Algorithmica manuscript No.
(will be inserted by the editor)

A Tight $(3/2 + \varepsilon)$-Approximation for Skewed Strip Packing *

Waldo Gálvez · Fabrizio Grandoni · Afrouz Jabal Ameli · Klaus Jansen · Arindam Khan · Malin Rau

Received: date / Accepted: date

Abstract In the *Strip Packing* problem, we are given a vertical half-strip $[0, W] \times [0, +\infty)$ and a collection of open rectangles of width at most W. Our goal is to find an axis-aligned (non-overlapping) packing of such rectangles into the strip such that the maximum height OPT spanned by the packing is as small as possible. It is NP-hard to approximate this problem within a factor $(3/2 - \varepsilon)$ for any constant $\varepsilon > 0$ by a simple reduction from the *Partition* problem, while the current best approximation factor for it is $(5/3 + \varepsilon)$.

It seems plausible that Strip Packing admits a $(3/2 + \varepsilon)$-approximation. We make progress in that direction by achieving such tight approximation guarantees

* A preliminary version of this paper appeared in the 23rd International Conference on Approximation Algorithms for Combinatorial Optimization Problems (APPROX 2020).

Waldo Gálvez is supported by the European Research Council, Grant Agreement No. 691672, project APEG. Fabrizio Grandoni and Afrouz Jabal Ameli are partially supported by the SNSF Excellence Grant 200020B_182865/1. Malin Rau is partially supported by the French research program ENERGUMEN ANR-18-CE25-0008.

W. Gálvez
Technical University of Munich, Germany
E-mail: galvez@in.tum.de

F. Grandoni
IDSIA, USI-SUPSI, Switzerland
E-mail: fabrizio@idsia.ch

A. Jabal Ameli
IDSIA, USI-SUPSI, Switzerland
E-mail: afrouz@idsia.ch

K. Jansen
University of Kiel, Germany
E-mail: kj@informatik.uni-kiel.de

A. Khan
Indian Institute of Science, Bengaluru, India
E-mail: arindamkhan@iisc.ac.in

M. Rau
Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP*, LIG, 38000 Grenoble, France
E-mail: malin.rau@inria.fr

for a special family of instances, which we call *skewed instances*. As standard in the area, for a given constant parameter $\delta > 0$, we call *large* the rectangles with width at least δW and height at least δOPT, and *skewed* the remaining rectangles. If all the rectangles in the input are large, then one can easily compute the optimal packing in polynomial time (since the input can contain only a constant number of rectangles). We consider the complementary case where all the rectangles are skewed. This second case retains a large part of the complexity of the original problem; in particular, it is NP-hard to approximate within a factor $(3/2 - \varepsilon)$ and we provide an (almost) tight $(3/2 + \varepsilon)$-approximation algorithm.

Keywords Strip Packing · Rectangle Packing · Approximation Algorithms

1 Introduction

In this paper, we consider the *Strip Packing* problem, a well-studied classical rectangle packing problem (see Section 2 for a formal definition). An instance consists of a vertical half-strip of (integral) width W plus a collection $\mathcal{R}$ of rectangles of width at most W, and our objective is to find an axis-aligned packing of $\mathcal{R}$ (where rectangles do not overlap) such that the maximum height spanned by the packing is as small as possible.

Strip Packing generalizes several well-studied key problems in combinatorial optimization. For example, it generalizes Makespan Minimization on identical machines [11] when all the rectangle widths are 1 (here W would be the number of processors), and also generalizes Bin Packing [12] when all the rectangle heights are 1 (here the height OPT of the optimal solution would be the optimal number of bins). Strip Packing has several natural applications. For example, there are many manufacturing settings where rectangular pieces have to be cut out of some roll of raw material while using a rectangular piece of that roll of minimum length. Another application is the minimization of the peak energy consumption in smart-grids [29,41]: here heights and widths model the energy consumption and duration, respectively, of a given set of jobs. For analogous reasons, it captures scenarios where a given set of jobs needs to be allocated a *consecutive* amount of a given resource (memory locations, frequencies, etc.) for a given amount of time.

Strip Packing is strongly NP-hard [15], and hence it is reasonable to consider approximation algorithms for it. A simple reduction from the *Partition* problem shows that it is not possible to obtain a $(\frac{3}{2} - \varepsilon)$-approximation algorithm (with polynomial running time) for any $\varepsilon > 0$ unless P=NP (more details on this reduction are given later). The first non-trivial approximation algorithm for Strip Packing, with approximation ratio 3, was given by Baker, Coffman, and Rivest [4]. The First-Fit-Decreasing-Height algorithm (FFDH) by Coffman et al. [13] gives a 2.7-approximation. Sleator [39] gave an algorithm that generates a packing of height $2OPT + \frac{h_{max}}{2}$, where h_{max} is the maximum height of a rectangle in the instance, hence achieving a 2.5-approximation. Afterwards, Steinberg [40] and Schiermeyer [38] independently improved the approximation ratio to 2. Harren and van Stee [21] first broke the barrier of 2 with their 1.9396-approximation. The present best $(\frac{5}{3} + \varepsilon)$-approximation is due to Harren et al. [20].

The Strip Packing problem has also been studied in the pseudopolynomial setting, i.e., when $W = n^{O(1)}$. After a series of recent improvements [37,1,18,22, 25], Jansen and Rau [24] have given a pseudopolynomial time algorithm with an almost tight $(\frac{5}{4} + \varepsilon)$-approximation ratio.

In terms of asymptotic approximations, the barrier of $\frac{3}{2}$ can also be beaten. The best results in these terms are an AFPTAS presented by Kenyon and Rémila [30] which produces a solution of height $(1 + \varepsilon)OPT + O\left(\frac{h_{max}}{\varepsilon^2}\right)$, and an APTAS which generates a solution of height $(1 + \varepsilon)OPT + h_{max}$ by Jansen and Solis-Oba [26]. For the variant of Strip Packing *with Rotations*, where the rectangles are allowed to be rotated by 90 degrees, Jansen and van Stee [27] provided an APTAS (see also [14,36] for related results).

1.1 Related Work

Strip Packing has rich connections with many other important geometric packing problems such as Two-dimensional Bin Packing (2BP) and Two-dimensional Geometric Knapsack (2GK). In 2BP, we are given a set of rectangles and unit square bins, and the goal is to pack all the rectangles into minimum number of bins. The problem is known to be APX-hard [6] and the present best approximation ratio is 1.405 [7] (these results hold in the asymptotic regime). In 2GK, we are given a set of rectangles (with associated profits) and a unit square knapsack, and the goal is to pack a subset of rectangles into the knapsack maximizing the total profit. This problem is strongly NP-hard even when all items are squares with unit profits [34]. The present-best approximation ratio is 1.89 due to Gálvez et al. [16] (see also [3, 28,19,32]).

Strip Packing has also been well studied for higher dimensions. The present best asymptotic approximation for 3-D Strip Packing is due to Jansen and Prädel [23] who presented a 1.5-approximation extending techniques from 2BP.

Another related problem is the *Independent Set of Rectangles* problem: here we are given a collection of axis-parallel rectangles embedded in the plane, and we need to find a maximum cardinality/weight subset of non-overlapping rectangles [2,8,9,33,35].

We refer the readers to [10,31] for surveys on geometric packing problems.

1.2 Our Contribution

In this paper, we study a special case of Strip Packing, where all rectangles are *skewed*. In more detail, we say that a rectangle R is δ-*large* if, for some fixed constant $\delta > 0$, its width is at least a δ fraction of the width W of the strip and its height is at least a δ fraction of the height OPT of the optimal packing; otherwise, the rectangle is δ-*skewed*. We just say that a rectangle is large or skewed when δ is clear from the context. An instance of Strip Packing is δ-skewed if all the rectangles in the input are such.

This special case is non-trivial: in particular, the mentioned $3/2-\varepsilon$ hardness of approximation holds also for this special case with minor adaptations (see Section 5). We also believe that this special case is practically relevant: e.g., it captures scenarios where no job can consume a significant amount of the global resource (energy, memory space, etc.) for a significant amount of time. Our main result is as follows (see Sections 3-4).

Theorem 1 *For any given constant $\varepsilon' > 0$ and a small enough positive constant $\delta \le (\varepsilon')^{(1/\varepsilon')^{O(1)}}$, there exists a polynomial-time $\left(\frac{3}{2} + \varepsilon'\right)$-approximation algorithm for δ-skewed Strip Packing.*

We remark that our algorithm does not need to recognize first if the instance is δ-skewed: It always returns a feasible solution, but only if the instance satisfies the requirements, its approximation ratio is guaranteed.

Our result suggests that, in order to obtain a better approximation guarantee for the general case of Strip Packing (possibly $3/2 + \varepsilon$), one of the main obstacles is the interaction between large and skewed rectangles.

1.2.1 Organization.

In Section 2, we introduce some useful notation and preliminary results. In Section 3, we prove the existence of a good enough solution with certain structural properties. The mentioned structure is exploited to derive an algorithm with the claimed approximation guarantee in Section 4. Section 5 contains our hardness of approximation result.

2 Preliminaries

A Strip Packing instance consists of a vertical strip of integral width W in the two-dimensional plane, i.e. $[0, W] \times \mathbb{R}_{\ge 0}$, and a set $\mathcal{R}$ of open rectangles, where each rectangle $R \in \mathcal{R}$ is characterized by its integral height $h(R)$ and integral width $w(R)$. An embedding of $\mathcal{R}$ is given by specifying a bottom-left position $(x(R), y(R))$ for each $R \in \mathcal{R}$. The interpretation is that R is embedded in the plane in the region $\big(x(R), x(R)+w(R)\big) \times \big(y(R), y(R)+h(R)\big)$. An embedding is a feasible packing into the strip if the following two conditions hold: (1) each $R \in \mathcal{R}$ is embedded inside the strip, namely $0 \le x(R) \le W - w(R)$ and $y(R) \ge 0$ and (2) rectangles do not overlap, namely, for any two $R_1, R_2 \in \mathcal{R}$, $\Big(\big(x(R_1), x(R_1) + w(R_1)\big) \times \big(y(R_1), y(R_1)+h(R_1)\big)\Big) \cap \Big(\big(x(R_2), x(R_2)+w(R_2)\big) \times \big(y(R_2), y(R-2) + h(R_2)\big)\Big) = \emptyset$. The height of a feasible packing is the maximum height spanned by any embedded rectangle, namely the maximum value of $y(R) + h(R)$ among the rectangles $R \in \mathcal{R}$. The goal of Strip Packing is to compute a feasible packing of minimum height OPT. Without loss of generality, we can restrict our attention to packings where the coordinates $(x(R), y(R))$ are integral as any feasible packing can be transformed into a feasible packing satisfying this property (intuitively, by pushing rectangles to the bottom-left as much as possible while keeping feasibility).

Given a subset of rectangles $\mathcal{S} \subseteq \mathcal{R}$, we denote by $w(\mathcal{S}) := \sum_{R\in\mathcal{S}} w(R)$, $h(\mathcal{S}) := \sum_{R\in\mathcal{S}} h(R)$, and $a(\mathcal{S}) := \sum_{R\in\mathcal{S}} h(R)w(R)$ the total width, height, and area of $\mathcal{S}$, respectively. The operation of changing the bottom-left corner of a rectangle R in a given packing from $(x(R), y(R))$ to $(x(R), y(R)+a)$ will be denoted by *shifting R vertically by a*. Analogously, changing the bottom-left coordinate from $(x(R), y(R))$ to $(x(R) + a, y(R))$ will be denoted by *shifting R horizontally by a*. These operations are only allowed if the resulting packing is feasible.

A *box* of size $w \times h$ denotes a rectangular region of width w and height h. We sometimes embed boxes into the strip analogously to the way we embed rectangles. A monotone polygonal chain is a curve specified by a sequence of points $(A_1, A_2, \ldots, A_n)$ called its vertices. The curve itself consists of the line segments connecting the consecutive vertices, and we require that the x-coordinates of points A_i are non-decreasing and the segments are horizontal or vertical. We say that a rectangle R in the packing *lies above (resp. below)* one such P if for any $x_1 \in (x(R), x(R)+w(R))$ we have that $y(R)$ (resp. $y(R)+h(R)$) is not smaller (resp. not larger) than the largest (resp., smallest) y-coordinate of P at x-coordinate x_1.

We can assume w.l.o.g. that W is lower bounded by a sufficiently large constant, in particular $W \geq 1/\varepsilon$. If it is not the case, one easily obtains a PTAS for δ-skewed instances[1].

2.1 Next Fit Decreasing Height

One of the most recurring tools, used as a subroutine in countless results on geometric packing problems, is the Next Fit Decreasing Height (NFDH) algorithm [13]. We will use a variant of this algorithm to pack rectangles inside a rectangular box and analyze its properties. We provide a full proof for the sake of completeness.

Suppose we are given a box C of size $w \times h$, and a set of rectangles $\mathcal{R}'$, each one fitting inside the box. NFDH computes in polynomial time a packing of a set $\mathcal{R}'' \subseteq \mathcal{R}'$ as follows. It sorts the rectangles $R \in \mathcal{R}'$ in non-increasing order of height, and considers rectangles in that order $R_1, \ldots, R_n$. Then the algorithm works in rounds $j \geq 1$. At the beginning of round j it is given an index $n(j)$ and a horizontal segment $L(j)$ going from the left to the right side of C (initially $n(1) = 1$ and $L(1)$ is the bottom side of C). In round j the algorithm packs a maximal set of rectangles $R_{n(j)}, \ldots, R_{n(j+1)-1}$, with bottom side touching $L(j)$ one next to the other from left to right (a *shelf*). The segment $L(j+1)$ is the horizontal segment containing the top side of $R_{n(j)}$ and ranging from the left to the right side of C. The process halts at round r when either all rectangles have being packed or $R_{n(r+1)}$ does not fit above $R_{n(r)}$.

The following lemma states the guarantees one can get with respect to the dimensions of the rectangles packed in the box.

Lemma 1 *[13] Let C be a given box of size $w \times h$ and $\mathcal{R}$ be a set of rectangles. Assume that, for some given parameter $\varepsilon' \in (0,1)$, for each $R \in \mathcal{R}$ one has $w(R) \leq \varepsilon' w$ and $h(R) \leq \varepsilon' h$. Then NFDH is able to pack in C a subset $\mathcal{R}' \subseteq \mathcal{R}$ of area at least $a(\mathcal{R}') \geq \min\{a(\mathcal{R}), (1-2\varepsilon')w \cdot h\}$. In particular, if $a(\mathcal{R}) \leq (1-2\varepsilon')w \cdot h$, all rectangles in $\mathcal{R}$ are packed.*

Proof The claim trivially holds if all rectangles are packed. Thus suppose that this is not the case. Observe that $\sum_{j=1}^{r+1} h(R_{n(j)}) > h$, otherwise rectangle $R_{n(r+1)}$ would fit in the next shelf above $R_{n(r)}$; hence $\sum_{j=2}^{r+1} h(R_{n(j)}) > h - h(R_{n(1)}) \geq (1-\varepsilon')h$. Observe also that the total width of rectangles packed in each round j is at least $w - \varepsilon' w = (1-\varepsilon')w$, since $R_{n(j+1)}$, of width at most $\varepsilon' w$, does not fit

[1] Choosing δ such that $\delta W < 1$ enforces each rectangle to have height at most δOPT (otherwise it would be large). A PTAS for this case follows, e.g., from [26].

to the right of $R_{n(j+1)-1}$. It follows that the total area of the rectangles packed in round j is at least $(w - \varepsilon' w)h(R_{n(j+1)-1})$, and thus

$$a(\mathcal{R}') \geq \sum_{j=1}^{r} (1-\varepsilon')w \cdot h(R_{n(j+1)-1}) \geq (1-\varepsilon')w \sum_{j=2}^{r+1} h(R_{n(j)})$$
$$\geq (1-\varepsilon')^2 w \cdot h \geq (1-2\varepsilon')w \cdot h.$$

2.2 Container Packings

Similar to recent work on related problems (e.g., [16,5]), we will exploit a *container-based* packing approach. The idea is to partition the solution into a constant number of axis-aligned rectangular regions (*containers*). The sizes (and therefore positions) of these containers can be *guessed* in polynomial time, and subsequently, rectangles are packed inside the containers in a simple way: either one next to the other from left to right (*vertical container*), or one on top of the other from bottom to top (*horizontal container*), or by means of NFDH (*area container*). We further require that the rectangles R packed into an area container of size $w \times h$ satisfy $w(R) \leq \varepsilon' w$ and $h(R) \leq \varepsilon' h$ for a constant $\varepsilon' > 0$ to be fixed later. We call this an ε'-area container.

We will make use of the following standard PTAS to pack rectangles into a constant number of containers. The basic idea is to reduce the problem to an instance of the Maximum Generalized Assignment Problem (GAP) with one bin per container, and then use a PTAS for the latter problem plus NFDH to repack rectangles in area containers. We recall that in GAP, we are given a collection of n items and a set of k (one-dimensional) bins, each one characterized by a positive size. Each item has a profit[2] and a positive size per bin (possibly different for different bins). Our goal is to compute a maximum profit subset of items and an assignment of them into the bins so that the total size of items packed in each bin is at most the size of the bin. GAP admits a PTAS for constant k (see e.g. Section E.2 in [17]) and the following lemma shows how to use it to pack the rectangles into a given set of containers.

Lemma 2 *For any constant $\varepsilon' > 0$, given a set of rectangles $\mathcal{R}$ that can be packed into a given set of k containers (each container being either vertical, horizontal or ε'-area), k constant, there is an algorithm to pack $\mathcal{R}' \subseteq \mathcal{R}$ with $a(\mathcal{R}') \geq (1 - 3\varepsilon')a(\mathcal{R})$ into the mentioned containers.*

Proof We let $w(C_j) \times h(C_j)$ be the size of the j-th container C_j. We build an instance of GAP as follows. We define an item R per rectangle $R \in \mathcal{R}$, with profit $a(R)$. For each horizontal container C_j, we create a knapsack j of size $S_j := h(C_j)$. Furthermore, we define the size $s(R, j)$ of rectangle R w.r.t. knapsack j as $h(R)$ if $h(R) \leq h(C_j)$ and $w(R) \leq w(C_j)$. Otherwise $s(R, j) = +\infty$ (meaning that R does not fit in C_j). The construction for vertical containers is symmetric. For each area container C_j we create a knapsack j of size $S_j = a(C_j)$ and define the size $s(R, j)$ of rectangle R w.r.t. knapsack j as $a(R)$ if $h(R) \leq \varepsilon' h(C_j)$ and $w(R) \leq \varepsilon' w(C_j)$,

[2] The same item might have different profit on different knapsacks; however, we do not need this extension here.

setting $b(R,j) = +\infty$ otherwise (meaning that the rectangle is not small enough with respect to the dimensions of the container).

We next apply the mentioned PTAS for GAP to this instance, so as to obtain a solution $\mathcal{R}''$ to GAP of profit at least $a(\mathcal{R}'') \geq (1-\varepsilon')a(\mathcal{R})$. We build a feasible packing of $\mathcal{R}' \subseteq \mathcal{R}''$ into the containers as follows. Let $\mathcal{R}_j$ be the items packed into knapsack j. If C_j is vertical, we pack rectangles $\mathcal{R}_j$ into this container bottom-most and from left to right one next to the other in any order. By definition all rectangles $\mathcal{R}_j$ will fit. A symmetric construction works if C_j is horizontal. If C_j is area, we pack a subset $\mathcal{R}'_j$ of $\mathcal{R}_j$ into it using NFDH. By Lemma 1, we either have $\mathcal{R}'_j = \mathcal{R}_j$, or it must be the case that $a(\mathcal{R}'_j) \geq (1-2\varepsilon')w(C_j)h(C_j) = (1-2\varepsilon')a(C_j)$. Consider the second case. Let $\mathcal{R}''_j = \mathcal{R}_j \setminus \mathcal{R}'_j$ be the rectangles which are not packed. Observe that $a(\mathcal{R}_j) \leq a(C_j)$ by the feasibility of the GAP solution, hence

$$\frac{a(\mathcal{R}''_j)}{a(\mathcal{R}_j)} = 1 - \frac{a(\mathcal{R}'_j)}{a(\mathcal{R}_j)} \leq 1 - \frac{(1-2\varepsilon')a(C_j)}{a(\mathcal{R}_j)} \leq 2\varepsilon'.$$

Thus altogether $a(\mathcal{R}') \geq a(\mathcal{R}'')(1-2\varepsilon') \geq a(\mathcal{R})(1-2\varepsilon')(1-\varepsilon') \geq a(\mathcal{R})(1-3\varepsilon')$.

Notice that the containers may have considerable free space inside, but the lemma just claims that the total area of the rectangles that the algorithm is not packing is negligible. Whenever this lemma is applied, we will pack the remaining rectangles into an extra rectangular box of small area and carefully argue where to place it.

2.3 Classification of Rectangles

From now on, we will assume that instance $(\mathcal{R}, W)$ is δ-skewed for some $\delta > 0$ to be fixed later. By OPT, we denote both the optimal height and an optimal packing: the meaning will be clear from the context. We can assume that OPT is even (otherwise, we can multiply heights by a factor 2).

We will assume that our algorithm is given in the input a value OPT' such that $OPT \leq OPT' \leq (1+\varepsilon)OPT$. This assumption can be removed as follows. We compute, say, a 2-approximate solution APX for the instance by means of Steinberg's algorithm [40] and then run our algorithm for all the (constantly many) values $OPT' = (1+\varepsilon)^j \frac{APX}{2(1+\varepsilon)}$ which fit in the range $[\frac{APX}{2(1+\varepsilon)}, APX(1+\varepsilon)]$. One of these values will satisfy the claim. In order to keep the notation light, we simply use OPT to denote this value OPT'. Therefore, all the approximation factors should be scaled by a factor $(1+\varepsilon)$ in order to consider the true value of OPT.

Along this work we will assume ε to be a positive constant and also for simplicity that $\frac{1}{\varepsilon} \in \mathbb{N}$. We will classify the rectangles according to their heights as follows:

- The set of *tall* rectangles $\mathcal{T} = \{R \in \mathcal{R} : h(R) > \frac{1}{2}OPT\}$;
- The set of *vertical* rectangles $\mathcal{V} = \{R \in \mathcal{R} : h(R) \in (\delta OPT, \frac{1}{2}OPT]\}$;
- The set of *short* rectangles $\mathcal{S} = \{R \in \mathcal{R} : h(R) \leq \delta OPT\}$;

2.4 Linear Grouping

We need the following lemma whose proof is based on *linear grouping*, a standard technique in the area of packing problems. Given a subset $\mathcal{S}$ of rectangles, let

$\mathcal{S}_{hslice}$ be the set of rectangles obtained by taking each $R \in \mathcal{S}$ and replacing it with $h(R)$ rectangles of height 1 and width $w(R)$ (*horizontal slices*). We define symmetrically the set $\mathcal{S}_{vslice}$ of *vertical slices*. Notice that any embedding of $\mathcal{S}$ naturally induces an embedding of $\mathcal{S}_{hslice}$ and $\mathcal{S}_{vslice}$.

Lemma 3 *Let $\varepsilon' > 0$ be a given constant, $\mathcal{P}$ be a rectangular region of size $W \times H$ and $\mathcal{H}$ be a subset or rectangles of height at most $\delta \cdot H$ each for some constant $\delta \in (0, 1]$. Suppose that $\mathcal{H}_{hslice}$ can be packed into a set $\mathcal{B}$ of $K = O_{\varepsilon'}(1)$ boxes contained in $\mathcal{P}$. Then, for $\delta \leq (\varepsilon'/K)^{(K/\varepsilon')^{O(1)}}$, there exists a partition of $\mathcal{H}$ into two sets $\mathcal{H}^{cont}$ and $\mathcal{H}^{disc}$ such that:*

1. *$\mathcal{H}^{cont}$ can be packed into a set of at most $K' = O_{\varepsilon'}(1)$ horizontal and ε'-area containers, where each container is fully contained in some box in $\mathcal{B}$.*
2. *$\mathcal{H}^{disc}$ can be packed into one horizontal container of size $\max_{R \in \mathcal{H}}\{w(R)\} \times (\varepsilon')^2 H$ and one ε'-area container of size $\varepsilon' W \times \varepsilon' H$.*
3. *The sizes of the above containers belong to a set that can be computed in polynomial time.*

A symmetric claim holds for a subset of rectangles $\mathcal{V}'$ of width at most $\delta \cdot W$ such that $\mathcal{V}'_{vslice}$ can be packed into the corresponding boxes.

Proof We prove the claim for $\mathcal{H}$, the case of $\mathcal{V}'$ being symmetric. For a proper parameter $\alpha > 0$ to be fixed later, we define a rectangle R (and its horizontal slices) to be *narrow* if $w(R) \leq \alpha^2 W$ and *wide* otherwise. We temporarily remove narrow rectangles, and start by computing a packing for the wide rectangles.

The first step in our construction is to round up the widths of the wide slices, while discarding a subset of them of small area. Let $\beta > 0$ be a parameter to be fixed later. Let us sort the wide slices $\mathcal{H}^{wide}_{hslice}$ in non-increasing order of width, and let us partition the obtained sequence into subsequences $\mathcal{H}_1, \ldots, \mathcal{H}_{1/\beta}$ of total height $\beta h(\mathcal{H}^{wide}_{hslice})$ each (excluding possibly the last group that can have smaller height). For a group i, we define w_i^{min} as the minimum width in $\mathcal{H}_i$. For each $i = 1, \ldots, 1/\beta - 1$, we define an injection between $\mathcal{H}_{i+1}$ and $\mathcal{H}_i$. Next, we delete slices $\mathcal{H}_1$. Let $\mathcal{H}_{wdisc1}$ denote the rectangles of which we removed at least one slice. Notice that all rectangles having some slice in $\mathcal{H}_1$ have all their slices in $\mathcal{H}_1$ excluding possibly one rectangle (which has part of its slices in $\mathcal{H}_2$). Observe that $h(\mathcal{H}^{wide}_{hslice}) \leq \frac{H}{\alpha}$ since otherwise $a(\mathcal{H}^{wide}_{hslice}) \geq \alpha W \cdot h(\mathcal{H})$, which would be too large to fit into the region of size $W \times H$. Hence $h(\mathcal{H}_1) = \beta h(\mathcal{H}) \leq \frac{\beta}{\alpha} H$. It follows that $h(\mathcal{H}_{wdisc1}) \leq (\frac{\beta}{\alpha} + \delta)H$. For any fixed α, this quantity is at most $\frac{(\varepsilon')^2}{2} H$ if $\beta \leq \alpha(\varepsilon')^2/4$ and $\delta \leq (\varepsilon')^2/4$.

For $i = 1, \ldots, 1/\beta - 1$, we temporarily increase the width of each $H \in \mathcal{H}_{i+1}$ to w_i^{min}, hence getting an enlarged slice $\overline{H}$. Then, we move each such $\overline{H}$ into the region that was occupied by the slice $H' \in \mathcal{H}_i$ associated with H according to the above injection. Notice that this is possible since we removed $\mathcal{H}_1$ and since $w(\overline{H}) = w_i^{min} \leq w(H')$. Let $\overline{\mathcal{H}}$ be the final set of enlarged slices. Observe that the number of possible distinct widths in $\overline{\mathcal{H}}$ is $1/\beta - 1$.

Let us focus on a specific box $B \in \mathcal{B}$ of size $w(B) \times h(B)$, and let $\mathcal{H}'_{hslice}$ be the slices contained in B. Next, we partition B into unit height stripes. We shift slices in each stripe as left as possible, and permute them so that slices in $\overline{\mathcal{H}}$ appear to the left of each stripe. We call a *configuration* C of a stripe the sequence of (enlarged) widths $(w_1, \ldots, w_q)$ of its slices in $\overline{\mathcal{H}}$ from left to right.

Notice that there are $1/\beta - 1$ possible enlarged widths, and each stripe can contain at most $1/\alpha^2$ wide slices. Hence the number of possible configurations is at most $n_{conf} = \sum_{i=0}^{1/\alpha^2} (1/\beta - 1)^i \leq 2(1/\beta - 1)^{1/\alpha^2} \leq (1/\beta)^{1/\alpha^2}$.

We reorder the stripes in $\mathcal{H}'_{hslice}$ vertically so that equal configurations appear consecutively from top to bottom, and stripes without narrow rectangles appear at the bottom. Suppose that the number of stripes in B with a given configuration $C = (w_1, \ldots, w_q)$ is $h(C)$, and $A(C)$ is the corresponding region. We initially cover $A(C)$ by creating q consecutive horizontal containers of height $h(C)$ and width $w_1, \ldots, w_q$ respectively. These containers altogether cover all the wide slices in B. The width of each container belongs to a set that can be computed in polynomial time (it is the width of some input rectangle). In order to enforce the same property for their heights, we round down the height of each such container to the largest multiple $h'(C)$ of $\frac{\delta}{\gamma}H$ not larger than $h(C)$, for some parameter $\gamma > 0$ to be fixed later. The number of these containers is $n_{wcont} \leq Kn_{conf}$.

We next use the obtained horizontal containers to place most of the wide rectangles. We consider the containers in non-increasing order of width and the slices of wide rectangles in the same order, breaking ties so that slices of the same rectangle appear consecutively. We also create a dummy final container of sufficient width and of height large enough to accommodate the total height of the wide slices minus the total height of the containers. Now, we place back the slices into the containers following the previous order. Notice that all slices will fit. We discard each wide rectangle whose slices are contained in two containers (three is not possible) and all the wide rectangles whose slices are contained in the dummy final container. Let $\mathcal{H}_{wdisc2}$ be the set of discarded rectangles. Their total height is

$$h(\mathcal{H}_{wdisc2}) \leq n_{wcont}\delta H + n_{wcont}\frac{\delta}{\gamma}H.$$

The above quantity is at most $\frac{(\varepsilon')^2}{2}H$ for any choice of ε', α, β, and γ, provided that $\delta \leq \frac{(\varepsilon')^2\gamma}{4n_{wcont}} \leq \frac{(\varepsilon')^2\gamma}{4K(1/\beta)^{(1/\alpha^2)}}$.

So we packed all the wide rectangles into horizontal containers except for the set $\mathcal{H}_{wdisc} = \mathcal{H}_{wdisc1} \cup \mathcal{H}_{wdisc2}$. The latter set has, by the above discussion, height at most $(\varepsilon')^2 H$, hence we can pack it into a container of size $\max_{R\in\mathcal{H}^{wide}}\{w(R)\} \times (\varepsilon')^2 H$.

It remains to pack the narrow rectangles. Consider again a given box B. For each configuration C, there is some free region $F(C)$ to the right of the containers built for C whose height is $h'(C)$ (in particular, a multiple of $\frac{\delta}{\gamma}H$) and of some width $w(F(C))$. We build an area container of the same height and having width equal to the largest multiple $w'(F(C))$ of $\frac{\alpha}{\gamma}W$ not larger than $w(F(C))$. We apply a similar construction to the free rectangular region F in B below all the previous containers, if any; in particular, we create an area container whose width is the largest multiple of $\frac{\alpha}{\gamma}W$ not larger than $w(F) = w$ and whose height is the largest multiple of $\frac{\delta}{\gamma}H$ not larger than $h(F)$. The total number of constructed area containers is $n_{ncont} \leq K \cdot n_{conf}$.

Next, we start packing the narrow rectangles in non-increasing order of height into the area containers using NFDH. Observe that these rectangles satisfy the claim of Lemma 1 with parameter γ. If all narrow rectangles are packed this way, we are done. Otherwise, let $\mathcal{H}_{ncont}$ and $\mathcal{H}_{ndisc}$ be the subset of narrow rectangles that are packed and not packed in the area containers, respectively. By Lemma 1, $a(\mathcal{H}_{ncont}) \geq (1-2\gamma)A_{acont}$, where A_{acont} is the total area of the area containers. Let a_{free} be the total area in the boxes not occupied by horizontal containers. Clearly $a_{free} \geq a(\mathcal{H}^{narrow})$ since all narrow slices did fit in a region of area not smaller than a_{free}. Due to the rounding involved in the construction, in each box there is some area which is not used by area containers nor by horizontal ones. The latter area is at most $W \cdot \frac{\delta}{\gamma}H + H \cdot \frac{\alpha}{\gamma}W$ per container, hence being at most $\Delta \leq K \cdot (\frac{\delta+\alpha}{\gamma})WH$ in total. We can conclude that

$$\begin{aligned} a(\mathcal{H}_{ncont}) &\geq (1-2\gamma)A_{acont} = (1-2\gamma)(a_{free} - \Delta) \\ &\geq (1-2\gamma)(a(\mathcal{H}^{narrow}) - K \cdot (\frac{\delta+\alpha}{\gamma})WH). \end{aligned}$$

Thus

$$a(\mathcal{H}_{ndisc}) \leq 2\gamma \cdot a(\mathcal{H}^{narrow}) + K \cdot (\frac{\delta+\alpha}{\gamma})WH) \leq (2\gamma + K \cdot (\frac{\delta+\alpha}{\gamma}))WH.$$

If we choose $\gamma \leq (\varepsilon')^2/6$, $\delta \leq \frac{(\varepsilon')^2\gamma}{6K}$ and $\alpha \leq \frac{(\varepsilon')^2\gamma}{6K}$, then the latter quantity is at most $\frac{(\varepsilon')^2}{2}WH$. Next, we create a new area container C_{darea} of size $\varepsilon' W \times \varepsilon' H$, and use NFDH to pack $\mathcal{H}_{ndisc}$ in it. It is not difficult to verify that, for such values of δ and α, rectangles in $\mathcal{H}_{ndisc}$ satisfy the conditions of Lemma 1 with parameter ε'. Thus we have

$$a(\mathcal{H}_{ndisc}) \leq \frac{1}{2}(\varepsilon')^2 WH = \frac{1}{2}a(C_{darea}) \leq (1-2\varepsilon')a(C_{darea}),$$

implying that all the rectangles in $\mathcal{H}_{ndisc}$ are indeed packed into C_{darea}.

It is possible to choose constant parameters α, β and γ such that the above conditions are all satisfied (for δ small enough) and the total number of containers is $O_{\varepsilon'}(1)$. More precisely, this is true if $\gamma = (\varepsilon')^2/6$, $\alpha = (\varepsilon')^4/(36 \cdot K)$, $\beta = (\varepsilon')^6/(144 \cdot K)$ and $\delta = \frac{(\varepsilon')^2\gamma}{4K(1/\beta)^{(1/\alpha^2)}} \in (\varepsilon'/K)^{(K/\varepsilon')^{O(1)}}$, leading to at most $(K/\varepsilon')^{(K/\varepsilon')^{O(1)}}$ containers. By the above construction, the sizes of the containers belong to a set that can be computed in polynomial time.

3 Existence of a Structured Solution

In this section, we will prove our main structural result.

Theorem 2 *For any given constant $\varepsilon > 0$ and any given instance of δ-skewed Strip Packing $(\mathcal{R}, W)$ with $\delta = \Omega_\varepsilon(1)$ small enough, there exists a feasible container packing such that the following holds:*

1. *The total height of the packing is $\left(\frac{3}{2} + O(\varepsilon)\right) OPT$.*
2. *The number of containers is $O_\varepsilon(1)$ and their possible sizes belong to a set that can be computed in polynomial time.*
3. *Given any fixed ordering of $\mathcal{T}$ in non-increasing order of height, $\mathcal{T}$ can be partitioned into subsequences each one fitting in precisely one vertical container.*
4. *It is possible to pack an extra rectangle (**free box**) of size $\varepsilon^2 W \times \frac{1}{2}OPT$ into the strip without increasing its final height.*

To achieve the above result, we proceed in three steps:

1. We describe a packing of $\mathcal{T} \cup \mathcal{S}_{hslice}$ with height at most $(3/2+O(\varepsilon))OPT$ (see Section 3.1) into $O_\varepsilon(1)$ boxes. This packing leaves a free space of at least $(1/2+\Omega(\varepsilon))OPT + a(\mathcal{V})$.
2. We describe how to pack $\mathcal{V}_{vslice}$ within the free space of the previous packing using $O_\varepsilon(1)$ extra boxes (see Section 3.2). Furthermore, we guarantee that there is a free box (not containing any rectangle) of size at least $\Omega(\varepsilon)W \times \frac{1}{2}OPT$. Guaranteeing the latter property is critical, and it is the main technical novelty in our approach.
3. Finally, we convert the above packing into a feasible container packing (via Lemma 3) inside the above boxes (see Section 3.3). The residual containers that do not fit into the boxes can be placed inside the free box (still leaving enough space) plus a new box of size $W \times O(\varepsilon^2)OPT$ that can be placed on top of the previous packing.

The reason for leaving a free box will be clearer in Section 4, where we will describe our final algorithm.

3.1 Packing of $\mathcal{T} \cup \mathcal{S}_{hslice}$

In this section, we describe a packing of $\mathcal{T} \cup \mathcal{S}_{hslice}$. The proof of the following Lemma is illustrated in Figure 1.

Lemma 4 *For any given constant $\varepsilon \in (0, 1/4]$ with $1/\varepsilon$ integral and $\delta = \Omega_\varepsilon(1)$ sufficiently small, it is possible to pack $\mathcal{T} \cup \mathcal{S}_{hslice}$ into the region $\mathcal{P} = [0, W] \times [0, (\frac{3}{2} + 15\varepsilon)OPT]$ in such a way that:*

1. *Rectangles in $\mathcal{T}$ are packed into at most $1/\varepsilon$* vertical boxes*, slices in $\mathcal{S}_{hslice}$ are packed into at most $1/\varepsilon + 1$* horizontal boxes*, and the remaining area is partitioned into at most $2/\varepsilon$* free boxes*. Furthermore, given any fixed ordering of $\mathcal{T}$ in non-increasing order of height, it is possible to partition $\mathcal{T}$ into subsequences such that each subsequence fits into precisely one vertical box.*
2. *The sizes of the boxes belong to a set that can be computed in polynomial time.*
3. *The total area of the free boxes is at least $(\frac{1}{2} + 9\varepsilon)OPT \cdot W + a(\mathcal{V})$.*

Proof Consider the embedding of $\mathcal{T} \cup \mathcal{S}_{hslice}$ induced by the optimal solution. Let us draw the horizontal line $y = \frac{1}{2}OPT$ and partition $\mathcal{S}_{hslice}$ into two sets $\mathcal{S}^{top}_{hslice}$ and $\mathcal{S}^{bottom}_{hslice}$ corresponding to the rectangles in $\mathcal{S}_{hslice}$ which are packed above and below the line $y = \frac{1}{2}OPT$ respectively (notice that this line does not intersect any rectangle in S_{hslice} as OPT is even by assumption). If we shift up rectangles in $\mathcal{S}^{bottom}_{hslice}$ by OPT, we obtain a feasible packing (since the region $[0, W] \times [OPT, \frac{3}{2}OPT]$ was empty) with final height at most $\frac{3}{2}OPT$. Notice that every rectangle in $\mathcal{T}$ intersects the horizontal segment $[0, W] \times \{\frac{1}{2}\}$.

Now, let us shift down each rectangle R in $\mathcal{T}$ so that its bottom coordinate becomes zero (again, the packing remains feasible). Next, we shift rectangles so that the ones in $\mathcal{T}$ appear one next to the other in the bottom left part of the packing, in non-increasing order of height. To this end, we proceed recursively as follows. Let $T_1, \ldots, T_q$ be the considered ordering of $\mathcal{T}$ in non-increasing order of height. At the beginning of iteration $i \geq 1$, we are given a feasible packing

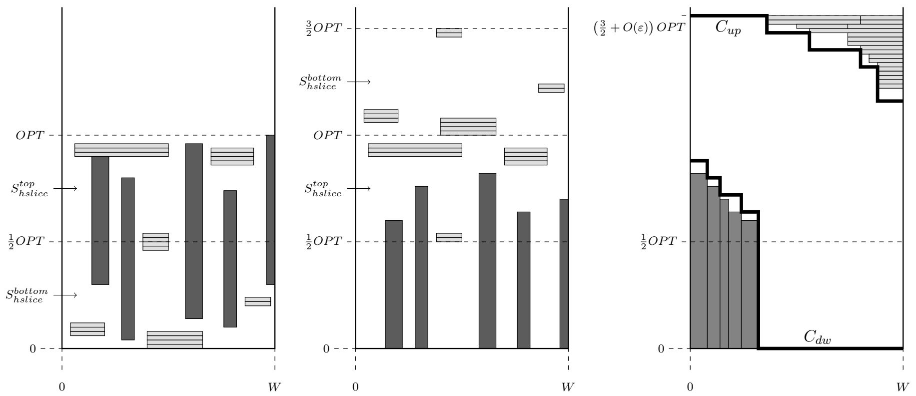

Fig. 1 Depiction of the proof of Lemma 4. **Left:** Packing of $\mathcal{T} \cup \mathcal{S}_{hslice}$ in the optimal solution. Light gray rectangles correspond to $\mathcal{S}_{hslice}$, dark gray rectangles correspond to $\mathcal{T}$. **Center:** By shifting $\mathcal{S}^{bottom}_{hslice}$ to the top we can shift down the rectangles in $\mathcal{T}$. **Right:** We can shift now horizontally rectangles in $\mathcal{T}$ and sort stripes. Finally one obtains C_{up} and C_{dw}.

where $T_1, \dots, T_{i-1}$ are packed from left to right one next to the other as left as possible (and with bottom coordinate 0). We consider the region $A_i := [w_{i-1}, W] \times [0, h(T_i)]$, where $w_{i-1} = \sum_{j \le i-1} w(T_j)$. Let L_i be the portion of A_i to the left of (the current embedding of) T_i. Note that every rectangle is either completely contained in or disjoint from L_i since T_i is the tallest rectangle contained in A_i and T_{i-1} is taller than T_i. We move T_i so that its left coordinate is w_{i-1}, and shift L_i to the right by $w(T_i)$, moving consistently all rectangles in L_i. Obviously the new packing satisfies the invariant for the next iteration. At the end of iteration q the packing satisfies the claim.

In the next step, we partition the area not occupied by $\mathcal{T}$ into unit-height stripes. Notice that each rectangle in $\mathcal{S}_{hslice}$ is fully contained in some stripe. We need (for a reason that will be clearer later) to temporarily discard, meaning that we remove them from the packing, some slices as follows. Let us say that a slice is *wide* if its width is at least εW, and *narrow* otherwise. Consider the slices $\mathcal{S}'_{hslice}$ in a given horizontal stripe. This set contains at most $1/\varepsilon$ wide slices. Let w' be the total width of the remaining narrow slices and let $w'' \le w'$ be the largest multiple of εW. We discard a minimal subset of narrow slices so that the remaining ones have width at most w''. We let $\mathcal{S}^{disc}_{hslice}$ be the set of discarded slices, and $\mathcal{S}^{sel}_{hslice}$ be the remaining (selected) slices.

Next, we push slices $\mathcal{S}^{sel}_{hslice}$ as right as possible. Afterward, we permute the y-coordinates of slices in pairs of stripes so that stripes are sorted from top to bottom in non-increasing order of the total width of the slices contained in them. Observe that this cannot create any conflict with $\mathcal{T}$ (hence the packing remains feasible).

We proceed as follows. We shift up $\mathcal{S}^{sel}_{hslice}$ by $11\varepsilon \cdot OPT$ and construct a polygonal chain C_{up} with the following procedure. The chain starts at coordinate $p^{up}_0 = (0, (\frac{3}{2} + 11\varepsilon)OPT)$. We extend the chain to the right (possibly by a zero amount) until the chain hits some rectangle in $\mathcal{S}^{sel}_{hslice}$. We denote this point by $(x_1, (\frac{3}{2} + 11\varepsilon)OPT)$ and extend the chain down by εOPT, hence reaching some point $p^{up}_1 = (x_1, (\frac{3}{2} + 10\varepsilon)OPT)$. We continue from p^{up}_1 in the same fashion.

The procedure ends when the chain reaches the x-coordinate W. Observe that, by construction, every rectangle in $\mathcal{S}^{sel}_{hslice}$ lies above C_{up}. Furthermore, C_{up} is defined by at most $1/\varepsilon$ axis parallel segments.

Afterward, we build symmetrically a polygonal chain C_{dw} as follows. We start from $p^{dw}_0 = (W, 0)$ and extend the chain to the left (possibly by a zero amount) until we hit some rectangle R in $\mathcal{T}$. Then, we extend the chain up by εOPT, hence reaching a point $p^{dw}_1 = (x_1, \varepsilon OPT)$. We continue from p^{dw}_1 in the same fashion until we reach the x-coordinate 0. Notice that each rectangle from $\mathcal{T}$ lies below C_{dw}. Notice also that C_{dw} is defined by at most $1/\varepsilon$ axis parallel segments. Furthermore, it is fully below C_{up}. To see the latter, take any coordinate $x \in (0, W)$ (which, for simplicity, is not the position of a vertex of any one of the two chains). Let y_{up} and y_{dw} be the corresponding y-coordinates in C_{up} and C_{dw}, resp. Suppose by contradiction that $y_{dw} > y_{up}$. Observe that by construction the segment $x \times (y_{up}, y_{up} + \varepsilon OPT)$ must hit some rectangle R_{up} in $\mathcal{S}^{sel}_{hslice}$. Symmetrically, the segment $x \times (y_{dw} - \varepsilon OPT, y_{dw})$ must hit some rectangle R_{dw} in $\mathcal{T}$. This however implies that R_{up} and R_{dw} overlapped before the shifting up of $\mathcal{S}^{sel}_{hslice}$ by $11\varepsilon OPT$, a contradiction[3].

We claim that points p^{up}_i and p^{dw}_i have coordinates that belong to a set that can be computed in polynomial time. Notice that the y coordinates of the points p^{up}_i and p^{dw}_i are multiples of εOPT (recall that OPT is known to the algorithm), hence they satisfy the requirement. Since we know the precise packing of $\mathcal{T}$, we can compute the x-coordinates of points p^{dw}_i explicitly. The x-coordinates of points p^{up}_i have value W minus the sum w'' of the widths of the slices in a given strip. By the previous discarding procedure, w'' is the sum of up to $1/\varepsilon$ widths of input rectangles, plus a multiple of εW. Hence we can compute the set of the possible coordinates in polynomial time.

Let us subdivide the area in the strip between C_{up} and $[0, W] \times (\frac{3}{2} + 11\varepsilon)OPT$ by extending to the right the horizontal segments in C_{up}. This gives up to $1/\varepsilon$ boxes $\mathcal{B}_{up}$ that fully contain $\mathcal{S}^{sel}_{hslice}$. Symmetrically, we can subdivide the area in the strip between C_{dw} and $[0, W]$ by extending down the vertical segments in C_{dw}. This provides up to $1/\varepsilon$ boxes $\mathcal{B}_{dw}$ that fully contain $\mathcal{T}$. Next, consider the free area between $\mathcal{B}_{up}$ and $\mathcal{B}_{dw}$. By extending down the vertical sides of the boxes in $\mathcal{B}_{up}$ until reaching $\mathcal{B}_{dw}$ and symmetrically extending up the vertical sides of the boxes in $\mathcal{B}_{dw}$ until reaching $\mathcal{B}_{up}$, we obtain a partition of the free area into up to $2/\varepsilon$ free boxes $\mathcal{B}_{free}$. By the previous discussion, the possible sizes of all the mentioned boxes can be computed in polynomial time.

It remains to pack $\mathcal{S}^{disc}_{hslice}$. To that end, we create a new box $\mathcal{B}_{disc}$ of width W and height $4\varepsilon OPT$ that we place on top of the current packing, hence increasing the total height to $(\frac{3}{2}+15\varepsilon)OPT$. Notice that each $R \in \mathcal{S}^{disc}_{hslice}$ satisfies $w(R) \le \varepsilon W$ and $h(R) = 1 \le \delta OPT$. Hence assuming $\varepsilon \le 1/4$ and $\delta \le 4\varepsilon^2$, Lemma 1 with $\varepsilon' = 1/4$ guarantees that $\mathcal{S}^{disc}_{hslice}$ can be fully packed in $\mathcal{B}_{disc}$ via NFDH.

Properties (1) and (2) follow by construction, it just remains to prove (3). Notice that by construction the area inside $\mathcal{B}_{up}$ not occupied by $\mathcal{S}^{sel}_{hslice}$ is at most $\varepsilon OPT \cdot W$. Indeed, as observed earlier, if we take any point (x, y) along C_{up}, where x does not correspond to a step of C_{up}, the segment $x \times [y, y + \varepsilon OPT]$ hits some rectangle in $\mathcal{S}^{sel}_{hslice}$. Thus $a(\mathcal{B}_{up}) \le a(\mathcal{S}^{sel}_{hslice}) + \varepsilon OPT \cdot W$. A symmetric argument

[3] A shifting up by $2\varepsilon OPT$ would be sufficient to achieve a contradiction here. The extra shift by $9\varepsilon OPT$ is used to create some more free space that is needed in the following arguments.

shows that $a(\mathcal{B}_{dw}) \leq a(\mathcal{T}) + \varepsilon OPT \cdot W$. We can therefore conclude that

$$\begin{aligned} a(\mathcal{B}_{free}) &\geq W \cdot \left(\frac{3}{2} + 11\varepsilon\right) OPT - a(\mathcal{S}^{sel}_{hslice}) - a(\mathcal{T}) - 2\varepsilon OPT \cdot W \\ &\geq W \cdot \left(\frac{3}{2} + 9\varepsilon\right) OPT - a(\mathcal{S} \cup \mathcal{T}) \\ &\geq W \cdot \left(\frac{3}{2} + 9\varepsilon\right) OPT - OPT \cdot W + a(\mathcal{V}) = W \cdot \left(\frac{1}{2} + 9\varepsilon\right) OPT + a(\mathcal{V}). \end{aligned}$$

3.2 Including $\mathcal{V}_{vslice}$

In this section, we show how to incorporate $\mathcal{V}_{vslice}$ into the packing from the previous subsection. Critically, we need to leave a free box of sufficiently large size.

Lemma 5 *Consider the packing from Lemma 4 and assume* ε *is small enough. It is possible to pack* $\mathcal{V}_{vslice}$ *inside the free boxes and furthermore define an empty rectangular region of size* $2\varepsilon^2 W \times \frac{1}{2}OPT$ *inside one of the free boxes.*

Proof Consider the set of (at most) $2/\varepsilon$ free boxes $\mathcal{B}_1, \dots, \mathcal{B}_q$ sorted non-decreasingly by height. We partition them into unit-width vertical stripes $\mathcal{S}' = \{S_1, \dots, S_k\}$ sorted in the same order, and breaking ties so that stripes of the same box appear consecutively. Recall that $a(\mathcal{S}') \geq \left(\frac{1}{2} + 9\varepsilon\right)OPT \cdot W + a(\mathcal{V})$.

We next place slices of $\mathcal{V}_{vslice}$ into these stripes from bottom to top in a greedy manner. In particular, we consider rectangles R in $\mathcal{V}_{vslice}$ in any order, and place R in the left-most stripe where it fits, as low as possible. Assume that the non-empty stripes are $\mathcal{S}'_{used}$. Notice that the unused space in these stripes is at most $(w(\mathcal{S}'_{used}) - 1)(\frac{OPT}{2} - 1) + OPT \leq (\frac{w(\mathcal{S}'_{used})+1}{2})OPT$, hence

$$a(\mathcal{S}'_{used}) \leq a(\mathcal{V}) + \frac{w(\mathcal{S}'_{used}) + 1}{2} OPT.$$

Let us partition the unused stripes $\mathcal{S}'_{unused}$ into $\mathcal{S}'^{tall}_{unused}$, the ones of height at least $OPT/2$, and $\mathcal{S}'^{short}_{unused}$, the remaining ones. It holds that

$$\begin{aligned} a(\mathcal{S}'^{tall}_{unused}) &\geq \left(\frac{1}{2} + 9\varepsilon\right)OPT \cdot W + a(\mathcal{V}) - a(\mathcal{S}'_{used}) - a(\mathcal{S}'^{short}_{unused}) \\ &\geq \left(\frac{1}{2} + 9\varepsilon\right)OPT \cdot W + a(\mathcal{V}) - \left(a(\mathcal{V}) + \frac{\left(w(\mathcal{S}'_{used}) + 1\right)}{2} OPT\right) \\ &\quad - \frac{OPT}{2} w(\mathcal{S}'^{short}_{unused}) \\ &= \left(\frac{1}{2} + 9\varepsilon\right)OPT \cdot W - \frac{\left(w(\mathcal{S}'_{used}) + w(\mathcal{S}'^{short}_{unused})\right)}{2} OPT - \frac{OPT}{2} \\ &\geq 9\varepsilon OPT \cdot W - \frac{OPT}{2} \geq 8\varepsilon OPT \cdot W, \end{aligned}$$

The second last inequality follows from the fact that $\left(w(\mathcal{S}'_{used}) + w(\mathcal{S}'^{short}_{unused})\right) \leq W$. In the last inequality, we used the assumption $W \geq 1/\varepsilon$.

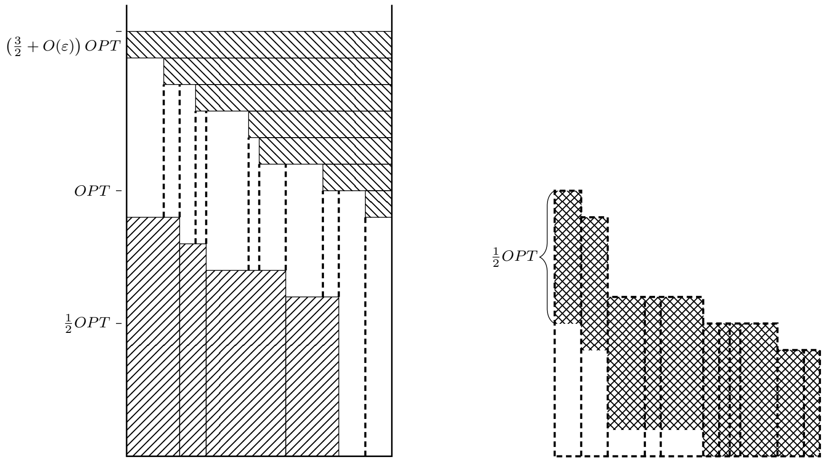

Fig. 2 Description of the packing of $\mathcal{V}_{vslice}$. **Left:** Packing of $\mathcal{T} \cup \mathcal{S}_{hslice}$ as described in Lemma 4 and the $O_\varepsilon(1)$ boxes for the free area defined by the dashed lines. **Right:** Boxes in the free area sorted by height. Even if we ignore $\frac{1}{2}OPT$ height from each box we have enough space for $\mathcal{V}_{vslice}$ and even to reserve space for future discarded vertical rectangles.

Since $a(\mathcal{S}'^{tall}_{unused}) \leq (\frac{3OPT}{2} + 11\varepsilon OPT)w(\mathcal{S}'^{tall}_{unused})$, it follows that, for $\varepsilon \leq 1/22$,

$$w(\mathcal{S}'^{tall}_{unused}) \geq \frac{8\varepsilon \cdot W}{\frac{3}{2} + 11\varepsilon} \geq 4\varepsilon \cdot W.$$

Next, consider the set of boxes spanned by $\mathcal{S}'^{tall}_{unused}$. All these boxes contain a free rectangular region of height $\frac{1}{2}OPT$ induced by the bottom part of $\mathcal{S}'^{tall}_{unused}$: let us call these regions $F_1, \ldots, F_k$. Since the number of these regions is at most $\frac{2}{\varepsilon}$ (i.e. the total number of boxes) and their total width is at least $4\varepsilon \cdot W$, by an averaging argument there exists one such F_i of width at least $2\varepsilon^2 W$.

3.3 Rounding

In this section, we show how to round the packing from Lemma 5 by means of Lemma 3, hence concluding the proof of Theorem 2.

Proof (Proof of Theorem 2) We start with the packing of $\mathcal{T} \cup \mathcal{S}_{hslice} \cup \mathcal{V}_{vslice}$ obtained from Lemma 5. Recall that $\mathcal{S}_{hslice}$ is packed into $1/\varepsilon + 1$ boxes $\mathcal{B}_S$ and $\mathcal{V}_{vslice}$ into $2/\varepsilon$ boxes $\mathcal{B}_V$. The total height of this packing is $(\frac{3}{2} + 15\varepsilon)OPT$, and this packing leaves a free region F of size $2\varepsilon^2 W \times \frac{1}{2}OPT$. Provided that δ is small enough, we can apply Lemma 3 to $(\mathcal{S}_{hslice}, \mathcal{B}_S)$ and obtain a packing of $\mathcal{S}$ into a set of containers fully contained in $\mathcal{B}_S$, plus two containers of size at most $W \times \frac{\varepsilon^2}{2}OPT$ each. We place the latter two containers on top of the packing, hence increasing the total height by $\varepsilon^2 OPT$. By applying the same Lemma to $(\mathcal{V}_{vslice}, \mathcal{B}_V)$, we obtain a packing of $\mathcal{V}$ into a set of containers fully contained in $\mathcal{B}_V$ plus two containers of size at most $\frac{\varepsilon^2}{2}W \times \frac{1}{2}OPT$ each. The latter two containers can be placed inside F without further increasing the height of the packing, still leaving a free region of size $\varepsilon^2 W \times \frac{1}{2}OPT$.

By construction, the number of used containers is $O_\varepsilon(1)$ and their sizes belong to a set that can be computed in polynomial time.

4 Algorithm

In this section, we describe an algorithm based on Theorem 2 to compute our final solution.

Consider the set of containers guaranteed by Theorem 2. In polynomial time we can guess such containers by trying all possibilities. By brute force we can also compute (in polynomial time) a packing of these containers plus the free box in the strip of total height at most $(\frac{3}{2} + O(\varepsilon))OPT$. We guess which ones among the vertical containers contain $\mathcal{T}$, and pack the whole set $\mathcal{T}$ there greedily in non-increasing order of height.

We next apply Lemma 2 with parameter $\varepsilon' = \varepsilon^3$ to the remaining containers and to the remaining rectangles $\mathcal{V} \cup \mathcal{S}$. This way we can pack a set $\mathcal{R}' \subseteq \mathcal{V} \cup \mathcal{S}$ of area at least $a(\mathcal{V} \cup \mathcal{S})(1 - \varepsilon^3)$. It remains to pack $\mathcal{R}'' := (\mathcal{V} \cup \mathcal{S}) \setminus \mathcal{R}'$, which satisfies that $a(\mathcal{R}'') \leq \varepsilon^3 a(\mathcal{V} \cup \mathcal{S}) \leq \varepsilon^3(\frac{3}{2} + O(\varepsilon))OPT \cdot W \leq 2\varepsilon^3 OPT \cdot W$. We partition $\mathcal{R}''$ into 3 subsets and pack them as follows:

1. The rectangles $\mathcal{V}'' \subseteq \mathcal{R}''$ of height at least $2\varepsilon OPT$ (notice that they have height at most $OPT/2$). By an area argument their total width is at most $\frac{2\varepsilon^3 OPT \cdot W}{2\varepsilon OPT} = \varepsilon^2 W$. Hence they fit in a vertical container of size $\varepsilon^2 W \times \frac{1}{2} OPT$ that can be placed in the area occupied by the free box (without increasing the height of the packing).
2. The rectangles $\mathcal{H}'' \subseteq \mathcal{R}''$ of width at least $\varepsilon^2 W$. By a similar area argument their total height is at most $\frac{2\varepsilon^3 OPT \cdot W}{\varepsilon^2 W} = 2\varepsilon OPT$. Hence they can be placed into an horizontal container of size $W \times 2\varepsilon OPT$ that can be placed on top of the current packing.
3. The remaining rectangles $\mathcal{S}'' \subseteq \mathcal{R}''$ with height at most $2\varepsilon OPT$ and width at most $\varepsilon^2 W$. By Lemma 1 with parameter $\varepsilon' = \sqrt{\varepsilon}$ and for small enough ε, we can pack $\mathcal{S}''$ by means of NFDH into an area container of size $\varepsilon\sqrt{\varepsilon} W \times 2\sqrt{\varepsilon} OPT$ to be placed on top of the current packing.

We now have all the ingredients to prove our main theorem.

Proof (Proof of Theorem 1) Consider the above algorithm which clearly has polynomial running time for any fixed parameter $\varepsilon > 0$. For δ small enough, it generates a feasible packing of all rectangles of total height at most $\left(\frac{3}{2} + O(\sqrt{\varepsilon})\right) OPT$. Considering the initial rounding of OPT by a factor $(1 + \varepsilon)$, this gives a $\frac{3}{2} + O(\sqrt{\varepsilon})$ approximation. The claim then follows by choosing ε appropriately.

5 Hardness of Approximation

In this section, we prove that the lower bound of $\frac{3}{2}$ on the approximability of Strip Packing still holds in the case of δ-skewed instances.

Lemma 6 *For any $\delta > 0$ and $\varepsilon > 0$, there is no polynomial-time $\left(\frac{3}{2} - \varepsilon\right)$-approximation for δ-skewed Strip Packing unless P=NP.*

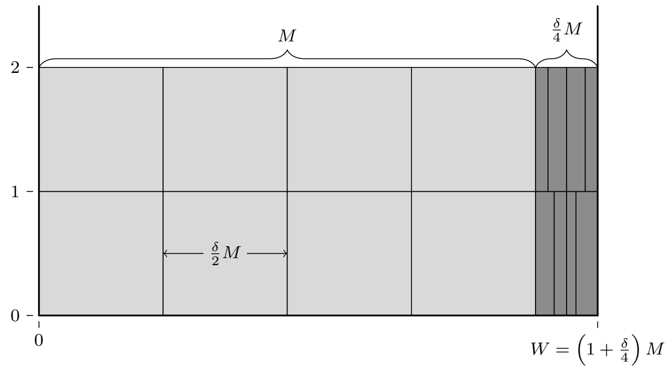

Fig. 3 Construction from Lemma 6. Light gray rectangles represent dummy rectangles and dark gray rectangles represent partition rectangles.

Proof We will prove this result via a reduction from the NP-complete *Partition* problem. Recall that in Partition we are given a set of integers $\mathcal{I} = \{x_1, \ldots, x_n\}$ whose sum is p. Our goal is to determine whether $\mathcal{I}$ can be partitioned into two sets $\mathcal{I}_1$ and $\mathcal{I}_2$ such that $\sum_{x_i \in \mathcal{I}_1} x_i = \frac{p}{2}$. We define our Strip Packing instance as follows: The width of the strip will be $W = (1+\delta/4)M$ where $M = \frac{2p}{\delta}$. Also, we will have $n + \frac{4}{\delta}$ rectangles in the instance, from which $\frac{4}{\delta}$ will have height 1 and width $\frac{\delta}{2}M$ (*dummy rectangles*), and the remaining n rectangles will have, for each $i = 1, \ldots, n$, height 1 and width x_i (*partition rectangles*). Notice that the instance is indeed δ-skewed as the width of the rectangles is either $\frac{\delta}{2}M \le \frac{\delta}{2}W$ or at most $p = \frac{\delta}{2}M \le \frac{\delta}{2}W$. Notice also that $OPT \ge 2$ since the area of the rectangles is $2W$.

We will now prove that the Partition instance is a YES instance if and only if $OPT = 2$. Since all the heights in the instance are 1, as a consequence a NO instance has height at least 3, hence concluding the proof of the claim. Notice that if the Partition instance is a YES instance then we can pack one next to the other $\frac{2}{\delta}$ dummy rectangles plus one side of the partition since their total width would be $M + \frac{p}{2} = \left(1+\frac{\delta}{4}\right)M$. We then analogously pack the rest of the rectangles on top, obtaining a packing of height 2 which is optimal as the total area of the rectangles is $2W$ (see Figure 3). On the other hand, if the optimal height of the Strip Packing instance is 2, the subregion $[0, W] \times [0, 2]$ in the strip must be fully occupied by rectangles. This actually implies that the horizontal segment $[0, W] \times \{1\}$ does not intersect the interior of any rectangle in the packing: indeed, if it is not the case, the space below that rectangle could not be occupied by any other rectangle (as heights are all 1). This divides the solution into two rows of height 1 and width W which are completely filled with rectangles. The only way to divide dummy rectangles into the two rows is to have exactly $\frac{2}{\delta}$ in each row (as the largest total width below W that they can sum up to is M and their total width is $2M$), hence the remaining partition rectangles in each row have total width exactly $\frac{p}{2}$, forming then a feasible solution to the Partition instance.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Adamaszek, A., Kociumaka, T., Pilipczuk, M., Pilipczuk, M.: Hardness of approximation for strip packing. ACM Transactions on Computation Theory **9**(3), 14:1–14:7 (2017). DOI 10.1145/3092026
2. Adamaszek, A., Wiese, A.: Approximation schemes for maximum weight independent set of rectangles. In: 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 400–409. IEEE Computer Society (2013). DOI 10.1109/FOCS.2013.50
3. Adamaszek, A., Wiese, A.: A quasi-ptas for the two-dimensional geometric knapsack problem. In: Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1491–1505. SIAM (2015). DOI 10.1137/1.9781611973730.98
4. Baker, B.S., Jr., E.G.C., Rivest, R.L.: Orthogonal packings in two dimensions. SIAM Journal on Computing **9**(4), 846–855 (1980). DOI 10.1137/0209064
5. Bansal, N., Caprara, A., Jansen, K., Prädel, L., Sviridenko, M.: A structural lemma in 2-dimensional packing, and its implications on approximability. In: Algorithms and Computation, 20th International Symposium (ISAAC), vol. 5878, pp. 77–86. Springer (2009). DOI 10.1007/978-3-642-10631-6_10
6. Bansal, N., Correa, J.R., Kenyon, C., Sviridenko, M.: Bin packing in multiple dimensions: Inapproximability results and approximation schemes. Mathematics of Operations Research **31**(1), 31–49 (2006). DOI 10.1287/moor.1050.0168
7. Bansal, N., Khan, A.: Improved approximation algorithm for two-dimensional bin packing. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 13–25. SIAM (2014). DOI 10.1137/1.9781611973402.2
8. Chalermsook, P., Chuzhoy, J.: Maximum independent set of rectangles. In: Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 892–901. SIAM (2009). URL `http://dl.acm.org/citation.cfm?id=1496770.1496867`
9. Chan, T.M., Har-Peled, S.: Approximation algorithms for maximum independent set of pseudo-disks. Discrete & Computational Geometry **48**(2), 373–392 (2012). DOI 10.1007/s00454-012-9417-5
10. Christensen, H.I., Khan, A., Pokutta, S., Tetali, P.: Approximation and online algorithms for multidimensional bin packing: A survey. Computer Science Review **24**, 63–79 (2017). DOI 10.1016/j.cosrev.2016.12.001
11. Coffman, E.G., Bruno, J.L.: Computer and job-shop scheduling theory. Wiley New York (1976)
12. Coffman Jr., E.G., Csirik, J., Galambos, G., Martello, S., Vigo, D.: Bin Packing Approximation Algorithms: Survey and Classification, pp. 455–531. Springer New York (2013). DOI 10.1007/978-1-4419-7997-1_35
13. Coffman Jr., E.G., Garey, M.R., Johnson, D.S., Tarjan, R.E.: Performance bounds for level-oriented two-dimensional packing algorithms. SIAM Journal on Computing **9**(4), 808–826 (1980). DOI 10.1137/0209062
14. Epstein, L., van Stee, R.: This side up! ACM Transactions on Algorithms **2**(2), 228–243 (2006). DOI 10.1145/1150334.1150339
15. Garey, M.R., Johnson, D.S.: "strong" np-completeness results: Motivation, examples, and implications. Journal of the ACM **25**(3), 499–508 (1978). DOI 10.1145/322077.322090
16. Gálvez, W., Grandoni, F., Heydrich, S., Ingala, S., Khan, A., Wiese, A.: Approximating geometric knapsack via L-packings. In: 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS), pp. 260–271. IEEE Computer Society (2017). DOI 10.1109/FOCS.2017.32
17. Gálvez, W., Grandoni, F., Heydrich, S., Ingala, S., Khan, A., Wiese, A.: Approximating geometric knapsack via L-packings. CoRR **abs/1711.07710** (2017). Available at `http://arxiv.org/abs/1711.07710`
18. Gálvez, W., Grandoni, F., Ingala, S., Khan, A.: Improved pseudo-polynomial-time approximation for strip packing. In: 36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), vol. 65, pp. 9:1–9:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2016). DOI 10.4230/LIPIcs.FSTTCS.2016.9
19. Gálvez, W., Grandoni, F., Khan, A., Ramírez-Romero, D., Wiese, A.: Improved approximation algorithms for 2-dimensional knapsack: Packing into multiple l-shapes, spirals, and more. In: 37th International Symposium on Computational Geometry (SoCG) (2021). To Appear. Full version available at `https://arxiv.org/pdf/2103.10406.pdf`
20. Harren, R., Jansen, K., Prädel, L., van Stee, R.: A $(5/3 + \varepsilon)$-approximation for strip packing. Computational Geometry **47**(2), 248–267 (2014). DOI 10.1016/j.comgeo.2013.08.008

21. Harren, R., van Stee, R.: Improved absolute approximation ratios for two-dimensional packing problems. In: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop (APPROX/RANDOM), vol. 5687, pp. 177–189. Springer (2009). DOI 10.1007/978-3-642-03685-9_14
22. Henning, S., Jansen, K., Rau, M., Schmarje, L.: Complexity and inapproximability results for parallel task scheduling and strip packing. Theory of Computing Systems **64**(1), 120–140 (2020). DOI 10.1007/s00224-019-09910-6
23. Jansen, K., Prädel, L.: A new asymptotic approximation algorithm for 3-dimensional strip packing. In: 40th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM), vol. 8327, pp. 327–338. Springer (2014). DOI 10.1007/978-3-319-04298-5_29
24. Jansen, K., Rau, M.: Closing the gap for pseudo-polynomial strip packing. In: 27th Annual European Symposium on Algorithms (ESA), vol. 144, pp. 62:1–62:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2019). DOI 10.4230/LIPIcs.ESA.2019.62
25. Jansen, K., Rau, M.: Improved approximation for two dimensional strip packing with polynomial bounded width. Theoretical Computer Science **789**, 34–49 (2019). DOI 10.1016/j.tcs.2019.04.002
26. Jansen, K., Solis-Oba, R.: New approximability results for 2-dimensional packing problems. In: Mathematical Foundations of Computer Science, 32nd International Symposium (MFCS), vol. 4708, pp. 103–114. Springer (2007). DOI 10.1007/978-3-540-74456-6_11
27. Jansen, K., van Stee, R.: On strip packing with rotations. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC), pp. 755–761. ACM (2005). DOI 10.1145/1060590.1060702
28. Jansen, K., Zhang, G.: Maximizing the total profit of rectangles packed into a rectangle. Algorithmica **47**(3), 323–342 (2007). DOI 10.1007/s00453-006-0194-5
29. Karbasioun, M.M., Shaikhet, G., Kranakis, E., Lambadaris, I.: Power strip packing of malleable demands in smart grid. In: Proceedings of IEEE International Conference on Communications (ICC), pp. 4261–4265. IEEE (2013). DOI 10.1109/ICC.2013.6655233
30. Kenyon, C., Rémila, E.: A near-optimal solution to a two-dimensional cutting stock problem. Mathematics of Operations Research **25**(4), 645–656 (2000). DOI 10.1287/moor.25.4.645.12118
31. Khan, A.: Approximation algorithms for multidimensional bin packing. Ph.D. thesis, Georgia Institute of Technology, Atlanta, GA, USA (2016). URL `http://hdl.handle.net/1853/54371`
32. Khan, A., Maiti, A., Sharma, A., Wiese, A.: On guillotine separable packings for the two-dimensional geometric knapsack problem. In: 37th International Symposium on Computational Geometry (SoCG) (2021). To Appear. Full version available at `https://arxiv.org/pdf/2103.09735.pdf`
33. Khan, A., Pittu, M.R.: On Guillotine Separability of Squares and Rectangles. In: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020), vol. 176, pp. 47:1–47:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2020). DOI 10.4230/LIPIcs.APPROX/RANDOM.2020.47
34. Leung, J.Y., Tam, T.W., Wong, C.S., Young, G.H., Chin, F.Y.L.: Packing squares into a square. Journal on Parallel and Distributed Computing **10**(3), 271–275 (1990). DOI 10.1016/0743-7315(90)90019-L
35. Mitchell, J.S.B.: Approximating maximum independent set for rectangles in the plane (2021). Available at `https://arxiv.org/pdf/2101.00326.pdf`
36. Miyazawa, F.K., Wakabayashi, Y.: Packing problems with orthogonal rotations. In: Theoretical Informatics, 6th Latin American Symposium (LATIN), vol. 2976, pp. 359–368. Springer (2004). DOI 10.1007/978-3-540-24698-5_40
37. Nadiradze, G., Wiese, A.: On approximating strip packing with a better ratio than 3/2. In: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1491–1510. SIAM (2016). DOI 10.1137/1.9781611974331.ch102
38. Schiermeyer, I.: Reverse-fit: A 2-optimal algorithm for packing rectangles. In: Algorithms, Second Annual European Symposium (ESA), vol. 855, pp. 290–299. Springer (1994). DOI 10.1007/BFb0049416
39. Sleator, D.D.: A 2.5 times optimal algorithm for packing in two dimensions. Information Processing Letters **10**(1), 37–40 (1980). DOI 10.1016/0020-0190(80)90121-0
40. Steinberg, A.: A strip-packing algorithm with absolute performance bound 2. SIAM Journal on Computing **26**(2), 401–409 (1997). DOI 10.1137/S0097539793255801

41. Tang, S., Huang, Q., Li, X., Wu, D.: Smoothing the energy consumption: Peak demand reduction in smart grid. In: 32nd IEEE International Conference on Computer Communications (INFOCOM), pp. 1133–1141. IEEE (2013). DOI 10.1109/INFCOM.2013.6566904