CCM,1}$ on the width and the height, respectively. Clearly, we have:

$$F^{p,q}(1,1) = \left\lceil \frac{\sum_{j\in J} f^p_{CCM,1}(h_j) \times f^q_{CCM,1}(w_j)}{f^p_{CCM,1}(H) \times f^q_{CCM,1}(W)} \right\rceil \tag{3.1}$$

$$= \left\lceil \frac{\sum_{j\in J} f^p_{CCM,1}(h_j) \times f^q_{CCM,1}(w_j)}{M_H(H, J_H(p)) \times M_W(W, J_W(q))} \right\rceil \tag{3.2}$$

Where $J_H(p) = \{j \in J : p \le h_j \le \frac{H}{2}\}$ and $J_W(q) = \{j \in J : q \le w_j \le \frac{W}{2}\}$. Carlier *et al.* [9] stated that:

$$F(1,1) = \max_{p,q} \left\lceil \frac{\sum_{j\in J} f^p_{CCM,1}(h_j) \times f^q_{CCM,1}(w_j)}{M_W(W, J_s(p,q)) \times M_H(H, J_s(p,q))} \right\rceil$$

However no relationship exists between subsets $J_s(p,q)$, $J_H(p)$ and $J_W(q)$. Indeed an item j having $p \le h_j \le \frac{H}{2}$ and $w_j < q$ belongs to $J_H(p)$, but not to $J_s(p,q)$. Similarly, let us consider an item j' having $h_{j'} > \frac{H}{2}$ and $w \le \frac{W}{2}$, this item belongs to $J_s(p,q)$ but not to $J_H(p)$. A similar observation can be made for $J_s(p,q)$ and $J_W(q)$ by interchanging the roles of the width the height. In their paper, Carlier *et al.* [9] wrongly used subset J_s instead of subsets J_W and J_H.

A Counterexample. We consider the following 6-item instance. The items' sizes are $(230,190)$, $(82,194)$, $(118,248)$, $(439,275)$, $(178,246)$, and $(283,25)$. The bin size is 920×430.
The corresponding values of the lower bounds are:

$$L^2_{CCM,1} = 1, L^2_{CCM,2} = 1 \text{ thus } L^2_{CCM} = 1$$
$$L^2_{BM,2} = 1, L^2_{BM,3} = 2, L^2_{BM,4} = 1 \text{ so } L^2_{BM} = 2$$

The value $L^2_{BM,3} = 2$ can be obtained by setting $p = 118$ and $q = 190$. Hence, as we see L^2_{CCM} does not dominate L^2_{BM}.

In the following, we give some details about the computation of $L^2_{BM,3}$ and $L^2_{CCM,2}$ for the latter instance. In particular, we restrict the details for $p = 118$ and $q = 190$.

We begin first with the computation of $L^2_{BM,3}$. Note that, in this instance $|J_{\text{Large}}(p,q) \cup J_{\text{Medium}}(p,q)| = 0$, thus

$$L^2_{BM,3}(p,q) = \max\left(0, \left\lceil \frac{|J_s(p,q)|}{M_W(W, J_s(p,q)) \times M_H(H, J_s(p,q))} \right\rceil\right)$$

TABLE 1. Details of computing DFF.

w_j	230	82	118	439	178	283	920
$f^{118}_{CCM,0}$	230	0	118	439	178	283	920
$f^{118}_{CCM,1}$	1	0	1	1	1	1	4
$f^{118}_{CCM,2}$	2	0	2	6	2	4	14
h_j	190	194	248	275	246	25	430
$f^{190}_{CCM,0}$	190	194	430	430	430	0	430
$f^{190}_{CCM,1}$	1	1	2	2	2	0	2
$f^{190}_{CCM,2}$	2	2	4	4	4	0	4

Furthermore, $J_s(p,q)=\{1,3,4,5\}, M_W(W,J_s(p,q))=3$ and $M_H(H,J_s(p,q))=1$. Therefore, $L^2_{BM,3}(p,q)=2$.

Regarding $L^2_{CCM,2}$, Table 1 provides the details of the computations of the different dual feasible functions on both dimensions. Therefore, we get $L^2_{CCM,2}=1$.

At this point, it is noteworthy that L^2_{CCM} dominates L^2_{MV} and L^2_{FS}. Indeed, since $\max(L^2_{BM,2}, L^2_{BM,4})$ dominates both $L^2_{MV}$and L^2_{FS} (see Boschetti and Mingozzi [6]) and $L^2_{CCM,1} \geq L^2_{BM,4}$ and $L^2_{CCM,2} \geq L^2_{BM,2}$ (see Carlier *et al.* [9]) then L^2_{CCM} dominates both L^2_{MV} and L^2_{FS} .

3.2. Additional new dominance results

In order to get a better picture of the dominance relationships between the different lower bounds, we provide hereafter some dominance relationships that were omitted in the literature so far.

- Martello and Vigo [34] claimed that $L^2_{MV,2}$ and $L^2_{MV,3}$ are better than L^2_0 and $L^2_{MV,1}$. However no dominance relation exists between $L^2_{MV,3}$ and L^2_0.

  We consider the following 5-item instance where the item sizes are (7,7), (3,3), (3,6), (3,6), (6,3), respectively. The bin size is 10×10. For this instance, we have $L^2_0 = 2$ and $L^2_{MV,3} = 1$. However, if we consider the following 5-item instance (Martello and Vigo [34]) where the item sizes are $(8,16), (3,3), (3,3), (3,3)$ and $(3,3)$, respectively, and the bin size is 10×20. For this latter instance, we have $L^2_0 = 1$ and $L^2_{MV,3} = 2$.
- $L^2_{BM,4}$ dominates L^2_0. Let $p=1$ and $q=1$ then

$$L^2_{BM,4} = \left\lceil \frac{\sum_{j \in J_{\text{Large}}(p,q)} WH + \sum_{j \in (J_s(p,q) \cup J_{\text{Medium}}(p,q))} w_j h_j}{HW} \right\rceil \geq L_0$$

- No dominance relationship exists between $L^2_{BM,3}$ and L_0.

  We consider the following 11-item instance where the item sizes are (6,6), (4,5), (5,4), (1,4), (1,4), (1,4), (1,4), (4,1), (4,1), (4,1) and (4,1), respectively. The bin size is 10×10. We have $L_0 = 2$ and $L^2_{BM,3} = 1$.
  However, if we consider the following 5-item instance where the item sizes are $(6,6)$, $(4,4)$, $(4,4)$, $(4,4)$, and $(4,4)$, respectively. The bin size is 10×10. We have $L_0 = 1$ and $L^2_{BM,3} = 2$.

3.3. Synthesis of dominance results

This section contains a brief overview of the dominance results already established between the polynomial lower bounds presented in this paper. In Figure 1 each lower bound is represented by a node. An outgoing arc from node A to node B means that lower bound A is dominated by lower bound B. The dominance was established in the reference located at the side or below the arc.

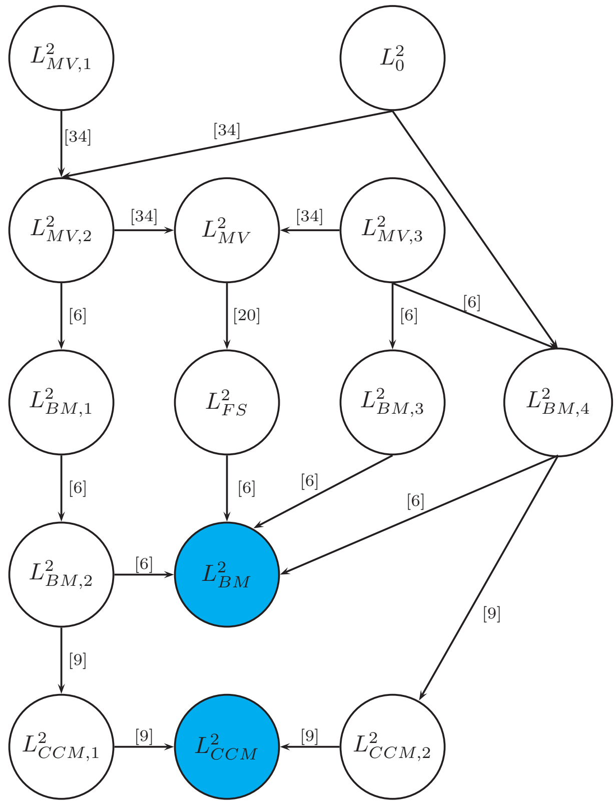

FIGURE 1. Synthesis of the dominance relationships.

4. COMPUTATIONAL RESULTS

In this section we compare the empirical performance of the previous lower bounds. We have coded them in C, and implemented on an Intel CORE Duo 2 2.4 GHz personal computer with 3GB RAM.

We tested the lower bounds on the benchmark available on the following web page:
http://www.or.deis.unibo.it/research_pages/ORinstances/ORinstances.htm
The benchmark is composed of 10 classes, the first 6 classes have been introduced by Berkey and Yang [2]. Martello and Vigo [34] proposed the remaining ones. For each class, the number of items is 20,40,60,80 and 100, respectively. For each combination of class and number of item there are 10 randomly generated instances.

A summary of the performance of ten lower bounds is depicted in Tables 2−5. It is noteworthy that the performance of $L^2_{MV,1}$ was omitted, since this latter is dominated by $L^2_{MV,2}$ and has the same complexity. In Tables 2−5 and 7−10 we report for each lower bound:

- Opt: number of times that the lower bound is equal to a proven optimal solution.
- Max: number of times that the lower bound yields the maximal value.

It should be precised that we did not report the CPU times because all these bounds are extremely fast and require only few milliseconds.

From Tables 2−5, we observe that the performance of most lower bounds (measured in terms of the number of times it yields an optimal value) is dependent on the instance density (that is, the ratio of the number of items

TABLE 2. Class 1 lower bounds results Classes 1−10.

		L_0^2		$L_{MV,2}^2$		$L_{BM,1}^2$				L_0^2		$L_{MV,2}^2$		$L_{BM,1}^2$	
Class	n	Opt	Max	Opt	Max	Opt	Max	Class	n	Opt	Max	Opt	Max	Opt	Max
1	20	4	4	6	7	7	8	6	20	10	10	10	10	10	10
	40	3	5	5	7	5	8		40	8	10	8	10	8	10
	60	2	3	3	5	5	7		60	10	10	10	10	10	10
	80	0	0	5	5	6	6		80	10	10	10	10	10	10
	100	3	3	7	7	9	9		100	8	10	8	10	8	10
	Tot	**12**	**15**	**26**	**31**	**32**	**38**		**Tot**	**46**	**50**	**46**	**50**	**46**	**50**
2	20	10	10	10	10	10	10	7	20	3	3	8	8	8	8
	40	10	10	10	10	10	10		40	0	1	7	8	7	8
	60	10	10	10	10	10	10		60	0	0	7	8	7	8
	80	10	10	10	10	10	10		80	0	0	1	6	1	6
	100	10	10	10	10	10	10		100	0	0	7	7	7	7
	Tot	**50**	**50**	**50**	**50**	**50**	**50**		**Tot**	**3**	**4**	**30**	**37**	**30**	**37**
3	20	5	6	5	7	5	7	8	20	3	3	7	7	7	7
	40	3	5	4	6	4	6		40	0	0	8	9	8	9
	60	1	3	4	6	4	6		60	0	0	8	9	8	9
	80	0	1	5	6	6	7		80	0	0	8	9	8	9
	100	1	2	4	6	4	6		100	0	0	4	9	4	9
	Tot	**10**	**17**	**22**	**31**	**23**	**32**		**Tot**	**3**	**3**	**35**	**43**	**35**	**43**
4	20	10	10	10	10	10	10	9	20	0	0	10	10	10	10
	40	10	10	10	10	10	10		40	0	0	6	6	6	6
	60	8	10	8	10	8	10		60	0	0	6	6	6	6
	80	8	10	8	10	8	10		80	0	0	2	2	3	3
	100	9	10	9	10	9	10		100	0	0	5	5	5	5
	Tot	**45**	**50**	**45**	**50**	**45**	**50**		**Tot**	**0**	**0**	**29**	**29**	**30**	**30**
5	20	2	2	5	5	5	5	10	20	6	7	8	9	8	9
	40	1	2	6	8	7	9		40	5	7	7	9	7	9
	60	0	0	3	4	4	5		60	3	6	6	9	6	9
	80	0	0	2	5	2	8		80	4	9	5	10	5	10
	100	0	1	2	4	2	4		100	4	10	4	10	4	10
	Tot	**3**	**5**	**18**	**26**	**20**	**31**		**Tot**	**22**	**39**	**30**	**47**	**30**	**47**

to the number of required bins). In Table 6, we report the mean number of items per bin which is computed for each instance as the total number of items to the optimal number of bins (or, near-optimal if the optimal value is unknown).

Looking at these tables, we can make the following observations:

- Lower bounds $L_{BM,2}^2$, $L_{CCM,1}^2$ and $L_{CCM,2}^2$ outperform all the other bounds. In particular $L_{CCM,2}^2$ performs very well and yields the optimal solution for 426 instances, and the best solution for 495 instances.
- Surprisingly, $L_{MV,3}^2$ and $L_{BM,3}^2$ exhibit a relative poor performance on most classes. However $L_{BM,3}^2$ outperforms L_0^2, $L_{MV,2}^2$ and $L_{BM,1}^2$ on Class 9 where the density is the smallest.
- The relative performance of L_0^2, $L_{MV,2}^2$ and $L_{BM,1}^2$ decreases as the instance density decreases. For example, L_0^2 yields the best solution for 150 instances in Classes $2, 4$ and 6, 22 instances in Class 10 and, 0 in Class 9.

TABLE 3. Class 2 lower bounds results Classes 1−5.

		$L^2_{BM,2}$		$L^2_{CCM,1}$		$L^2_{MV,3}$		$L^2_{BM,3}$		$L^2_{BM,4}$	
Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max
1	20	9	10	9	10	0	0	4	4	7	8
	40	7	10	7	10	0	0	1	1	6	9
	60	7	9	7	9	0	0	0	0	7	9
	80	8	8	9	9	0	0	0	0	8	8
	100	10	10	10	10	0	0	0	0	7	7
	Tot	**41**	**47**	**42**	**48**	**0**	**0**	**5**	**5**	**35**	**41**
2	20	10	10	10	10	10	10	10	10	10	10
	40	10	10	10	10	1	1	1	1	10	10
	60	10	10	10	10	0	0	0	0	10	10
	80	10	10	10	10	0	0	0	0	10	10
	100	10	10	10	10	0	0	0	0	10	10
	Tot	**50**	**50**	**50**	**50**	**11**	**11**	**11**	**11**	**50**	**50**
3	20	8	10	8	10	1	1	4	6	6	8
	40	8	10	8	10	0	0	3	3	8	10
	60	6	8	6	8	0	0	2	3	6	8
	80	8	9	8	9	0	0	1	1	8	9
	100	7	9	7	9	0	0	0	1	7	9
	Tot	**37**	**46**	**37**	**46**	**1**	**1**	**10**	**14**	**35**	**44**
4	20	10	10	10	10	10	10	10	10	10	10
	40	10	10	10	10	1	1	1	1	10	10
	60	8	10	8	10	0	0	0	1	8	10
	80	8	10	8	10	0	0	0	0	8	10
	100	9	10	9	10	0	0	0	0	9	10
	Tot	**45**	**50**	**45**	**50**	**11**	**11**	**11**	**12**	**45**	**50**
5	20	10	10	10	10	0	0	4	4	5	5
	40	7	10	7	10	1	1	5	6	7	10
	60	7	9	7	9	0	0	2	3	7	9
	80	3	9	4	10	0	0	3	5	4	10
	100	4	6	4	6	0	0	2	2	8	10
	Tot	**31**	**44**	**32**	**45**	**1**	**1**	**16**	**20**	**31**	**44**

Actually, the density is about 25 for Classes 2, 4 and 6, 6 for Class 10, and 1.4 for Class 9. However, we observe that the performance of $L^2_{BM,2}$, $L^2_{CCM,1}$ and L^2_{FS} is much less dependent on the instance density. Interestingly, this dependence seems drastically reduced for $L^2_{BM,4}$ and $L^2_{CCM,2}$.

TABLE 4. Class 2 lower bounds results Classes 6−10

		$L^2_{BM,2}$		$L^2_{CCM,1}$		$L^2_{MV,3}$		$L^2_{BM,3}$		$L^2_{BM,4}$	
Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max
6	20	10	10	10	10	10	10	10	10	10	10
	40	8	10	8	10	3	5	3	5	8	10
	60	10	10	10	10	0	0	1	1	10	10
	80	10	10	10	10	0	0	0	0	10	10
	100	8	10	8	10	0	0	0	0	8	10
	Tot	**46**	**50**	**46**	**50**	**13**	**15**	**14**	**16**	**46**	**50**
7	20	8	8	8	8	0	0	3	3	8	8
	40	7	8	7	8	0	0	0	0	8	9
	60	7	8	7	8	0	0	0	0	8	9
	80	2	7	2	7	0	0	0	0	2	7
	100	7	7	7	7	0	0	0	0	8	8
	Tot	**31**	**38**	**31**	**38**	**0**	**0**	**3**	**3**	**34**	**41**
8	20	10	10	10	10	0	0	1	1	7	7
	40	8	9	8	9	0	0	0	1	9	10
	60	8	9	8	9	0	0	0	0	8	9
	80	8	9	8	9	0	0	0	0	9	10
	100	4	9	4	9	0	0	0	0	5	10
	Tot	**38**	**46**	**38**	**46**	**0**	**0**	**1**	**2**	**38**	**46**
9	20	10	10	10	10	5	5	10	10	10	10
	40	10	10	10	10	1	1	8	8	7	7
	60	10	10	10	10	2	2	8	8	8	8
	80	10	10	10	10	1	1	5	5	7	7
	100	10	10	10	10	0	0	8	8	8	8
	Tot	**50**	**50**	**50**	**50**	**9**	**9**	**39**	**39**	**40**	**40**
10	20	8	9	8	9	3	3	7	8	7	8
	40	8	10	8	10	0	0	2	4	8	10
	60	6	9	6	9	0	0	0	2	4	7
	80	5	10	5	10	0	0	0	0	4	9
	100	4	10	4	10	0	0	0	0	4	10
	Tot	**31**	**48**	**31**	**48**	**3**	**3**	**9**	**14**	**27**	**44**

- For Classes 2, 4 and 6, we see that all the considered lower bounds, except $L^2_{MV,3}$ and $L^2_{BM,3}$, yield the best solution and achieve the optimal value for 141 instances. Hence, these classes appear to be the easiest ones. They are characterized by a large number of items per bin.

Furthermore, we report in Table 7 the performance of the following four lower bounds $L^2_{MV}, L^2_{FS}, L^2_{BM}$ and L^2_{CCM}.

TABLE 5. DFF based lower bounds results Classes 1−10.

		L^2_{FS}		$L^2_{CCM,2}$				L^2_{FS}		$L^2_{CCM,2}$	
Class	n	Opt	Max	Opt	Max	Class	n	Opt	Max	Opt	Max
1	20	8	9	9	10	6	20	10	10	10	10
	40	6	9	7	10		40	8	10	8	10
	60	7	9	8	10		60	10	10	10	10
	80	8	8	10	10		80	10	10	10	10
	100	10	10	10	10		100	8	10	8	10
	Tot	**39**	**45**	**44**	**50**		**Tot**	**46**	**50**	**46**	**50**
2	20	10	10	10	10	7	20	8	8	10	10
	40	10	10	10	10		40	7	8	9	10
	60	10	10	10	10		60	7	8	9	10
	80	10	10	10	10		80	1	6	5	10
	100	10	10	10	10		100	7	7	10	10
	Tot	**50**	**50**	**50**	**50**		**Tot**	**30**	**37**	**43**	**50**
3	20	6	8	8	10	8	20	7	7	9	9
	40	8	10	8	10		40	8	9	9	10
	60	5	7	7	9		60	8	9	9	10
	80	8	9	9	10		80	8	9	9	10
	100	6	8	7	9		100	4	9	5	10
	Tot	**33**	**42**	**39**	**48**		**Tot**	**35**	**43**	**41**	**49**
4	20	10	10	10	10	9	20	10	10	10	10
	40	10	10	10	10		40	7	7	10	10
	60	8	10	8	10		60	8	8	10	10
	80	8	10	8	10		80	7	7	10	10
	100	9	10	9	10		100	8	8	9	9
	Tot	**45**	**50**	**45**	**50**		**Tot**	**40**	**40**	**49**	**49**
5	20	5	5	9	9	10	20	8	9	9	10
	40	7	10	7	10		40	8	10	8	10
	60	7	9	8	10		60	6	9	7	10
	80	4	9	4	10		80	5	10	5	10
	100	5	7	8	10		100	4	10	4	10
	Tot	**28**	**40**	**36**	**49**		**Tot**	**31**	**48**	**33**	**50**

TABLE 6. Average number of items per bin.

Class	1	2	3	4	5	6	7	8	9	10
items/bin	3.02	24.19	4.38	25.21	3.42	27.77	3.66	3.63	1.4	6.16

We see from this table that L^2_{CCM} provides the best bound over *all* instances. On the other hand, we observe that L^2_{BM} exhibits a remarkable performance on all problem classes, except on Class 7, and is only marginally dominated by L^2_{CCM}.

In order to get a better picture of the performance of L^2_{CCM}, we report in Tables 8 and 9 the performance of $F(u, v)$ (for $u, v = 0, 1$ and 2).

TABLE 7. Overall lower bounds results Classes 1−10.

		L^2_{MV}		L^2_{FS}		L^2_{BM}		L^2_{CCM}				L^2_{MV}		L^2_{FS}		L^2_{BM}		L^2_{CCM}	
Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max
1	20	6	7	8	9	9	10	9	10	6	20	10	10	10	10	10	10	10	10
	40	5	7	6	9	7	10	7	10		40	8	10	8	10	8	10	8	10
	60	3	5	7	9	7	9	8	10		60	10	10	10	10	10	10	10	10
	80	5	5	8	8	9	9	10	10		80	10	10	10	10	10	10	10	10
	100	7	7	10	10	10	10	10	10		100	8	10	8	10	8	10	8	10
	Tot	**26**	**31**	**39**	**45**	**42**	**48**	**44**	**50**		**Tot**	**46**	**50**	**46**	**50**	**46**	**50**	**46**	**50**
2	20	10	10	10	10	10	10	10	10	7	20	8	8	8	8	8	8	10	10
	40	10	10	10	10	10	10	10	10		40	7	8	7	8	8	9	9	10
	60	10	10	10	10	10	10	10	10		60	7	8	7	8	8	9	9	10
	80	10	10	10	10	10	10	10	10		80	1	6	1	6	2	7	5	10
	100	10	10	10	10	10	10	10	10		100	7	7	7	7	8	8	10	10
	Tot	**50**	**50**	**50**	**50**	**50**	**50**	**50**	**50**		**Tot**	**30**	**37**	**30**	**37**	**34**	**41**	**43**	**50**
3	20	5	7	6	8	8	10	8	10	8	20	7	7	7	7	10	10	10	10
	40	4	6	8	10	8	10	8	10		40	8	9	8	9	9	10	9	10
	60	4	6	5	7	7	9	8	10		60	8	9	8	9	8	9	9	10
	80	5	6	8	9	8	9	9	10		80	8	9	8	9	9	10	9	10
	100	4	6	6	8	8	10	8	10		100	4	9	4	9	5	10	5	10
	Tot	**22**	**31**	**33**	**42**	**39**	**48**	**41**	**50**		**Tot**	**35**	**43**	**35**	**43**	**41**	**49**	**42**	**50**
4	20	10	10	10	10	10	10	10	10	9	20	10	10	10	10	10	10	10	10
	40	10	10	10	10	10	10	10	10		40	6	6	7	7	10	10	10	10
	60	8	10	8	10	8	10	8	10		60	6	6	8	8	10	10	10	10
	80	8	10	8	10	8	10	8	10		80	2	2	7	7	10	10	10	10
	100	9	10	9	10	9	10	9	10		100	5	5	8	8	10	10	10	10
	Tot	**45**	**50**	**45**	**50**	**45**	**50**	**45**	**50**		**Tot**	**29**	**29**	**40**	**40**	**50**	**50**	**50**	**50**
5	20	5	5	5	5	10	10	10	10	10	20	8	9	8	9	9	10	9	10
	40	6	8	7	10	7	10	7	10		40	7	9	8	10	8	10	8	10
	60	3	4	7	9	8	10	8	10		60	6	9	6	9	6	9	7	10
	80	2	5	4	9	4	10	4	10		80	5	10	5	10	5	10	5	10
	100	2	4	5	7	8	10	8	10		100	4	10	4	10	4	10	4	10
	Tot	**18**	**26**	**28**	**40**	**37**	**50**	**37**	**50**		**Tot**	**30**	**47**	**31**	**48**	**32**	**49**	**33**	**50**

From Tables 8 and 9, we see that $F(2,2)$ exhibits the best performance since it yields the optimal solution for 391 instances and the maximal solution for 460 instances. Interestingly, in Class 7, the performance of L^2_{CCM} is due to some different combination (namely $F(0,1)$ and $F(2,1)$). Indeed, for this class of instances, these latter bounds yields the maximal value for 47 instances while $F(2,2)$ is maximal for 41 instances only.

Before closing this section, we perform a comparison between $L^2_{CCM,2}$ and the two lower bounds of Caprara and Monaci [8], hereafter said $L^2_{CM,1}$ and $L^2_{CM,2}$. These two lower bounds are based on solving bilinear programs and have been implemented on a PC with AMD Athlon 4200+. The comparison is performed on the same set of 83 instances that were considered in the paper of Caprara and Monaci [8]. These instances belong to Classes 1, 3, 4, 5, 6, 7, 8 and 10 described above.

TABLE 8. CCM DFF results Classes 1−5.

		$F(0,0)$		$F(0,1)$		$F(0,2)$		$F(1,0)$		$F(1,1)$		$F(1,2)$		$F(2,0)$		$F(2,1)$		$F(2,2)$	
Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max
1	20	8	9	7	8	8	9	8	9	7	8	8	9	8	9	7	8	8	9
	40	6	9	5	8	6	9	5	8	5	8	5	8	6	9	5	8	6	9
	60	7	9	5	7	8	10	5	7	4	6	6	8	7	9	5	7	8	10
	80	8	8	7	7	7	7	7	7	7	7	7	7	10	10	9	9	10	10
	100	10	10	5	5	10	10	5	5	3	3	5	5	9	9	5	5	10	10
	Tot	**39**	**45**	**29**	**35**	**39**	**45**	**30**	**36**	**26**	**32**	**31**	**37**	**40**	**46**	**31**	**37**	**42**	**48**
2	20	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	40	10	10	4	4	10	10	4	4	1	1	4	4	10	10	4	4	10	10
	60	10	10	5	5	10	10	5	5	0	0	5	5	10	10	5	5	10	10
	80	10	10	0	0	10	10	0	0	0	0	0	0	10	10	0	0	10	10
	100	10	10	0	0	10	10	0	0	0	0	0	0	10	10	0	0	10	10
	Tot	**50**	**50**	**19**	**19**	**50**	**50**	**19**	**19**	**11**	**11**	**19**	**19**	**50**	**50**	**19**	**19**	**50**	**50**
3	20	6	8	6	8	6	8	8	10	7	9	8	10	6	8	6	8	6	8
	40	8	10	8	10	8	10	8	10	7	8	8	10	8	10	8	10	8	10
	60	5	7	5	7	5	7	6	8	4	6	6	8	6	8	5	7	7	9
	80	7	8	8	9	8	9	6	7	6	7	7	8	8	9	8	9	9	10
	100	6	8	5	7	6	8	3	5	3	5	3	5	7	9	5	7	7	9
	Tot	**32**	**41**	**32**	**41**	**33**	**42**	**31**	**40**	**27**	**35**	**32**	**41**	**35**	**44**	**32**	**41**	**37**	**46**
4	20	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	40	10	10	4	4	10	10	4	4	1	1	4	4	10	10	4	4	10	10
	60	8	10	5	7	8	10	5	7	0	1	5	7	8	10	5	7	8	10
	80	8	10	0	2	8	10	0	0	0	0	0	0	8	10	0	2	8	10
	100	9	10	0	0	9	10	0	0	0	0	0	0	9	10	0	0	9	10
	Tot	**45**	**50**	**19**	**23**	**45**	**50**	**19**	**21**	**11**	**12**	**19**	**21**	**45**	**50**	**19**	**23**	**45**	**50**
5	20	5	5	6	6	5	5	7	7	7	7	7	7	5	5	7	7	5	5
	40	7	10	6	9	7	10	7	10	4	7	6	9	7	10	6	9	7	10
	60	5	7	7	9	6	8	7	9	7	9	7	9	7	9	7	9	7	9
	80	3	9	3	9	3	9	4	9	4	8	4	9	4	10	4	9	4	10
	100	3	5	5	7	6	8	4	5	5	5	6	7	6	8	7	9	8	10
	Tot	**23**	**36**	**27**	**40**	**27**	**40**	**29**	**40**	**27**	**36**	**30**	**41**	**29**	**42**	**31**	**43**	**31**	**44**

We report on Table 10 the results of the three latter lower bounds. In addition to Opt and Max, we report *Nins* the number of considered instances on each class and *CPU* the computing time in seconds of each considered lower bound.

From Table 10, we observe that $L^2_{CM,2}$ outperforms $L^2_{CM,1}$ and $L^2_{CCM,2}$. It yields the best performance on all considered instances and achieves the optimal solution on 29 instances. $L^2_{CM,1}$ slightly outperforms $L^2_{CCM,2}$ these two lower bounds yield the maximal solution for 71 and 66 instances, respectively. Not surprisingly, $L^2_{CM,1}$ and $L^2_{CM,2}$ are extremely time consuming. Actually, the average time of $L^2_{CM,1}$ and $L^2_{CM,2}$ are 75 and 872 times longer than $L^2_{CCM,2}$, respectively.

TABLE 9. CCM DFF results Classes 6−10.

		$F(0,0)$		$F(0,1)$		$F(0,2)$		$F(1,0)$		$F(1,1)$		$F(1,2)$		$F(2,0)$		$F(2,1)$		$F(2,2)$	
Class	n	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max	Opt	Max
6	20	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	40	8	10	4	6	8	10	4	6	3	5	4	6	8	10	4	6	8	10
	60	10	10	6	6	10	10	7	7	0	0	7	7	10	10	6	6	10	10
	80	10	10	0	0	10	10	0	0	0	0	0	0	10	10	0	0	10	10
	100	8	10	0	0	8	10	0	0	0	0	0	0	8	10	0	0	8	10
	Tot	**46**	**50**	**20**	**22**	**46**	**50**	**21**	**23**	**13**	**15**	**21**	**23**	**46**	**50**	**20**	**22**	**46**	**50**
7	20	8	8	8	8	8	8	4	4	4	4	4	4	8	8	8	8	8	8
	40	7	8	8	9	8	9	1	1	1	1	1	1	7	8	8	9	8	9
	60	7	8	9	10	8	9	0	0	0	0	0	0	7	8	9	10	8	9
	80	1	6	5	10	2	7	0	0	0	0	0	0	1	6	5	10	2	7
	100	7	7	10	10	8	8	0	0	0	0	0	0	7	7	10	10	8	8
	Tot	**30**	**37**	**40**	**47**	**34**	**41**	**5**	**5**	**5**	**5**	**5**	**5**	**30**	**37**	**40**	**47**	**34**	**41**
8	20	7	7	0	0	7	7	8	8	2	2	8	8	7	7	1	1	7	7
	40	8	9	0	1	8	9	8	9	0	1	8	9	9	10	0	1	9	10
	60	8	9	0	0	8	9	9	10	0	0	9	10	8	9	0	0	8	9
	80	8	9	0	0	8	9	9	10	0	0	9	10	9	10	0	0	9	10
	100	4	9	0	0	4	9	5	10	0	0	5	10	5	10	0	0	5	10
	Tot	**35**	**43**	**0**	**1**	**35**	**43**	**39**	**47**	**2**	**3**	**39**	**47**	**38**	**46**	**1**	**2**	**38**	**46**
9	20	10	10	3	3	10	10	4	4	2	2	4	4	10	10	3	3	10	10
	40	7	7	7	7	7	7	8	8	6	6	8	8	7	7	7	7	7	7
	60	8	8	8	8	8	8	4	4	4	4	4	4	8	8	8	8	8	8
	80	7	7	9	9	7	7	8	8	8	8	8	8	7	7	9	9	7	7
	100	8	8	8	8	8	8	7	7	6	6	7	7	8	8	8	8	8	8
	Tot	**40**	**40**	**35**	**35**	**40**	**40**	**31**	**31**	**26**	**26**	**31**	**31**	**40**	**40**	**35**	**35**	**40**	**40**
10	20	8	9	6	7	8	9	8	9	7	8	8	9	7	8	6	7	7	8
	40	8	10	8	10	8	10	6	8	6	8	6	8	8	10	8	10	8	10
	60	6	9	4	7	6	9	1	4	1	4	1	4	5	8	3	6	5	8
	80	5	10	0	3	4	9	1	3	0	1	1	2	5	10	0	3	4	9
	100	4	10	1	3	4	10	0	4	0	2	0	4	4	10	1	3	4	10
	Tot	**31**	**48**	**19**	**30**	**30**	**47**	**16**	**28**	**14**	**23**	**16**	**27**	**29**	**46**	**18**	**29**	**28**	**45**

5. Conclusion

In this paper, we addressed the deterministic two-dimensional bin-packing problem and considered the version where the items cannot be rotated. We surveyed the polynomial-time lower bounding strategies that were developed so far and we investigated dominance relationships. In this regard, we showed that, in contrast to what was previously claimed in the literature, L^2_{CCM} does not dominate L^2_{BM}. Also, we presented the results of an extensive computational study that was conducted on a large set of benchmark instances, and we provided empirical evidence that $L^2_{CCM,2}$ performs extremely well and outperforms all other lower bounds. Nevertheless, we observed that this bound failed to provide tight values for many large instances (in particular, for Classes 5 and 10). This is a clear indication, that there is still room for further improvement.

TABLE 10. Comparison of $L^2_{CCM,2}$ with Caprara and Monaci (2009) lower bounds.

		$L^2_{CCM,2}$			$L^2_{CM,1}$			$L^2_{CM,2}$		
Class	Nins	Max	Opt	*CPU*	Max	Opt	*CPU*	Max	Opt	*CPU*
1	7	6	5	0.000	6	5	0.001	7	6	0.064
3	11	8	3	0.002	10	5	0.050	11	6	0.686
4	5	5	0	0.001	5	0	0.080	5	0	2.194
5	13	11	1	0.006	13	3	0.065	13	3	1.175
6	4	4	0	0.002	4	0	0.210	4	0	5.745
7	15	7	0	0.004	7	0	0.024	15	8	0.993
8	9	8	0	0.007	8	0	0.049	9	1	1.818
10	19	17	3	0.004	18	4	0.257	19	5	2.443
	83	66	12	0.013	71	17	0.970	83	29	11.210

The last decade has witnessed the resurgence of the concept of dual feasible functions that proved extremely fruitful for generating enhanced lower bounds. We believe that novel ideas are necessary for producing a new generation of tighter lower bounds.

Acknowledgements. This work was carried out in the framework of the Labex MS2T, which was funded by the French Government, through the program "Investments for the future" managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

REFERENCES

[1] C. Alves, J. Valério de Carvalho, F. Clautiaux and J. Rietz, Multidimensional dual-feasible functions and fast lower bounds for the vector packing problem. *Eur. J. Oper. Res.* **233** (2015) 43–63.

[2] J.O. Berkey and P.Y. Wang, Two-dimensional finite bin packing algorithms. *J. Oper. Res. Soc.* **38** (1987) 423–429.

[3] A. Bortfeldt and T. Winter, A genetic algorithm for the two-dimensional knapsack problem with rectangular pieces. *Inter. Trans. Oper. Res.* **16** (2009) 685–713.

[4] M.A. Boschetti, New lower bounds for the three-dimensional finite bin packing problem. *Discrete Appl. Math.* **140** (2004) 241–258.

[5] M.A. Boschetti and L. Montaletti, An exact algorithm for the two-dimensional strip-packing problem. *Oper. Res.* **58** (2010) 1774–1791.

[6] M.A. Boschetti and A. Mingozzi, The two-dimensional finite bin packing problem, Part I: New lower bounds for the oriented case. **4OR** (2003) 27–42.

[7] A. Caprara, A. Lodi and M. Monaci, An approximation scheme for the two-stage, two-dimensional knapsack problem. *Discrete Optimiz.* **7** (2010) 114–124.

[8] A. Caprara and M. Monaci, Bidimensional packing by bilinear programming. *Math. Progr.* **118** (2009) 75–108.

[9] J. Carlier, F. Clautiaux and A. Moukrim, New reduction procedures and lower bounds for the two-dimensional bin packing problem with fixed orientation. *Comput. Oper. Res.* **34** (2007a) 2223–2250.

[10] J. Carlier and E. Néron, Computing redundant resources for the resource constrained project scheduling problem. *Europ. J. Oper. Res.* **176** (2007b) 1452–1463.

[11] G.F. Cintra, F.K. Miyazawa, Y. Wakabayashi and E.C. Xavier, Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation. *Europ. J. Oper. Res.* **191** (2008) 61–85.

[12] F. Clautiaux, J. Carlier, A. Moukrim, A new exact method for the two-dimensional bin-packing problem with fixed orientation. *Oper. Res. Lett.* **35** (2007a) 357–364.

[13] F. Clautiaux, J. Carlier and A. Moukrim, A new exact method for the two-dimensional orthogonal packing problem. *Europ. J. Oper. Res.* **183** (2007b) 1196–1211.

[14] F. Clautiaux, C. Alves and J. Valerio de Carvalho, A survey of dual-feasible and superadditive functions. *Ann. Oper. Res.* **179** (2010) 317–342.

[15] J.F. Côté, M. Dell'Amico and M. Iori, Combinatorial Benders' cuts for the strip packing problem. *Oper. Res.* **62** (2014) 643–661.

[16] Y.P. Cui, Y. Cui and T. Tang, Sequential heuristic for the two-dimensional bin-packing problem. *Europ. J. Oper. Res.* **240** (2015) 43–53.

[17] M. Dell'Amico and S. Martello, Optimal scheduling of tasks on identical parallel processors. *ORSA J. Comput.* **7** (1995) 191–200.

[18] J. Egeblad and D. Pisinger, Heuristic approaches for the two- and three-dimensional knapsack packing problem. *Comput. Oper. Res.* **36** (2009) 1026–1049.

[19] S. Fekete and J. Schepers, New classes of fast lower bounds for bin packing problems. *Math. Program.* **60** (2001) 311–329.

[20] S. Fekete and J. Schepers, A general framework for bounds for higher-dimensional orthogonal packing problems. *Math. Methods Oper. Res.* **60** (2004) 311–329.

[21] S. Fekete, J. Schepers and J.C. van der Veen, An exact algorithm for higher-dimensional orthogonal packing. *Oper. Res.* **55** (2007) 569–587.

[22] G. Fuellerer, K.F. Doerner, R.F. Hartl and M. Iori, Ant colony optimization for the two-dimensional loading vehicle routing problem. *Comput. Oper. Res.* **36** (2009) 655–673.

[23] M. Gendreau, M. Iori, G. Laporte and S. Martello, A tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints. *Networks* **51** (2008) 4–18.

[24] E. Hadjiconstantinou and N. Christofides, An exact algorithm for general, orthogonal, two-dimensional knapsack problem. *Europ. J. Oper. Res.* **83** (1995) 39–56.

[25] S. Hong, D. Zhang, H.C. Lau, X. Zeng and Y. Sic, A hybrid heuristic algorithm for the 2D variable-sized bin packing problem, *Europ. J. Operat. Res.* **238** (2014) 95–103.

[26] M. Kenmochi, T. Imamichi, K. Nonobe, M. Yagiura and H. Nagamochi, Exact algorithms for the two-dimensional strip packing problem with and without rotations. *Europ. J. Oper. Res.* **198** (2009) 73–83.

[27] M. Iori and S. Martello, Routing problems with loading constraints. *TOP* **18** (2010) 4–27.

[28] D.S. Johnson, *Near-Optimal Bin Packing Algorithms.* Ph.D. thesis, Massachusetts Institute of Technology (1973).

[29] A. Lodi, S. Martello and M. Monaci, Two-dimensional packing problems: A survey. *Europ. J. Oper. Res.* **141** (2002a) 241–252.

[30] A. Lodi, S. Martello and D. Vigo, Recent advances on two-dimensional bin packing problems. *Discrete Appl. Math.* **123** (2002b) 379–396.

[31] S. Martello, M. Monaci and D. Vigo, An exact approach to the strip-packing problem. *INFORMS J. Comput.* **15** (2003) 310–319.

[32] S. Martello, M. Monaci, Models and algorithms for packing rectangles into the smallest square. *Comput. Oper. Res.* **63** (2015) 161–171.

[33] S. Martello and P. Toth, Knapsack problems: Algorithms and computer implementations. John Wiley and Sons, Chichester (1990).

[34] S. Martello and D. Vigo, Exact solution of the two-dimensional finite bin packing problem. *Manag. Sci.* **44** (1998) 388–399.

[35] D. Pisinger and M.M. Sigurd, Using decomposition techniques and constraint programming for solving the two-dimensional bin-packing problem. *INFORMS J. Comput.* **19** (2007) 36–51.

[36] L. Wei, T. Tian, W. Zhu and A. Lim, A block-based layer building approach for the 2D guillotine strip packing problem. *Europ. J. Operat. Res.* **239** (2014) 58–69.

[37] L. Wei, Z. Zhang, D. Zhang and A. Lim, A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. *Eur. J. Oper. Res.* **243** (2015) 798–814.

[38] E.E. Zachariadis, C.D. Tarantilis and C.T. Kiranoudis, A guided tabu search for the vehicle routing problem with two-dimensional loading constraints. *Europ. J. Oper. Res.* **195** (2009) 729–743.