A harmonic algorithm for the 3D strip packing problem

Citation for published version (APA):

DOI:
10.1137/070691607

Document status and date:
Published: 01/01/2013

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 06. Jul. 2021
A HARMONIC ALGORITHM FOR THE 3D STRIP PACKING PROBLEM

NIKHIL BANSAL†, XIN HAN‡, KAZUO IWAMA§, MAXIM SVIRIDENKO¶, AND GUOCHUAN ZHANG∥

Abstract. In the three-dimensional (3D) strip packing problem, we are given a set of 3D rectangular items and a 3D box $B$. The goal is to pack all the items in $B$ such that the height of the packing is minimized. We consider the most basic version of the problem, where the items must be packed with their edges parallel to the edges of $B$ and cannot be rotated. Building upon Caprara’s work for the two-dimensional (2D) bin packing problem, we obtain an algorithm that, given any $\epsilon > 0$, achieves an approximation of $T_\infty + \epsilon \approx 1.69103 + \epsilon$, where $T_\infty$ is the well-known number that occurs naturally in the context of bin packing. Our key idea is to establish a connection between bin packing solutions for an arbitrary instance $I$ and the strip packing solutions for the corresponding instance obtained from $I$ by applying the harmonic transformation to certain dimensions. Based on this connection, we also give a simple alternate proof of the $T_\infty + \epsilon$ approximation for 2D bin packing due to Caprara. In particular, we show how his result follows from a simple modification of the asymptotic approximation scheme for 2D strip packing due to Kenyon and Rémyila.

Key words. strip packing, bin packing, harmonic algorithm

AMS subject classifications. 68W40, 68W25, 68W27, 68Q25

DOI. 10.1137/070691607

1. Introduction. In the three-dimensional (3D) strip packing problem, we are given a set of 3D rectangular items specified by their depth, width, and height. The goal is to pack these items into a single 3D rectangular box of unlimited height such that the height of the packing is minimized. We consider the most basic variant of the problem, the so-called “orthogonal packing without rotations” variant, where the items are not allowed to be rotated and must be packed parallel to the edges of the box. In any feasible packing, items are not allowed to overlap. By scaling the dimensions appropriately, it can be assumed that the box has width and depth of 1 and that the items have size at most 1 in any dimension. The two-dimensional (2D) strip packing problem is defined similarly, and the goal is to find a minimum height packing of 2D items into a strip of unit width and unlimited height.

Strip packing and the closely related bin packing problems are some of the most fundamental combinatorial optimization problems and have been studied extensively both theoretically and practically. In the $d$-dimensional bin packing problem, we
are given a collection of $d$-dimensional rectangular items with size at most 1 in each dimension, and the goal is to pack these items into the minimum number of unit sized $d$-dimensional bins. We will mainly be interested in the case of $d = 1, 2, \text{ or } 3$.

The 3D strip packing is a common generalization of both the 2D bin packing problem and the 2D strip packing problem. When each item has height exactly 1, the 3D strip packing problem is identical to the 2D bin packing problem. Similarly, when the width (or depth) of each item is exactly 1, the 3D strip packing problem is identical to the 2D strip packing problem. The NP-hardness of 3D strip packing and 2D bin packing follows easily from the NP-hardness of the one-dimensional (1D) bin packing problem, and hence we will be interested in polynomial time approximation algorithms for these problems.

For packing problems, the worst case approximation ratio usually occurs only for specialized “small” instances, and hence the standard and the more appropriate measure used is the asymptotic approximation ratio. Given a polynomial time algorithm $A$, the asymptotic approximation ratio $R_A^\infty$ is defined by

$$R_A^\infty = \max \left\{ \frac{A(I)}{\text{Opt}(I)} \mid \text{Opt}(I) = n \right\} \quad \text{and} \quad R_A^\infty = \lim_{n \to \infty} \sup R_A^n,$$

where $I$ ranges over the set of all problem instances and $A(I)$ and $\text{Opt}(I)$ are the values of $A$ and the optimal algorithm, respectively, on $I$. A problem is said to admit an asymptotic polynomial time approximation scheme (APTAS) if, for every $\epsilon > 0$, there is a polynomial time algorithm $A_\epsilon$ with an asymptotic approximation ratio of $(1 + \epsilon)$.

Fernandez de la Vega and Lueker, in their celebrated work [9], gave the first APTAS for the 1D bin packing problem. Later, this was substantially improved by Karmarkar and Karp [13], who gave a polynomial time guarantee of $\text{Opt}(I) + O(\log^2 \text{Opt}(I))$. For the 2D strip packing problem, starting with an asymptotic approximation guarantee of 3 due to Baker, Coffman, and Rivest [1], the ratio was improved in a sequence of papers until the breakthrough work of Kenyon and Rémila [14], where they gave an APTAS for the problem. The Kenyon–Rémila (KR) algorithm of is based on elegant extensions of the ideas of [9] and will play a key role in this paper. Recently, Han, Iwama, and Zhang [10] established a general algorithmic framework relating bin packing and strip packing. They proved that any offline 1D bin packing algorithm can be applied to 2D strip packing maintaining the same asymptotic worst-case ratio and a class of harmonic-based online algorithms for 1D bin packing can be applied to online 2D strip packing maintaining the same asymptotic worst-case ratio.

On the negative side, Bansal et al. [4] showed that the 2D bin packing problem does not admit an APTAS unless $P = NP$. This directly implies that the 3D strip packing problem does not admit an APTAS either. Currently, the best known algorithm for the 3D strip packing problem is a relatively recent result, due to Jansen and Solis-Oba [12], which has an asymptotic approximation ratio of 2. This was a substantial improvement upon a long sequence of works that achieved approximation guarantees of 3.25 [15], 2.89 [16], and 2.67 [19] for the problem. However, better results are known for the special case of the 2D bin packing problem. Caprara [5, 6] gave the first algorithm which broke the barrier of 2 and, given any $\epsilon > 0$ as a parameter, achieved an asymptotic approximation ratio of $T_\infty + \epsilon \approx 1.69103 + \epsilon$. Here $T_\infty$ is the well-known constant that appears ubiquitously in the context of bin packing, whose technical definition we give in section 2. Very recently, the asymptotic approximation
ratio for the problem has been further improved to \(1 + \ln(T_\infty) \approx 1.52\) [2, 3].

In this paper, we give an algorithm for the 3D strip packing problem that, given any \(\epsilon > 0\) as a parameter, has performance guarantee within \(T_\infty + \epsilon\). In addition to improving the previously known guarantees, our algorithm is very natural and simple and is based on relating strip packing and bin packing in easy but nonobvious ways. These relations between strip packing and bin packing also give more intuitive and simpler proofs (in our opinion) of previously known results. In fact, we begin by giving an alternative view of Caprara’s \(T_\infty + \epsilon \approx 1.69103 + \epsilon\) approximation for 2D bin packing. Even though we are reproving a known result, we believe that this result should be interesting by itself. Moreover, this introduces the main ideas and techniques that will be needed for our result on 3D strip packing.

The paper is organized as follows. We begin by describing the preliminaries in section 2, where among other things we give an overview of the KR asymptotic approximation scheme for 2D strip packing. In section 3, we show how Caprara’s result for 2D bin packing follows by a variant of the KR algorithm applied to the modified instance obtained by applying the harmonic transformation to the item heights. Finally, in section 4, we build upon these ideas and describe and analyze our harmonic algorithm for the 3D strip packing problem.

2. Notation and preliminaries. We will often use bins with sizes different from 1. A bin of size \((h_x, h_y)\) is a rectangular box with \(x\)-dimension \(h_x\) and \(y\)-dimension \(h_y\). We use \((h_x, \infty)\) to denote a strip with unlimited height in the \(y\)-direction and \(x\)-dimension equal to \(h_x\). In two dimensions we refer to the \(x\)-dimension as width, and \(y\)-dimension as height. For 3D boxes and strips, we define the bin size \((h_x, h_y, h_z)\) analogously. Similarly, we use \((h_x, h_y, \infty)\) to denote a strip which is unlimited in the \(z\)-direction. In three dimensions, we refer to the \(x\)-direction as depth, the \(y\)-dimension as width, and the \(z\)-dimension as height. Note that our labeling of the axes in three dimensions is not the usual one and moreover inconsistent with the labeling for the 2D case. However, this unusual labeling will be convenient for our purposes later when we will relate 3D strip packing to 2D packing in various ways (we encourage the reader to always refer to Figure 3 for a pictorial view of our labeling of the axes in three dimensions, wherever there may be a cause for confusion). As usual, we will always assume that the dimensions of the bins involved and the item sizes are all rational numbers (and hence integral multiples of \(1/M\) for some arbitrarily large integer \(M\)).

Let \(\text{Opt}_{(h_x, h_y)}(I)\) denote the optimal number of bins with size \((h_x, h_y)\) required to pack \(I\). Thus \(\text{Opt}_{(1,1)}(I)\) is the optimal number of unit size bins required to pack \(I\). We also use \(\text{Opt}_{(h_x, \infty)}(I)\) to denote the optimal height (\(y\)-dimension) required to pack \(I\) in a strip of width \(h_x\). We use \(A_{(h_x, h_y)}(I)\) to denote the number of bins used by algorithm A to pack an instance \(I\), and use analogous notation for 2D strip packing, 3D strip packing, and 3D bin packing.

**Definition of \(T_\infty\).** Let \(t_1 = 1\) and \(t_{i+1} = t_i(t_i + 1)\) for \(i > 1\).

\[
T_\infty = \sum_{i=1}^{\infty} \frac{1}{t_i} = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \cdots \approx 1.69103\ldots,
\]

where sequence \(t_i + 1\) is known as Sylvester’s sequence [21] for \(i \geq 1\), which was first used for the bin packing problem in [11].

We now describe three well-known results which are key building blocks in most packing algorithms.
Harmonic transformation. This idea was first introduced by Lee and Lee [17]. Let $k$ be a positive integer and $x$ be a positive real in $(0,1]$. The harmonic transformation $f_k$ with parameter $k$ is defined as follows: $f_k(x) = 1/q$ if $x \in (1/(q+1), 1/q]$ for some integer $q = 1, \ldots, k-1$ and $f_k(x) = kx/(k-1)$ if $x \in (0,1/k]$. The crucial property of this transformation is the following.

**Lemma 2.1.** For any sequence $x_1, \ldots, x_n$ with $x_i \in (0,1]$ for $i = 1, \ldots, n$ and $\sum_{i=1}^{n} x_i \leq 1$, we have $\lim_{k \to \infty} \sum_{i=1}^{n} f_k(x_i) \leq T_{\infty}$.

For our purposes, we will use a slight variant of the transformation $f_k$ described above. We call this variant $h_k$. For $x \in (1/(q+1), 1/q]$, where $q = 1, \ldots, k-1$, we set $h_k(x) = 1/q$ as previously; however, for $x \in (0,1/k]$, we set $h_k(x) = x$. Since $h_k(x) \leq f_k(x)$ for all $x$, it easily follows that Lemma 2.1 holds for our variant too. In fact, observe that, for any $x$, the quantity $h_k(x)$ is a nondecreasing function of $k$. This follows from the fact that if $x \in (1/(q+1), 1/q]$, then $h_k(x) = 1/q$ for $k > q$ and $h_k(x) = x$ for $k \leq q$. This implies that for any $x_1, \ldots, x_n$ such that $\sum_{i=1}^{n} x_i \leq 1$, it holds that $\sum_i h_k(x_i) \leq \lim_{k \to \infty} \sum_i f_k(x_i) \leq \lim_{k \to \infty} \sum_i f_k(x_i) \leq T_{\infty}$. We use $G_k$ to denote the maximum of $\sum_i h_k(x_i)$ over all combinations of $n$ numbers $x_1, \ldots, x_n$ (i.e., for our variant). Thus the above discussion implies that $G_k \leq T_{\infty}$.

Next fit decreasing height (NFDH) algorithm. This idea was first introduced by Coffman et al. [7]. Suppose we are given a collection of rectangles that need to be packed in a 2D bin. The NFDH algorithm orders the rectangles in the decreasing order of their heights and packs them in sublevels, shown in Figure 1. The bottom of the first level coincides with the bottom of the strip (or the bottom of the first bin, if we are packing bins). Items are taken from the list and packed in the current level with their bottoms coinciding with the bottom of the level, and pushed as far to the left as possible, given the items already packed in the level. This continues until no items remain (in which case the packing is complete) or the current item is wider than the space remaining at the right side of the strip/bin. In this latter case the level is closed and a new level is created whose bottom coincides with the top of the leftmost (and hence tallest) item in the old level. This is a complete description of the algorithm in the strip packing case. If we are packing into bins, then we must consider what happens when the current item does not fit in the new level because it is taller than the gap between the bottom of the new level and the top of the bin. In this case we close the bin and the new level is taken to coincide with the bottom of the next bin.

Let $w$ and $h$ be the maximal width and height of a set of rectangles. If $w$ and $h$ are relatively small compared to the dimensions of the bin, then the packing produced
by NFDH is very efficient.

**Lemma 2.2 (Coffman et al. [7]).** Given a set of rectangles with width at most \( w \) and height at most \( h \), if NFDH is used to pack these items in a bin of width \( a \) and height \( b \), then the total used volume in that bin is at least \((a - w)(b - h)\) (provided there are enough rectangles that NFDH never runs out of them).

APTAS of Kenyon and Rémiola for 2D strip packing [14]. Given a collection of rectangles \( I \), let \( \text{Opt}(I) \) denote the height of some optimal strip packing of \( I \). The KR algorithm takes as input a parameter \( \epsilon > 0 \) and produces a packing with height \((1 + \epsilon)\text{Opt}(I) + O(1/\epsilon^2)\). More precisely, the algorithm considers a fractional linear programming relaxation of the problem, which can be viewed as a packing where each item can be arbitrarily sliced horizontally into a collection of items with the same width and the same total height, and the pieces can be packed arbitrarily. Let \( \text{LP}(I) \) denote the optimal value of this relaxation. The algorithm of [14] actually produces a packing of height \((1 + \epsilon)\text{LP}(I) + O(1/\epsilon^2)\). We now sketch the algorithm and describe the main ideas.

Given a parameter \( \epsilon > 0 \), the algorithm begins by classifying items with width greater than \( \epsilon/(1 + \epsilon) \) as wide, and thin otherwise. It then rounds up the widths of wide items into \( \ell = O(1/\epsilon^2) \) different values \( w_1, \ldots, w_\ell \), in such a way that the optimal value of the fractional relaxation of this rounded instance is at most \((1 + \epsilon)\) times that of the original instance.\footnote{For our purposes, we do not care about the details of this rounding procedure. However, it can be done by sorting the wide items in increasing order of their widths, and forming \( \ell \) groups of consecutive items such that the cumulative height of items in each group is almost the same. For each item, round its width to that of the widest item in its group.} For the rounded instance, the linear program (LP) restricted to the wide items has a very nice form. Let a configuration denote some positive integer linear combination of these rounded widths that sums up to at most 1. Thus, a configuration is some valid way of placing the wide items along the width of the bin. Let \( H_i \) denote the cumulative height of items of width \( w_i \) in the instance. There is a variable for each configuration which represents the height of this configuration in the solution. The goal is to choose configurations with minimum total height such that for each width \( w_i \), exactly \( H_i \) height is allocated by these configurations. As there are \( \ell \) constraints, there is a basic optimal solution to the LP where only \( \ell = O(1/\epsilon^2) \) configurations are assigned a nonzero value.

To round this solution, consider each configuration sequentially and start placing the items of appropriate width as determined by this configuration. The only place some item may not fit completely is when a configuration finishes. In this case we take these items that fit partially and place them separately on a horizontal line. This adds at most one unit of extra height. Since there are at most \( \ell = O(1/\epsilon^2) \) relevant configurations, this adds at most \( O(1/\epsilon^2) \) to the height of the packing. Finally, it remains to pack the thin items. To do this, the configurations are pushed to the left side of the strip, leaving \( O(1/\epsilon^2) \) empty rectangular regions to the right. The thin items are then packed in these regions using the NFDH algorithm. Figure 2 (on the left) shows an example of the configuration LP solution and these regions. If all thin items do not fit in these regions, the remainder are packed on the top using NFDH. Note that if additional height is used to pack thin items, then the packing must essentially waste less than \( \epsilon \) fraction of the area.

Kenyon and Rémiola [14] show how to fill details in the above description to obtain an approximation scheme. Moreover, as pointed out in [14], by using the techniques in [20], the algorithm can be implemented to run in time \( O(n \log n + \log^3 n \epsilon^{-6} \log^3 (\epsilon^{-1})) \).
3. Two-dimensional bin packing. The following is a well-known technique to obtain a $(2 + \epsilon)$ approximation for 2D bin packing using the APTAS for 2D strip packing. Given an input instance $I$, find an almost optimal strip packing $A_{(1,\infty)}(I) \approx (1 + \epsilon/2)\text{Opt}_{(1,\infty)}(I)$ using the KR algorithm. Cut this strip into slices of height 1 using the horizontal lines $y = i$ for $i = 1, \ldots, \lfloor A_{(1,\infty)}(I) \rfloor$. For each such $i$, remove the rectangles that intersect the line $y = i$ and pack them into a separate bin. This gives a feasible bin packing using at most $2A_{(1,\infty)}(I) \approx (2+\epsilon)\text{Opt}_{(1,\infty)}(I) \leq (2+\epsilon)\text{Opt}_{(1,1)}(I)$ bins.

Our approach for 2D bin packing is based on a natural refinement of this idea: Suppose we can produce a strip packing of $\tilde{I}$ with the property that any item that is cut by the horizontal line $y = i$ has height at most $\epsilon$. We call this the tall not sliced property. In this case we could take up to $1/\epsilon$ slices consisting of the items cut by such horizontal lines and pack them in a single bin. This would yield a bin packing where the number of bins used is at most $(1 + \epsilon)$ times the height of the strip. Of course, in general one cannot guarantee an almost optimal strip packing which has the tall not sliced property mentioned above. Indeed, if possible, it would imply an APTAS for the 2D bin packing problem which is impossible unless $P = NP$ [4]. However, as we show, it turns out that if all the item heights are the reciprocals of integers, then one can, with only a small penalty, create a packing for which the tall not sliced property does hold. In particular, if all the items with height more than $\epsilon$ have heights equal to $1/q$ for some integer $q$, then a simple modification of the KR algorithm can be used to find an almost optimal strip packing that additionally satisfies the tall not sliced property. Thus our overall algorithm is to convert an arbitrary instance into one which has inverse integral heights (for large height items) and then find an almost optimal strip packing of this instance that also satisfies the tall not sliced property. The approximation factor $T_\infty \approx 1.69103$ is incurred only in the process of converting an arbitrary instance into one where the heights are harmonic.

We now describe the details. Let $\epsilon > 0$ be a precision parameter, and let us assume without loss of generality that $1/\epsilon$ is an integer. Given an input instance $I$, we derive an input instance $\tilde{I}$ by applying the previously defined transformation $h_k$, with $k = 1/\epsilon$, to the height of each item in $I$. The following straightforward lemma relates $\tilde{I}$ to $I$.

**Lemma 3.1.** The number of bins of size $(1, T_\infty)$ required to pack $\tilde{I}$ is no more than the number of unit size bins required to pack $I$. That is, $\text{Opt}_{(1,T_\infty)}(\tilde{I}) \leq \text{Opt}_{(1,1)}(I)$.

**Proof.** Fix some optimal solution $\text{Opt}_{(1,1)}(I)$ and use same packing for corresponding items in $\tilde{I}$. By Lemma 2.1, if a subset of items in $I$ fit in a $(1,1)$ bin, then the corresponding items in $\tilde{I}$ also fit in a bin of size $(1, G_k)$. This gives a feasible packing of $\tilde{I}$ using $\text{Opt}_{(1,1)}(I)$ bins of size $(1, G_k)$. As $G_k \leq T_\infty$ for our variant of the harmonic rounding, the result follows. \qed

We will be interested in the strip packing of $\tilde{I}$. The following lemma shows that there is almost optimal packing of $\tilde{I}$ which additionally satisfies the tall not sliced property.

**Lemma 3.2.** Suppose $\epsilon > 0$ and we are given a 2D instance $\tilde{I}$, where all item heights that are greater than $\epsilon$ are of the form $1/q$, where $q$ is an integer. Let $LP(\tilde{I})$ denote the value of the KR fractional relaxation for this instance. Then, a slight modification of the KR algorithm produces a strip packing that satisfies the tall not sliced property and has height at most $(1 + \epsilon)LP(\tilde{I}) + O(1/\epsilon^2)$.

**Proof.** Consider the KR algorithm described in section 2. We will form the configuration LP exactly as in the KR algorithm. The only difference will be in...
the items denoted with $U$ have height 1, those with $H$ have height $1/2$, those with $T$ have height $1/3$, and those with $F$ have height $1/4$. Some items such as $T_3$ and $H_5$ are packed fractionally. Note that on the right, we continue placing items for each width type in a configuration in the decreasing height order. If a particular height for some width type in a configuration is over, we move to the next integer starting position. For example, in the second width type in the first configuration, we leave empty space after placing $H_3$ and start $T_3$ from height $y=2$.

To round the solution consider the configurations an arbitrary order. In each configuration start placing the items of appropriate width (as dictated by this configuration) in decreasing height order. While placing "tall" items of height $1/q$, we ensure the property that the starting $y$-position of each such item is an integral multiple of $1/q$. In fact, for simplicity we will assume that whenever we start placing items of a different (new) height in a configuration, we start these items from the next $y$-position. For short items (with height smaller than $\epsilon$), we do not care where they start. Figure 2 shows an example of this transformation. In this rounding, we lose some space only when we change configurations or shift from one harmonic height class to another for some width type. Since there are $O(1/\epsilon^2)$ different widths for wide items (and therefore this many different configurations to consider), and $O(1/\epsilon)$ dif-

![Diagram](image_url)
different possible large heights for each of these widths, this can happen at most \(O(1/\epsilon^3)\) times and hence add at most \(O(1/\epsilon^3)\) to the total height used.

It now remains to pack thin items. As in the KR algorithm, the wide items are pushed to the left, leaving \(l = O(1/\epsilon^2)\) empty rectangular regions on the right for thin items which are packed using NFDH, as adapted in the obvious way to handle the variable height and width bins represented by the regions. As before, we ensure that tall items of height \(1/q\) start at \(y\)-position equal to an integral multiple of \(1/q\). We do it by starting a new shelf each time the current item has height in a different class from the previous item, and by ensuring that this shelf starts at some integral multiple of the item size. We do not care where the short items start. Again this adds at most \(O(1/\epsilon)\) to the height, since the height changes \(O(1/\epsilon)\) times and configurations change at most \(l\) times.

By construction, the packing produced satisfies the tall not sliced property. Moreover, the height of the packing produced above is at most \(O(1/\epsilon^3)\) more than the height produced by the KR algorithm.

**Final algorithm.** The final algorithm works as follows. Given an instance \(I\), we round up the heights of items using the harmonic transformation and obtain the instance \(\tilde{I}\). After that, we apply the modification of the KR algorithm in Lemma 3.2 to \(\tilde{I}\) to obtain a strip packing \(P\). Since \(P\) satisfies the tall not sliced property, we convert it into a feasible 2D bin packing solution while losing a factor of at most \((1 + \epsilon)\). Since \(\tilde{I}\) was obtained from \(I\) by rounding each item up, this also produces a feasible packing of \(I\).

**Theorem 3.3.** Let \(A_{(1,1)}(I)\) denote the number of bins used by our algorithm. Then \(A_{(1,1)}(I) \leq (1 + \epsilon^2)T_\infty \cdot \text{Opt}_{(1,1)}(I) + O(1/\epsilon^3)\).

**Proof.** As each item in \(I\) is not larger than the corresponding item in \(\tilde{I}\), it follows that \(A_{(1,1)}(I) \leq A_{(1,1)}(\tilde{I})\). Since the packing produced by Lemma 3.2 has the tall not sliced property, \(A_{(1,1)}(\tilde{I})\) (the solution produced by slicing the strip by the lines \(y = 1, y = 2, \ldots\)) is at most \((1 + \epsilon)\) times the height \(A_{(1,\infty)}(\tilde{I})\). Therefore, Lemma 3.2 implies that

\[
(3.1) \quad A_{(1,1)}(I) \leq A_{(1,1)}(\tilde{I}) \leq (1 + \epsilon^2)\text{Opt}_{(1,\infty)}(\tilde{I}) + O(1/\epsilon^3).
\]

Consider the optimal bin packing solution \(\text{Opt}_{(1,T_\infty)}(\tilde{I})\) of \(\tilde{I}\) into bins of width 1 and height \(T_\infty\). Since placing these bins on top of each other yields a feasible strip packing solution for \(\tilde{I}\), it trivially follows that

\[
(3.2) \quad \text{Opt}_{(1,\infty)}(\tilde{I}) \leq T_\infty \cdot \text{Opt}_{(1,T_\infty)}(\tilde{I}).
\]

By Lemma 3.1, we have \(\text{Opt}_{(1,T_\infty)}(\tilde{I}) \leq \text{Opt}_{(1,1)}(I)\), which together with (3.1) and (3.2) gives our desired result.

**4. The algorithm for the 3D strip packing.** Let \(I\) denote the input instance consisting of 3D items \((x_i, y_i, z_i)\), for \(i = 1, \ldots, n\). We will assume that the strip where these items needs to be packed is infinite in the \(z\)-direction. Thus, in our notation we wish to find an approximation to \(\text{Opt}_{(1,1,\infty)}(I)\). We will use an approach similar to that in the previous section of applying the harmonic transformation to one dimension and then relating bin packing and strip packing on this rounded instance. We give an overview below.

**4.1. Overview.** Given an accuracy parameter \(\epsilon > 0\), we will relate the 3D strip packing problem to the 2D version in various steps, where we possibly lose a factor \((1 + \epsilon)\) and an additive term \(O_\epsilon(1)\) at each step.
Let $c = \lceil 1/\epsilon \rceil$. First, we observe that packing in a 3D strip $(1, 1, \infty)$ is essentially the same as packing in bins of size $(1, 1, c)$. In particular, slicing the strip packing of height $H$ at $z$-coordinate equal to integral multiples of $c$ and repacking the sliced items gives a packing into bins of size $(1, 1, c)$ using at most $(1 + \epsilon)H/c$ bins (see Lemma 4.1 for details). Thus it suffices to approximate $\text{Opt}_{(1,1,c)}(I)$.

To approximate $\text{Opt}_{(1,1,c)}(I)$, we work with the strip $(1, \infty, c)$ which is infinite in the $y$-direction (see Figure 3 for a pictorial view of this idea). As in section 3, we show that if the width of items are harmonic, then we can find an almost optimal packing of $(1, \infty, c)$ with the tall not sliced property replaced by the analogous wide not sliced property. That is, we do not cut items of width more than $\epsilon$, when we slice the strip along the $y$-axis by the hyperplanes $y = 1, y = 2, \ldots$. Since we lose the $T_\infty$ ratio in the harmonic transformation, and a packing in the strip $(1, \infty, c)$ with the wide not sliced property can be converted to a packing in bins $(1, 1, c)$ losing a factor of at most $(1 + \epsilon)$, this allows us to obtain an approximation ratio approaching $T_\infty$.

However, as the strip $(1, \infty, c)$ is 3D, we cannot directly apply the results of section 3 for 2D strips. To get around this issue, our idea is to first group the 3D items (appropriately) into slabs of size $(x, y, c)$, where $x$ is arbitrary and $y$ is harmonic. As the size of a slab in the $z$-dimension is $c$, the problem of packing slabs can be viewed as a 2D strip packing problem with harmonic widths and we can use the results from section 3.

In general, forming the slabs in advance and allowing the algorithm to only work with them can be overly restrictive and lead to a very suboptimal packing. So the main part of the algorithm and the proof is to show how to form the slabs appropriately, and argue that the optimal slab packing for them is not much worse than an optimal packing of items. We now describe the details.

**4.2. Preliminaries.** We first state some simple results that will be useful later.

**Lemma 4.1.** For any instance $I$, we have $\text{Opt}_{(1,1,\infty)}(I)/c \leq \text{Opt}_{(1,1,c)}(I) \leq (1 + 1/c)\text{Opt}_{(1,1,\infty)}(I)/c + 2$.

*Proof.* The first inequality is clear since, given any packing in bins of size $(1, 1, c)$, we can place the bins on top of each other to produce a valid 3D strip packing solution.

For the second inequality, consider some optimal solution for the 3D strip packing problem. Make cuts in the $z$-dimension at the integral multiples of $c$. These cuts define...
Opt_{1,1,\infty}(I)/c + 1 bins. Pack items that are cut in bins of size (1,1,c). Since each item has height at most 1, the number of additional bins of size (1,1,c) required is upper bounded by the number of cuts divided by c (rounded up), which is at most Opt_{1,1,\infty}(I)/c^2 + 1. Hence the total number of bins is at most (1+1/c)Opt_{1,1,\infty}/c + 2.

Let $k = \lceil 1/\epsilon \rceil + 2$. Given an instance $I$ of 3D items, let $\tilde{I}$ denote the instance obtained by applying transformation $h_\kappa$ with $\kappa = k^2$ to the item widths (i.e., the $y$-dimension). The following simple lemma relates $I$ and $\tilde{I}$.

**Lemma 4.2.** If some items in $I$ can be packed in a bin of size $(1,1,c)$, then the corresponding items in $\tilde{I}$ can be packed in a bin of size $(1,T_\infty,c)$. In particular, this implies that $\text{Opt}_{1,1,\infty}(\tilde{I}) \leq \text{Opt}_{1,1,\infty}(I)$.

**Proof.** By an argument identical to the one in Lemma 3.1, each solution to the original instance and bins of size $(1,1,c)$ is a feasible solution for the rounded instance and bins of size $(1,T_\infty,c)$.

The following observation is obvious, but it will be useful in relating 3D and 2D strip packing.

**Observation 4.3.** Let $K$ be a collection of 3D items, each of height $\delta$. Let $\tilde{K}$ denote the corresponding 2D instance obtained by ignoring the height, i.e., an item $(x,y,\delta) \in K$ is replaced by item $(x,y) \in \tilde{K}$. Then the problem of packing $K$ into the strip $(1,\infty,\delta)$ is equivalent to that of packing $\tilde{K}$ into $(1,\infty)$.

### 4.3. Slab formation procedure

Let $\tilde{I}$ denote the rounded instance as defined above. We now focus on the problem of packing $\tilde{I}$ into the strip $(1,\infty,c)$, and we show how to reduce this to a 2D strip packing problem by packing the items into slabs in advance. Here, a slab refers to any 3D box which has height $c$ (but arbitrary width and depth).

The items in $\tilde{I}$ are naturally partitioned into $k^2$ classes according to their rounded $y$-dimension. For $q = 1, \ldots, k^2 - 1$, let $S_q$ be the set of items with width $\tilde{y}_i = 1/q$, and let $S_{k^2}$ be the set of items with $\tilde{y}_i \in (0,1/k^2]$. Within a set $S_q$, we order the items in the decreasing order of their $x$-dimension. We now describe how to group the items in $\tilde{I}$ into slabs.

**Forming slabs for items in $S_q$ for $q = 1, \ldots, k^2 - 1$.** Choose a maximal prefix of items according to their indices (i.e., in nonincreasing depth) with total height at most $c$. Place these items in a slab $(x_1,1/q,c)$, where $x_1$ is the maximal depth among items currently in $S_q$. Delete these items from $S_q$ and continue the process. Let $G_q(1), \ldots, G_q(n_q)$ denote the slabs produced in this order.

**Forming slabs for items in $S_{k^2}$.** For items in $S_{k^2}$, the groups are slightly more involved since the items may have different widths. To handle this, we will pack these items in slabs of width $1/k$ and height $c$. Since the items in $S_{k^2}$ have width less than $1/k^2$ (which is much less than the width of the slab), we will be able to argue that the packing in slabs is “efficient.”

To form the slabs, choose a maximal prefix of the items in $S_{k^2}$ with total $yz$-area of items at most $(1/k - 1/k^2)(c - 1)$. Here, the $yz$-area of an item $i$ of size $(x_i, y_i, z_i)$ is the quantity $y_i z_i$. Place these items in a slab of dimensions $(x_1, 1/k,c)$ in the NFDH order (where $x_1$ is the maximal depth among items currently in $S_{k^2}$). This is always possible by Lemma 2.2. Delete these items from $S_{k^2}$ and continue the process. Let $G_q(1), \ldots, G_q(n_q)$ denote the slabs produced in this order.

**Remark.** Note that for $q = k^2$, the items within a slab are not necessarily packed in the order of nonincreasing depth (even though the prefix considers them in nonincreasing depth), but in the order of decreasing height.
The following observation shows that each slab (except for at most \( k^2 = O(1) \) many) is packed quite tightly.

**Observation 4.4.**

1. For \( q = 1, \ldots, k^2 - 1 \), each of the slabs produced from \( S_q \) (except possibly for the last one) has height at least \( c - 1 \).
2. For \( q = k^2 \), for each of the slabs produced (except possibly for the last one), the total \( yz \)-area of the items in the slab is at least \((1/k - 1/k^2)(c - 1) - 1/k^2 > (1/k - 2/k^2)(c - 1)\). Note that the \( 1/k^2 \) term is subtracted from \((1/k - 1/k^2)(c - 1)\) as the \( yz \)-area of any item in \( S_{k^2} \) is at most \( 1/k^2 \).

Clearly, any packing of slabs into \((1, \infty, c)\) also gives a valid packing of \( \tilde{I} \). We note here the following simple observation which will allow us to show a partial converse later.

**Observation 4.5.** Let \( I_q(i) \) denote the set of items of \( \tilde{I} \) that are packed in the slab \( G_q(i) \). Then for each \( q \) and \( i = 1, \ldots, n_q - 1 \) it holds that the depth of every item in \( I_q(i) \) is more than the depth of every item in \( I_q(i + 1) \). This follows as the items are considered in nonincreasing depth during the slab formation.

Let \( J \) denote the instance consisting of these slabs. Given this slab formation, we define a 2D strip packing problem \( J \) as follows. For each slab of size \((x, y, c)\) in \( \tilde{I} \) define an item of size \((x, y)\). Observe that the width of each item in the instance \( J \) is the reciprocal of an integer; it is either \( 1/q \) for \( q = 1, \ldots, k^2 - 1 \) corresponding to slabs of items in \( S_q \), or \( 1/k \) for slabs of items in \( S_{k^2} \). Thus we can use Lemma 3.2 to produce an almost optimal 2D strip packing solution of \( J \) that also has the *wide not sliced* property (and hence it can be cut along the hyperplanes \( y = 1, y = 2, \ldots \) without cutting any item).

### 4.4. The 3D strip packing algorithm.

We can now state our 3D strip packing algorithm.

- Given the instance \( I \), apply the harmonic transformation to the widths of items in \( I \) to obtain \( \tilde{I} \). Form slabs of items in \( \tilde{I} \) as described above to form the 2D strip packing instance \( J \).
- Apply the modified KR algorithm in Lemma 3.2 to \( J \), which gives a packing of slabs into the strip \((1, \infty, c)\), where \( c = \lceil 1/\epsilon \rceil \). Cut the packing in this strip by hyperplanes \( y = 1, y = 2, \ldots \) to obtain a packing of items in \( \tilde{I} \) into bins of size \((1, 1, c)\).
- Place these bins of size \((1, 1, c)\) on top of each other in \( z \)-dimension to produce a feasible 3D strip packing into \((1, 1, \infty)\).

We note that the running time of the algorithm is bounded by that of the KR algorithm.

### 4.5. Analysis.

Let \( A_{(1, \infty)}(I) \) denote the height of the 3D strip packing produced by our algorithm. Let \( KR_{(1, \infty)}(J) \) denote the value produced by our variant of the KR algorithm applied to \( J \). Let \( LP_{(1, \infty)}(J) \) denote the value of the KR linear programming relaxation for strip packing when items can be packed fractionally in the \( y \)-direction. As this solution satisfies the *wide not sliced* property, by Lemma 3.2 we obtain

\[
A_{(1, \infty)}(I) \leq c[\text{KR}_{(1, \infty)}(J)] \leq c[(1 + \epsilon)\text{LP}_{(1, \infty)}(J) + O(1/\epsilon^2))].
\]

We now lower bound the value of the optimal 3D strip packing.

By Lemmas 4.2 and 4.1 and as \( c \geq 1/\epsilon \), we get

\[
\text{Opt}_{(1, \infty, c)}(\tilde{I}) \leq \text{Opt}_{(1, 1, c)}(I) \leq \frac{1 + \epsilon}{c}\text{Opt}_{(1, 1, \infty)}(I) + 2.
\]
By placing the \((1, T_{\infty}, c)\) sized bins side by side along the \(y\)-dimension to obtain a strip packing, we also have \(\text{Opt}_{(1, \infty, c)}(\tilde{I}) \leq T_{\infty} \cdot \text{Opt}_{(1, 1, \infty)}(\tilde{I})\). Together these inequalities imply that

\[
\text{Opt}_{(1, \infty, c)}(\tilde{I}) \leq (1 + \epsilon)T_{\infty} \cdot \text{Opt}_{(1, 1, \infty)}(\tilde{I})/c + O(1).
\]

In the rest of this section, our goal will be to show that

\[
\text{LP}_{(1, \infty)}(J) \leq \frac{1}{1 - 3\epsilon} \text{Opt}_{(1, \infty, c)}(\tilde{I}) + O(1/\epsilon^2).
\]

Together with (4.1) and (4.2), this would imply that

\[
A_{(1, 1, \infty)}(I) \leq \frac{(1 + \epsilon)^2}{1 - 3\epsilon} T_{\infty} \cdot \text{Opt}_{(1, 1, \infty)}(I) + O(1/\epsilon^3)
\]

and hence that the algorithm has an approximation ratio approaching \(T_{\infty}\).

Proving inequality (4.3). To prove inequality (4.3), it will be convenient to define a fractional notion of packing 3D items in a 3D strip. This notion is analogous to the one that Kenyon and Rémila use for a 2D strip [14]. We remark that this notion is only used for the purpose of the proof and is not used by our algorithm.

Given an instance of 3D items \(I\), let us define a fractional packing in a strip \((1, \infty, c)\) as one where each item can be sliced arbitrarily along both the \(y\)-dimension and the \(z\)-dimension (i.e., only the \(x\)-dimension remains unchanged), and these slices can be placed arbitrarily. In other words, let \(\delta\) be some fixed small rational number such that both \(y_s\) and \(z_s\) are integer multiples of \(\delta\). Then, in the fractional packing we treat an item \(s\) of size \((x_s, y_s, z_s)\) as a collection of \(y_s z_s/\delta^2\) “pencils” of size \((x_s, \delta, \delta)\), and these pencils can be placed arbitrarily in the strip \((1, \infty, c)\).

For a 3D instance \(I\), let \(F_{(1, \infty, c)}(\tilde{I})\) denote the value of an optimal fractional packing (over all choices of \(\delta\)) in \((1, \infty, c)\). As in the KR algorithm [14], we have the following.

Lemma 4.6. The value of the optimal fractional packing \(F_{(1, \infty, c)}(\tilde{I})\) is given by a linear program.

Proof. For the instance \(\tilde{I}\), let us define a valid configuration \(C\) as any multiset of items \(s \in \tilde{I}\) such that the total sum of depths in \(C\) is at most 1, i.e., \(C\) is a valid configuration if \(\sum_{s \in C} x_s \leq 1\). Let \(C\) denote the set of all valid configurations. Define \(n_{s,C}\) as the number of times the item \(s\) appears in the configuration \(C\), and the variable \(v_C\) as the total amount of configuration \(C\) used. Then the \(F_{(1, \infty, c)}(\tilde{I})\) is given by an optimal solution to the following LP:

\[
\min \frac{1}{c} \sum_{C \in \mathcal{C}} v_C \quad \text{s.t.} \quad \sum_{C} n_{s,C} \cdot v_C \geq x_s y_s z_s \quad \forall s \in \tilde{I} \quad \text{and} \quad v_C \geq 0 \quad \forall C.
\]

This follows, as any valid fractional packing must satisfy the constraints of this LP, and conversely any solution to the LP above directly gives a feasible fractional packing of \(\tilde{I}\) for a suitable choice of \(\delta\).

Consider the instance \(\tilde{J}\) consisting of slabs \((x, y, c)\), and the derived 2D instance \(J\). Since each item \(s\) has height \(z_s = c\) in \(\tilde{J}\), the LP for \(F_{(1, \infty, c)}(\tilde{J})\) is identical to the KR LP for a fractional 2D strip, and hence we have that

\[
F_{(1, \infty, c)}(\tilde{J}) = \text{LP}_{(1, \infty)}(J).
\]
Also, the fractional packing of $\bar{I}$ is a relaxation of an actual packing,

\begin{equation}
F_{(1,\infty,c)}(\bar{I}) \leq \text{Opt}_{(1,\infty,c)}(\bar{I}).
\end{equation}

We will now show that

\[ F_{(1,\infty,c)}(J) \leq \frac{1}{1-\epsilon} F_{(1,\infty,c)}(\bar{I}) + O(1/\epsilon^2). \]

Note that together with (4.4) and (4.5), this will imply (4.3).

**Lemma 4.7.** Let $\eta = (1-1/\epsilon)(1-2/k)$ and $k = \lceil 1/\epsilon \rceil + 2$. For the instances $\bar{I}$ and $J$ as defined above, it holds that

\[ F_{(1,\infty,c)}(J) \leq \frac{1}{\eta} F_{(1,\infty,c)}(\bar{I}) + k^2. \]

**Proof.** We define an intermediate instance $J'$ from $\bar{J}$ as follows. For each slab $G_q(i) \in J$ such that $q = 1, \ldots, k^2$ and $i = 2, \ldots, n_q$ (i.e., all slabs except the first one in each class $q$), we define a new slab $G'_q(i-1)$ which has the same $x$- and $y$-dimension as $G_q(i)$, but has height $\eta c$ (instead of $c$).

So $\bar{J}$ contains at most $k^2$ extra slabs as compared to $J'$, and the other slabs are a copy of slabs in $J'$ with the $z$-dimension scaled up by $1/\eta$ times. As scaling an instance in the $y$- or $z$-dimension only affects the value of the LP for the fractional packing by the same amount, it follows that

\[ F_{(1,\infty,c)}(J) \leq \frac{1}{\eta} F_{(1,\infty,c)}(J') + k^2. \]

We now show that $F_{(1,\infty,c)}(J') \leq F_{(1,\infty,c)}(\bar{I})$, which together with the above inequality will complete the proof of this lemma.

Consider some optimal fractional packing of $\bar{I}$ attaining the value $F_{(1,\infty,c)}(\bar{I})$. We claim that for each $q = 1, \ldots, k^2$ and $i = 2, \ldots, n_q - 1$, the slab $G'_q(i)$ can be packed fractionally wherever the items $I_q(i)$ are packed (recall that these are items from $\bar{J}$ in the slab $G_q(i)$). This follows, as each item in $I_q(i)$ has depth more than that of every item in $I_q(i+1)$ (by Observation 4.5) and hence the depth of slab $J'_q(i)$ (which is equal to the maximal item depth among items in $I_q(i+1)$). Moreover, by Observation 4.4, the $yz$-area for each slab $G'_q(i)$ for $q = 1, \ldots, k^2$ and $i = 1, \ldots, n_q - 1$ is no more than the $yz$-area of $I_q(i)$. This follows, as for $q = 1, \ldots, k^2 - 1$, each slab $G_q(i)$ (except for $i = n_q$) has $yz$-area at least $(c-1)/q$, while the slab $G'_q(i)$ has $yz$-area equal to $\eta c/q$. Similarly, for $q = k^2$, each slab $G_q(i)$ (except for $i = n_q$) has $yz$-area at least $(c-1)/(1/k^2 - 2/k^2) = \eta c/k$, while the slab $G'_q(i)$ has area exactly $\eta c/k$. \qed

5. **Concluding remarks.** We designed an approximation algorithm with performance guarantee approaching $T_\infty \approx 1.69103$ for the 3D strip packing problem based on the connection to the 2D bin packing algorithm and the algorithm of Caprara [5]. Similarly, using the known APTAS for 2D bin packing where all items are squares [4], our technique implies an APTAS for the special case of 3D strip packing where all the items have square bases (bottoms). This result has also been mentioned by Correa [8]. However, we do not know how to use this connection in general. In particular, we are not aware how to extend the $1 + \ln(T_\infty) \approx 1.52$ approximation for 2D bin packing problem [2, 3] to 3D strip packing.
REFERENCES