



Improved Approximation Algorithms for Three-Dimensional Bin Packing

Downloaded from: <https://research.chalmers.se>, 2026-04-22 15:24 UTC




Citation for the original published paper (version of record):

Kar, D., Khan, A., Rau, M. (2025). Improved Approximation Algorithms for Three-Dimensional Bin Packing. Leibniz International Proceedings in Informatics, LIPIcs, 334.

<http://dx.doi.org/10.4230/LIPIcs.ICALP.2025.104>

N.B. When citing this work, cite the original published paper.

Improved Approximation Algorithms for Three-Dimensional Bin Packing

Debajyoti Kar   

Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, India

Arindam Khan   

Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, India

Malin Rau  

Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, Göteborg, Sweden

Abstract

We study three fundamental three-dimensional (3D) geometric packing problems: 3D (Geometric) Bin Packing (3D-BP), 3D Strip Packing (3D-SP), and Minimum Volume Bounding Box (3D-MVBB), where given a set of 3D (rectangular) cuboids, the goal is to find an axis-aligned nonoverlapping packing of all cuboids. In 3D-BP, we need to pack the given cuboids into the minimum number of unit cube bins. In 3D-SP, we need to pack them into a 3D cuboid with a unit square base and minimum height. Finally, in 3D-MVBB, the goal is to pack into a cuboid box of minimum volume.

It is NP-hard to even decide whether a set of rectangles can be packed into a unit square bin – giving an (absolute) approximation hardness of 2 for 3D-BP and 3D-SP. The previous best (absolute) approximation for all three problems is by Li and Cheng (SICOMP, 1990), who gave algorithms with approximation ratios of 13, $46/7$, and $46/7 + \varepsilon$, respectively, for 3D-BP, 3D-SP, and 3D-MVBB. We provide improved approximation ratios of 6, 6, and $3 + \varepsilon$, respectively, for the three problems, for any constant $\varepsilon > 0$.

For 3D-BP, in the asymptotic regime, Bansal, Correa, Kenyon, and Sviridenko (Math. Oper. Res., 2006) showed that there is no asymptotic polynomial-time approximation scheme (APTAS) even when all items have the same height. Caprara (Math. Oper. Res., 2008) gave an asymptotic approximation ratio of $T_\infty^2 + \varepsilon \approx 2.86$, where T_∞ is the well-known Harmonic constant in Bin Packing. We provide an algorithm with an improved asymptotic approximation ratio of $3T_\infty/2 + \varepsilon \approx 2.54$. Further, we show that unlike 3D-BP (and 3D-SP), 3D-MVBB admits an APTAS.

2012 ACM Subject Classification Theory of computation → Packing and covering problems

Keywords and phrases Approximation Algorithms, Geometric Packing, Multidimensional Packing

Digital Object Identifier 10.4230/LIPIcs.ICALP.2025.104

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version:* <https://arxiv.org/abs/2503.08863> [39]

Funding *Debajyoti Kar:* Google PhD Fellowship.

Arindam Khan: Google India Research Award, SERB Core Research Grant (CRG/2022/001176) on “Optimization under Intractability and Uncertainty”, Ittiam Systems CSR grant, and the Walmart Center for Tech Excellence at IISc (CSR Grant WMGT-23-0001).

1 Introduction

Three-dimensional (3D) packing problems are used to model several practical settings in production and transportation planning – ranging from cargo management, manufacturing, 3D printing and prototyping, to cutting and loading applications. In the 1960s, Gilmore and Gomory [27] introduced 3D packing in the context of the cutting stock problem in operations research, where given a stock material (3D cuboid), the goal is to cut out a set of required



© Debajyoti Kar, Arindam Khan, and Malin Rau;
licensed under Creative Commons License CC-BY 4.0

52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025).

Editors: Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis

Article No. 104; pp. 104:1–104:20



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



items (smaller 3D cuboids) by a sequence of end-to-end cuts. Around the same time, Meir and Moser [50] asked a combinatorial question: given a set of cubes, when can we pack them in a given cuboid? Since then, due to its inherent mathematical aesthetics, computational nature, and practical relevance, the study of 3D packing has led to the development of several techniques in mathematics, computer science, and operations research.

In this paper, we consider three classical 3D packing problems. In all of these problems, the input is a collection of (rectangular) cuboids (items), each specified by their height, width, and depth. In the 3D Bin Packing (3D-BP) problem, the goal is to output a packing of all the items using the minimum number of bins, where each bin is a unit cube. In the 3D Strip Packing (3D-SP) problem, we are given a three-dimensional strip having a 1×1 square base and unbounded height, and we have to pack all items minimizing the height of the strip. Finally, in the Minimum Volume Bounding Box (3D-MVBB) problem, we seek to obtain a cuboidal box of minimum volume that can accommodate all input items. In all these problems, the items cannot be rotated about any axis, and they must be packed non-overlappingly. Further, we assume that all items and bins/boxes are axis-aligned.

With the recent exponential growth in transportation and shipping, specially with the advent of e-commerce and UAVs, these problems are receiving increasingly more attention. For instance, in container ship loading, it is crucial to optimize the placement of cargo to maximize space usage while minimizing the number of containers needed. In pallet loading, manufacturers strive to stack goods on pallets in a way that maximizes storage capacity and ensures secure transport. Further, in supply chain management, it is crucial to optimize the arrangement of goods in storage to fit within the smallest possible space, reducing storage costs and enhancing inventory accessibility. The survey by Ali, Ramos, Carravilla, and Oliveira [3] provides a comprehensive overview of 3D packing, with more than two hundred research articles. We refer readers to [22, 32, 45, 17, 53, 16, 38, 44, 48, 47] and [26, 11, 4, 20, 56, 55] for important empirical procedures and heuristics to 3D-BP and 3D-SP, respectively. There are also many practical programming competitions for these problems, e.g., OPTIL 3D Bin Packing Challenge [2] and ICRA VMAC Palletization Competition [1].

In contrast to the above, the theoretical exploration of 3D packing has been significantly limited due to its inherently complicated nature. All three considered problems are NP-hard. In fact, 3D-BP and 3D-SP generalize several classical strongly NP-hard problems in scheduling and packing, including (1D) bin packing, multiprocessor scheduling [43], packing squares into squares [23], and packing cubes into cubes [46]. In this paper, we study the absolute and asymptotic approximation algorithms for these problems. Given an algorithm \mathcal{A} for a minimization problem Π , the *absolute approximation ratio* of \mathcal{A} is defined as $\max_{I \in \mathcal{I}} \{\mathcal{A}(I)/\text{OPT}(I)\}$, where \mathcal{I} is the set of all input instances for Π , and $\mathcal{A}(I), \text{OPT}(I)$ are the values of the solution provided by \mathcal{A} and the optimal solution for an input instance I , respectively. The *asymptotic approximation ratio* (AAR) is defined as: $\limsup_{m \rightarrow \infty} \max_{I \in \mathcal{I}} \left\{ \frac{\mathcal{A}(I)}{\text{OPT}(I)} \mid \text{OPT}(I) = m \right\}$. A problem is said to admit an asymptotic polynomial-time approximation scheme (APTAS) if, for any $\varepsilon > 0$, there exists a polynomial-time algorithm \mathcal{A}_ε with AAR of $(1 + \varepsilon)$. 3D-BP and 3D-SP generalize 2D Bin Packing. Thus they do not admit an APTAS, as 2D Bin Packing has an asymptotic approximation hardness of $1 + 1/2196$ [13]. Furthermore, even for squares, it is NP-hard to decide if a set of squares can be packed in a single square bin or not [23] – thus giving an absolute approximation hardness of 2 for 3D-BP and 3D-SP.

Two-dimensional variants of these problems have been extensively studied. For 2D-BP, Harren, Jansen, Prädell, Schwarz, and van Stee [30] gave a *tight* absolute 2-approximation,

and a line of work [12, 7, 35] culminated in an asymptotic 1.406-approximation due to Bansal and Khan [10]. For 2D-SP, the asymptotic approximation regime is settled by the AFPTAS due to Kenyon and Rémila [40]. However, the best-known absolute approximation ratio for 2D-SP stands at $(5/3 + \varepsilon)$ [29] and there is a 3/2-hardness. In pseudopolynomial-time, there is an almost tight (absolute) $(5/4 + \varepsilon)$ -approximation algorithm [36, 31]. Finally, for 2D-MVBB, Bansal, Correa, Kenyon, and Sviridenko [8] gave a PTAS.

For 3D-BP, Csirik and van Vliet [18] gave an asymptotic approximation ratio of $T_\infty^3 + \varepsilon \approx 4.836$, where $T_\infty \approx 1.691$ is the omnipresent Harmonic constant [41] in Bin Packing, and the same ratio was achieved by Epstein and van Stee [21] by an online algorithm using bounded space. This was later improved to $T_\infty^2 + \varepsilon \approx 2.86$ by Caprara [12], which stands as the currently best-known asymptotic approximation ratio for 3D-BP.

For 3D-SP, Li and Cheng [43] demonstrated that simple heuristics such as NFDH or FFDH for 2D packing [15] have unbounded AARs. Then they provided an algorithm that returns a packing into a strip of height at most $(13/4)\text{OPT}_{3\text{D-SP}} + 8h_{\max}$, where $\text{OPT}_{3\text{D-SP}}$ denotes the optimal Strip Packing height, and h_{\max} is the maximum height of an input item. Afterwards, there has been a long line of work [43, 42, 52, 51, 37, 9] on the asymptotic approximability of 3D-SP, culminating in a $(3/2 + \varepsilon)$ -approximation by Jansen and Prädél [34]. However, all these improved asymptotic approximation algorithms incur huge additive loss (more than 100).

The authors in [19] obtained an absolute approximation ratio of $(29/4 + \varepsilon)$ for 3D-SP, and claimed it to be the best-known ratio for the problem. However, Li and Cheng [43] had also designed an algorithm that returns a packing into a strip of height at most $(32/7)\text{OPT}_{3\text{D-SP}} + 2h_{\max}$. This already gives a better absolute approximation ratio of $46/7$ for 3D-SP. Since $\text{OPT}_{3\text{D-SP}}$ is a lower bound on the minimum number of unit (cube) bins needed to pack all items, an absolute α -approximation for 3D-SP directly implies an absolute $(2\lfloor \alpha \rfloor + 1)$ -approximation for 3D-BP – one can obtain bins by cutting a 3D-SP solution at integral heights, followed by packing the items intersected by the cutting planes into additional separate bins. Thus the $46/7$ -approximation for 3D-SP implies an absolute 13-approximation for 3D-BP, which we believe to be the best-known approximation ratio for 3D-BP.

Finally, it is easy to obtain an absolute $\alpha(1 + \varepsilon)$ -approximation for 3D-MVBB from an absolute α -approximation for 3D-SP. Applying this strategy to the claimed $(29/4 + \varepsilon)$ -approximation algorithm for 3D-SP [19], Alt and Scharf [6] obtained a $(29/4 + \varepsilon)$ -absolute approximation for 3D-MVBB. However, as mentioned before, the result of Li and Cheng [43] can also be extended to a $(46/7 + \varepsilon)$ -approximation for 3D-MVBB.

There have been some improvements for special cases. For example, Bansal, Correa, Kenyon, and Sviridenko [8] provided an APTAS for d -dimensional bin packing with d -dimensional cubes. Harren [28] gave APTAS for d -dimensional strip packing with d -dimensional cubes when the base of the strip has a bounded aspect ratio. Jansen, Khan, Lira, and Sreenivas [33] extended the APTAS to more general bases (not necessarily rectangular).

However, for general cuboids, there has been no progress on the absolute approximation ratios for any of the three problems since 1990 [43] and for asymptotic approximation ratio of 3D-BP since 2008 [12]. In [8], the authors mention the inherent difficulty in extending results from 2D packing to 3D packing, due to the more complicated nature of interactions between different types of items in three dimension. In fact, improved approximability of d -dimensional geometric Bin Packing and Strip Packing, for $d > 2$, was listed as *one of the ten major open problems* in the survey on multidimensional packing [14].

■ **Table 1** Summary of results. WR denotes the case when 90° rotation around any axis is allowed.

Problem	Absolute Approximation Ratio		Asymptotic Approximation Ratio	
	Previous Best	Our Result	Previous Best	Our Result
3D-BP	13 [43]	6 (WR: 5)	$T_\infty^2 + \varepsilon < 2.86$ [12]	$3T_\infty/2 + \varepsilon < 2.54$
3D-SP	$46/7 \approx 6.58$ [43]	6	$3/2 + \varepsilon$ [34]	–
3D-MVBB	$46/7 + \varepsilon$ [43, 6]	$3 + \varepsilon$	$46/7 + \varepsilon$ [43, 6]	$1 + \varepsilon$

1.1 Our contribution

We present improved absolute approximation algorithms for 3D-BP, 3D-SP, and 3D-MVBB. Further, we obtain improved asymptotic approximation algorithms for 3D-BP and 3D-MVBB. Our results are summarized in Table 1.

First, we discuss our results on the absolute approximation algorithms. We show how a packing in k bins can be transformed into $6k$ *structured* bins; following which, for constant k , it is possible to find such a structured packing efficiently using a variant of the Generalized Assignment Problem – giving us an absolute approximation ratio of 6. One interesting idea is that we use an asymptotic approximation algorithm to obtain improved absolute approximation. One of our key ingredients is the asymptotic approximation algorithm for 3D-SP by Jansen and Prädél [37], which provides a packing into height at most $(3/2 + \varepsilon)\text{OPT}_{3\text{D-SP}} + \varepsilon + O_\varepsilon(1)h_{\max}$.¹ Thus, for a sufficiently small appropriate constant μ , if $h_{\max} \leq \mu$, then we can actually pack all items into $\lfloor 3k/2 \rfloor + 1$ bins, assuming there exists a packing into k bins. So, we partition the items into four classes: L (large items: all dimensions are greater than μ), I_w, I_d, I_h (width, depth, height less than μ , resp.). If an item belongs to multiple classes, we assign them to any one arbitrarily. Now *large* items can be packed in k bins by brute-force enumeration in polynomial-time (for constant k, μ). Each of the remaining three classes can be packed into $\lfloor 3k/2 \rfloor + 1$ bins. In total, we get $3(\lfloor 3k/2 \rfloor + 1) + k \leq 7k$ bins.

To improve further, we pack *large* items together with some items from one of the three classes. First, we observe that one of these classes has volume less than $k/3$. W.l.o.g. let us assume it to be I_h . Now first we use a volume-based argument and use an algorithm from [43] to show that we can pack all items in I_h whose width or depth is less than $1/2$. The remaining items in I_h have both width and depth exceeding $1/2$. Next, we show that we can guess the packing of *large* items and almost all items in I_h , except a set of items with small volume. However, with a refined and technical analysis, we finally show that even these remaining items can be packed in the free regions of the six bins. For 3D-BP, this yields an improvement over the previous bound of 13 [43].

► **Theorem 1.** *There exists a polynomial-time 6-approximation algorithm for 3D-BP.*

This directly implies an absolute $(6 + \varepsilon)$ -approximation for 3D-SP – guess the optimal Strip Packing height within a $(1 + \varepsilon)$ -factor, then use appropriate scaling to apply the above 3D-BP result, and finally stack the obtained six bins. With a more careful analysis, we can show there is some extra empty space in the strip and the resulting height is strictly below 6.

¹ The notation $O_\varepsilon(f(n))$ means that the implicit constant hidden in big- O notation can depend on ε .

► **Theorem 2.** *There exists a small absolute constant $\rho > 0$, such that for any $\varepsilon > 0$, there is a polynomial-time $(6 - \rho + \varepsilon)$ -approximation algorithm for 3D-SP.*

When the items can be rotated by 90° around any axis, the approximation ratio for 3D-BP improves to 5 with a more careful analysis.

► **Theorem 3.** *There exists a polynomial-time 5-approximation for 3D-BP with rotations.*

Another implication of our result is a $(6 + \varepsilon)$ -approximation for the 3D-MVBB problem, using the connection between 3D-SP and 3D-MVBB [6]. However, we then use the power of resource augmentation in 2D-BP to obtain an APTAS for 3D-SP when we are allowed to use resource augmentation. With additional technical adaptations, we obtain a $(3 + \varepsilon)$ -approximation for 3D-MVBB.

Furthermore, surprisingly, unlike 3D-BP and 3D-SP, we show that 3D-MVBB admits an APTAS – settling the asymptotic approximability for the problem.

► **Theorem 4.** *For any $\varepsilon > 0$, there exists a polynomial-time $(3 + \varepsilon)$ -approximation algorithm and an asymptotic polynomial-time approximation scheme for 3D-MVBB.*

Finally, we turn our attention to the asymptotic approximability of 3D-BP. Towards this, we exploit connections between 3D-SP and 3D-BP. Let $\text{OPT}_{3\text{D-SP}}(I)$, $\text{OPT}_{3\text{D-BP}}(I)$ be the values of the optimal solution for Strip Packing and Bin Packing, for input I , respectively. Then $\text{OPT}_{3\text{D-SP}}(I) \leq \text{OPT}_{3\text{D-BP}}(I)$, as the bins can be stacked on top of each other to provide a feasible solution for Strip Packing. Thus, one can trivially obtain a $(3 + \varepsilon)$ -approximation algorithm as follows. First, we obtain a packing in height $(\frac{3}{2} + \varepsilon)\text{OPT}_{3\text{D-SP}}(I) + O_\varepsilon(1)$ using [34]. We then can cut the strip into unit cube bins by cutting it at integral heights. All items that are completely contained within heights $[i, i + 1)$ are packed in the $(2i + 1)$ -th bin. Remaining items that are cut by the x - y axis-aligned plane at height i (these items form one layer of items where each item has height at most $h_{\max} \leq 1$) are packed in $(2i)$ -th bin. This would give us a packing into $(3 + \varepsilon)\text{OPT}_{3\text{D-SP}}(I) + O_\varepsilon(1)$ bins.

To improve beyond T_∞^2 , our approach will be to find a packing such that the items that are cut do not have large heights. Towards this, we use *harmonic rounding* [41], where the function f_k rounds up $\alpha \in (1/k, 1]$ to nearest larger number of the form $1/q$ where $q \in \mathbb{Z}$. Thus, for $\alpha_i \in (1/(q + 1), 1/q]$, $f_k(\alpha_i) := 1/q$, for $q \in [k - 1]$. Otherwise, $f_k(\alpha_i) := \alpha_i$. It is well-known [9] that, for any sequence $\alpha_1, \alpha_2, \dots, \alpha_n$, with $\alpha_i \in (0, 1]$ and $\sum_{i=1}^n \alpha_i \leq 1$, for a small enough ε , we have $\sum_{i=1}^n f_{1/\varepsilon}(\alpha_i) \leq T_\infty + \varepsilon \approx 1.691$.

We first round the item heights in I using $f_{1/\varepsilon}$ to obtain a new set of items I^∞ and obtain a 3D Strip Packing of them using the algorithm by [37]. Let $\text{OPT}_{3\text{D-BP}}^{T_\infty}(I^\infty)$ be the optimal number of $1 \times 1 \times T_\infty$ -sized bins needed to pack all items in I^∞ . Then, it is easy to see that $\text{OPT}_{3\text{D-SP}}(I^\infty) \leq T_\infty \text{OPT}_{3\text{D-BP}}^{T_\infty}(I^\infty)$.² Then we have $\frac{3}{2}\text{OPT}_{3\text{D-SP}}(I^\infty) \leq \frac{3T_\infty}{2}\text{OPT}_{3\text{D-BP}}^{T_\infty}(I^\infty) \leq \frac{3T_\infty}{2}\text{OPT}_{3\text{D-BP}}(I)$. The last inequality follows from harmonic rounding.

Now we need to ensure that the *tall* items in I^∞ packed in the strip with height $\frac{3}{2}\text{OPT}_{3\text{D-SP}}(I^\infty)$ are not cut by the cutting planes at integral heights – we call this *tall-not-sliced* property. A similar idea was used by Bansal, Han, Iwama, Sviridenko, and Zhang [9] to obtain an alternate $(T_\infty + \varepsilon)$ -approximation for 2D-BP. However, 3D packing is much more involved. For this, we exploit the structural properties from the packing by [34]. First, we show that the strip can be divided into $O_\varepsilon(1)$ cuboids such that, for each cuboid, the corresponding items packed inside are *similar*. Next, we show that we can pack almost all

² For simplicity, we are ignoring the $O_\varepsilon(1)$ in the following discussion in this section.

tall items in I^∞ of the same height ($1/q$ for some $q \in [k-1]$) at heights that are multiple of $1/q$ and incur only a small additive loss. This will ensure that none of these items are cut by planes at integral heights. For items with big width and depth, we use a linear program to assign items to the containers. For other items (except a small volume of them), the packing is based on variants of the Next-Fit-Decreasing-Height (NFDH) algorithm [15]. Finally, we show that we can pack the remaining items in the remaining free regions and an additional $O_\varepsilon(1)$ bins. This provides an improved guarantee for 3D-BP after nearly two decades.

► **Theorem 5.** *For any $\varepsilon > 0$, there exists a polynomial-time algorithm for 3D-BP with an asymptotic approximation ratio $(3T_\infty/2 + \varepsilon) \approx 2.54$.*

Organization of the paper. In Section 2, we present some preliminaries needed for our results. Section 3 provides absolute approximation algorithms for 3D-BP and 3D-SP, and we prove Theorems 1 and 2. Section 4 deals with the asymptotic approximation algorithm for 3D-BP and establishes Theorem 5. In Section 5, we discuss results related to 3D-MVBB and prove Theorem 4. Section 6 provides absolute approximation guarantee for 3D-BP with rotation. Finally, Section 7 ends with a conclusion. Due to space constraints, most of the details have been shifted to the full version of the paper [39].

1.2 Related work

For $d > 3$, Caprara [12] gave an algorithm with AAR of T_∞^{d-1} for both d -dimensional Bin Packing and Strip Packing. Sharma [54] gave T_∞^{d-1} -asymptotic approximation for these two problems when the items can be orthogonally rotated about all or a subset of axes. For 3D-MVBB, if the items are allowed to be rotated by 90 degrees about any axis, Alt and Scharf [6] gave a 17.738-approximation. Another related problem is the 3D Knapsack problem, in which each item additionally has an associated profit, and the goal is to obtain a maximum profit packing inside a unit cube knapsack. The authors in [19] have given a $(7 + \varepsilon)$ -approximation algorithm. For other related problems, we refer the readers to the surveys on approximation algorithms for multidimensional packing [5, 14].

2 Preliminaries

We define width, depth, and height along x, y, z axes, respectively. Let I be the given set of n items, where each item $i \in I$ is an axis-aligned cuboid having height, width, and depth equal to h_i, w_i, d_i , respectively. Let $h_{\max}, w_{\max}, d_{\max} \in (0, 1]$ be the maximum height, width, and depth of an item in I , respectively. Given a box $B := [0, W] \times [0, D] \times [0, H]$, if the (bottom-left-back corner of) item i is placed (by translation) at (x_i, y_i, z_i) then it occupies the region: $[x_i, x_i + w_i] \times [y_i, y_i + d_i] \times [z_i, z_i + h_i]$, and the packing is feasible if $x_i \in [0, W - w_i], y_i \in [0, D - d_i], z_i \in [0, H - h_i]$. In this placement, we define the top, right, and back faces of item i to be $[x_i, x_i + w_i] \times [y_i, y_i + d_i] \times \{z_i + h_i\}$, $\{x_i + w_i\} \times [y_i, y_i + d_i] \times [z_i, z_i + h_i]$, and $[x_i, x_i + w_i] \times \{y_i + d_i\} \times [z_i, z_i + h_i]$, respectively. Analogously, bottom, left, and front faces are defined. Two items do not overlap if their interiors are disjoint. The volume of item i is $v(i) := h_i w_i d_i$. For any set T , let $v(T)$ denote the total volume of items in T . We define $\text{OPT}_\Pi(I)$ to be the value of the optimal solution for problem Π on instance I .

2.1 Algorithms for 3D Packing

We now state three results on 3D packing that will be crucial for our results. The first two of them give a volume-based guarantee.

- **Theorem 6** ([43]). *Let T be a set of 3D items where each item has height bounded by h_{\max} .*
1. *All items in T can be packed into a strip with 1×1 base and height $4v(T) + 8h_{\max}$.*
 2. *If further each item has either width or depth (or both) not exceeding $1/2$, then all items in T can be packed inside a strip with 1×1 base and height $3v(T) + 8h_{\max}$.*

- **Theorem 7** ([49]). *Given a set of 3D items T , there is a polynomial-time algorithm that places these items into at most $8v(T) + O(1)$ bins.*

The last result is regarding the asymptotic approximation of 3D-SP.

- **Theorem 8** ([34]). *Given a set of 3D items I where each item has height bounded by h_{\max} , for any constant $\varepsilon > 0$, there is a polynomial-time algorithm that returns a packing of I into a strip of height at most $(3/2 + \varepsilon)\text{OPT}_{3\text{D-SP}}(I) + \varepsilon + O_\varepsilon(1)h_{\max}$.*

3 Absolute 6-approximation for 3D-BP

In this section, our goal is to prove Theorem 1. Let $K > 0$ be a large constant such that the algorithm of Caprara [12] already yields an absolute 6-approximation when $\text{OPT}_{3\text{D-BP}} > K$. Our goal is to obtain a 6-approximation for the case when $\text{OPT}_{3\text{D-BP}} \leq K$. Let $\lambda = 1/40$, and $\delta < \lambda$ be a sufficiently small constant. The following lemma follows from a standard shifting argument.

- **Lemma 9.** *There exists a polynomial-time computable $\mu \leq \delta$ such that the total volume of the items that have at least one of the dimensions in the range $(\mu^4, \mu]$ is at most δ .*

We classify the items depending on their dimensions: let L be the items whose height, width and depth all exceed μ (called *large* items), I_h be the items with height at most μ^4 , I_w be the remaining items having width at most μ^4 and I_d be the remaining items with depth at most μ^4 . Finally, let I^{rem} be the remaining items each having at least one of the dimensions in the range $(\mu^4, \mu]$. Note that $v(I^{\text{rem}}) \leq \delta$ owing to Lemma 9. We further classify the items of I^{rem} in a similar way – let $I_h^{\text{rem}} \subseteq I^{\text{rem}}$ be the items with height at most μ , $I_w^{\text{rem}} \subseteq I^{\text{rem}} \setminus I_h^{\text{rem}}$ be the remaining items with width at most μ , and $I_d^{\text{rem}} = I^{\text{rem}} \setminus (I_h^{\text{rem}} \cup I_w^{\text{rem}})$.

In the remainder of the section, we prove the following result.

- **Proposition 10.** *If there exists a packing of all items into $k \leq K$ bins, then a packing using at most $6k$ bins can be computed in polynomial-time.*

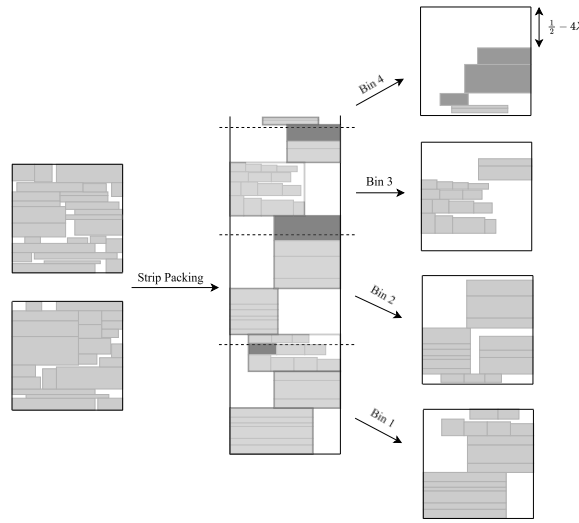
For this, we first show the following lemma, which follows from a simple application of Theorem 8 (see Figure 1 for an intuitive proof).

- **Lemma 11.** *Let T be a set of items each having a height (analogously width, depth) of at most μ , and suppose that there exists a packing of T into $k \leq K$ bins. Then it is possible to compute a packing of T using $\lfloor 3k/2 \rfloor + 1$ bins in polynomial-time. Further one of these bins has an empty strip with 1×1 base and height (analogously width, depth) $1/2 - 4\lambda$.*

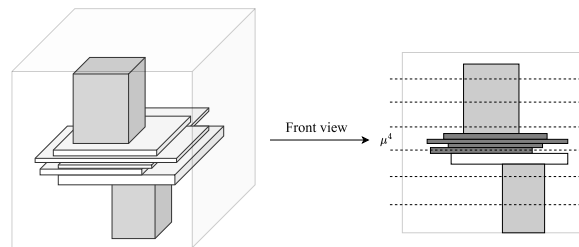
We divide the proof of Proposition 10 into two cases depending on $v(L)$.

3.1 Case 1: $v(L) > 64\delta K$

In this case, for some $j \in \{h, w, d\}$, the total volume of the items in I_j must not exceed $(k - 64\delta K)/3 \leq (1/3 - 21\delta)k$ – w.l.o.g. assume that $j = h$. We first pack the items of $I_w \cup I_w^{\text{rem}}$ and $I_d \cup I_d^{\text{rem}}$ into $\lfloor 3k/2 \rfloor + 1$ bins each, using Lemma 11. Our goal next is to pack the items of $L \cup I_h \cup I_h^{\text{rem}}$ into $2k$ bins. To this end, we further classify the items of I_h depending on their width and depth. Let $I_{h,\ell} := \{i \in I_h \mid w_i, d_i > 1/2\}$ and let $I_{h,s} := I_h \setminus I_{h,\ell}$. We first pack the items of $I_{h,s}$ into k bins using the following lemma, by applying Theorem 6.



■ **Figure 1** Packing from Lemma 11 (only the front view is shown for simplicity) for $k = 2$. The dark gray items are sliced while cutting out $\lfloor 3k/2 \rfloor + 1$ bins from the Strip Packing solution. Finally, sliced items are packed into the empty regions of the last bin.



■ **Figure 2** The light gray items are items of L . The dark gray items are deleted in order to position the upper large item at a multiple of μ^4 .

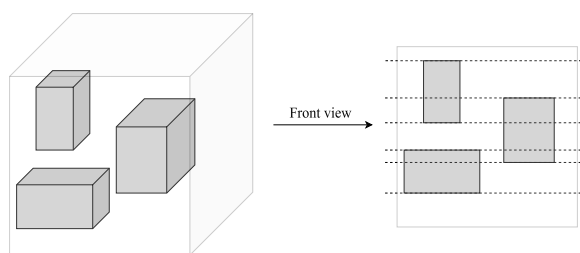
► **Lemma 12.** *The items of $I_{h,s}$ can be completely packed into k bins where each bin additionally has an empty strip with 1×1 base and height 59δ .*

Now consider the optimal packing inside k bins restricted to the items of $I_{h,\ell} \cup L$. Our goal is to compute a packing of all these items, barring a subset of items from $I_{h,\ell}$ that have a total volume of at most $O(\mu)k$. For this, we first discretize the positions of the large items inside the bins. We focus on the packing inside any one bin.

► **Lemma 13.** *By discarding items of $I_{h,\ell}$ having a total volume of at most 2μ , the number of distinct positions of the items of L can be assumed to be polynomially-bounded.*

Proof. Let us start with the optimal packing inside k bins restricted to the items of $I_{h,\ell} \cup L$. For discretizing the x - and y -positions, we push all items as much to the left and front as possible. Then the distance of the left (resp. front) face of an item of L from the left (resp. front) face of the bin can be written as the sum of the widths (resp. depths) of at most $1/\mu$ large items and at most one item of $I_{h,\ell}$. Thus there are polynomially-many choices.

In order to discretize the positions along the z -axis, we draw horizontal planes inside the bin at heights as integral multiples of μ^4 and consider the large items inside the bin from bottom to top. For each $i \in L$ in the order of increasing height of their bases, we discard items of $I_{h,\ell}$ in order to pull the item i downwards, until the bottom face of i hits one of



■ **Figure 3** The regions between two consecutive dotted lines correspond to slots.

the horizontal planes at heights multiples of μ^4 or the top face of another large item lying below i (see Figure 2). At the end of this process, it can be seen that the distance between the bottom face of any large item and the base of the bin can be written as the sum of a multiple of μ^4 and the heights of at most $1/\mu$ items of L . The total volume of the discarded items is bounded by $(1/\mu^3) \cdot 2\mu^4 = 2\mu$. ◀

We next draw horizontal planes passing through the top and bottom faces of each large item and discard the items of $I_{h,\ell}$ that are intersected by these planes. The volume of these discarded items is bounded by $(2/\mu^3) \cdot \mu^4 = 2\mu$. This partitions the bin into at most $2/\mu^3 + 1$ slots, where each slot is penetrated from top to bottom by at most $1/\mu^2$ large items (see Figure 3). Note also that no large item begins or ends in the interior of a slot. Together with Lemma 13, we thus have the following result.

► **Lemma 14.** *There exists a subset $I'_{h,\ell} \subseteq I_{h,\ell}$ with $v(I'_{h,\ell}) \geq v(I_{h,\ell}) - 4\mu k$ such that the items of $I'_{h,\ell}$ are completely packed inside the slots formed by the large items in the k bins.*

Our algorithm essentially tries to compute a packing close to the one guaranteed by the above lemma. As mentioned before, we obtain a packing into k bins of all items of L , and a large volume subset of $I_{h,\ell}$ that is packed inside the slots formed by the large items.

► **Lemma 15.** *In polynomial-time, it is possible to compute a set $I''_{h,\ell} \subseteq I_{h,\ell}$ with $v(I''_{h,\ell}) \geq v(I_{h,\ell}) - 5\delta k$, and a packing of all items in $I''_{h,\ell} \cup L$ into k bins.*

Proof. We first guess the positions (from polynomially many choices) of all the (at most K/μ^3) large items inside the k bins, and create slots by extending their top and bottom faces. Our goal is to pack a maximum volume subset of $I_{h,\ell}$ into these slots by creating an instance of the Generalized Assignment Problem (GAP) with $O(1)$ knapsacks, for which there exists a PTAS [25] (see Section A.3 for details). Each slot corresponds to a knapsack with capacity equal to the height of the slot. For an item $i \in I_{h,\ell}$, the size of i for a slot equals its height h_i if the item fits inside the slot, given the relative positions of the large items inside it, and ∞ otherwise. The profit of an item is the same as its volume. By Lemma 14, the optimal packing packs items of $I_{h,\ell}$ having a volume of at least $v(I_{h,\ell}) - 4\mu k$. We apply the PTAS for GAP with parameter δ/K to our instance, yielding a packing of a subset $I''_{h,\ell} \subseteq I_{h,\ell}$ into the slots formed by the items of L , whose volume is least $(1 - \delta/K)(v(I_{h,\ell}) - 4\mu k) \geq v(I_{h,\ell}) - 5\delta k$. ◀

It remains to pack the items of $I_{h,\ell} \setminus I''_{h,\ell}$ and I_h^{rem} . Intuitively, they have a small volume, and hence we can pack them into the empty regions inside the already-existing bins.

► **Lemma 16.** *The items of $I_{h,\ell} \setminus I''_{h,\ell}$ can be completely packed by using a height of 25δ from each of the empty regions inside the bins that were used to pack the items of $I_{h,s}$. Further, the items in I_h^{rem} can be packed within a height of 12δ inside one of the bins.*

Altogether, we used $2 \cdot (\lfloor 3k/2 \rfloor + 1) \leq 4k$ bins for packing items in $I_w \cup I_w^{\text{rem}} \cup I_d \cup I_d^{\text{rem}}$ and $2k$ bins for packing the items of $L \cup I_h \cup I_h^{\text{rem}}$, resulting in at most $6k$ bins overall.

3.2 Case 2: $v(L) \leq 64\delta K$

In this case, we first pack the items of $I_h \cup I_h^{\text{rem}}$, $I_w \cup I_w^{\text{rem}}$ and $I_d \cup I_d^{\text{rem}}$ into $\lfloor 3k/2 \rfloor + 1$ bins each, using Lemma 11 (note that these items have height, width and depth bounded by μ , respectively). Let B_h, B_w, B_d be the bins having empty strips of height, width, and depth $1/2 - 4\lambda$, respectively, that are guaranteed by Lemma 11. Intuitively, since the large items have a very small volume, they can be completely packed inside these empty strips.

► **Lemma 17.** *The items in L can be completely packed inside three strips, each having a 1×1 base aligned with the xy -, yz - and zx -planes, respectively. The strips have height, width, and depth of 33λ , respectively, and therefore they fit inside the bins B_h, B_w , and B_d .*

Overall, we obtain a packing into $3 \cdot (\lfloor 3k/2 \rfloor + 1) \leq 6k$ bins, establishing Proposition 10.

Overall algorithm. We first run the algorithm of Caprara [12] that already returns a 6-approximate solution when $\text{OPT}_{3\text{D-BP}} > K$. Next, for each guessed value of $\text{OPT}_{3\text{D-BP}} = k \leq K$, we run the algorithm of Proposition 10. For this, we first compute a value of μ using Lemma 9 and classify the items as discussed. Next, we divide into two cases depending on the volume of the large items. If $v(L) > 64\delta K$, we find $j \in \{h, w, d\}$ for which the volume of the items in I_j does not exceed $(1/3 - 21\delta)k$; w.l.o.g we take $j = h$. We pack the items of $I_w \cup I_w^{\text{rem}}$ and $I_d \cup I_d^{\text{rem}}$ into $\lfloor 3k/2 \rfloor + 1$ bins each using Lemma 11. We classify items of I_h into $I_{h,\ell}$ and $I_{h,s}$ depending on their width and depth, and obtain a packing of $I_{h,s}$ into k bins, ensuring each of these bins has an empty strip of height 59δ . Next we compute a set $I''_{h,\ell} \subseteq I_{h,\ell}$ such that $v(I_{h,\ell} \setminus I''_{h,\ell}) \leq 5\delta k$, and pack the items of $I''_{h,\ell} \cup L$ into k bins via a reduction to the Generalized Assignment Problem (Lemma 15). Finally, the items in $(I_{h,\ell} \setminus I''_{h,\ell}) \cup I_h^{\text{rem}}$ are packed into the empty spaces inside the bins for $I_{h,s}$ using Lemma 16. For the other case when $v(L) \leq 64\delta K$, we pack items in $I_h \cup I_h^{\text{rem}}$, $I_w \cup I_w^{\text{rem}}$ and $I_d \cup I_d^{\text{rem}}$ into $\lfloor 3k/2 \rfloor + 1$ bins each, using Lemma 11, ensuring one of the bins has a sufficiently large empty strip, and then pack items of L inside these empty strips using Lemma 17.

3.3 Implication on 3D-SP

We now establish Theorem 2. We use the following observation from our 3D-BP algorithm.

► **Lemma 18.** *If there exists a packing of all items into a single bin, then it is possible to compute a packing into 6 bins in polynomial-time, where one of the bins is filled up to a height of at most $1 - 22\delta$.*

Consider now the optimal Strip Packing of the input instance. Since the height of this packing must lie in $[h_{\max}, nh_{\max}]$, we can assume the optimal height to be of the form $h_{\max}(1 + \varepsilon)^j$, by losing only a factor of $1 + \varepsilon$. We scale the height of each item by the guessed height so that all items now fit inside a $1 \times 1 \times 1$ bin. Using Lemma 18, we compute a packing into 6 bins and stack these bins one on top of the other along the height, so that the resulting height of the packing is at most $6 - 22\delta$. This establishes Theorem 2 with $\rho = 22\delta$.

The above result holds for the case when the strip is unbounded along the z -axis. If instead the strip could be extended along any of the x -, y - or z -axes and the goal were to minimize the length of the strip along that direction, then we improve further.

► **Lemma 19.** *If there exists a packing of all items into a single bin, then for any $\varepsilon > 0$, it is possible to compute a packing into 6 bins in polynomial-time, where one of the bins is filled up to a length (height/width/depth) of at most $1/2 + O(\varepsilon)$.*

W.l.o.g. assume that one of the bins is filled up to height $1/2 + O(\varepsilon)$. Thus, stacking the bins along the z -axis would be of height at most $11/2 + O(\varepsilon)$, implying the following result.

► **Corollary 20.** *For any $\varepsilon > 0$, there exists a polynomial-time $(11/2 + \varepsilon)$ -approximation for 3D-SP, if the strip can be extended along any of the x -, y - or z -axes.*

4 Asymptotic $(\frac{3}{2} \cdot T_\infty + \varepsilon)$ -approximation for 3D-BP

In this section, we prove Theorem 5 and present an improved asymptotic approximation algorithm for 3D-BP. We will utilize ideas from the algorithm for 3D-SP by Jansen and Prädél [34] which packs I into strip height of $(3/2 + \varepsilon) \cdot \text{OPT}_{3\text{D-SP}}(I) + O_\varepsilon(1)h_{\max}$. As mentioned earlier, the naive approach of cutting the strip at integral height will result in $(3 + \varepsilon)$ -approximation. Instead, we exploit the structural properties of the solution provided by the algorithm, along with the *harmonically rounded heights* of the items, to ensure that items with a height larger than ε are not sliced.

Recall the definition of harmonic rounding: For $\alpha_i \in (1/(q+1), 1/q]$, $f_k(\alpha_i) := 1/q$, for $q \in [k-1]$; and for $\alpha_i \in (0, 1/k]$, $f_k(\alpha_i) := \alpha_i$. Also, if $\sum_{i=1}^n \alpha_i \leq 1$, then $\lim_{k \rightarrow \infty} \sum_{i=1}^n f_k(\alpha_i) \leq T_\infty \approx 1.691$. In the following, we assume $k = 1/\varepsilon$ to be large enough such that $\sum_{i=1}^n f_{1/\varepsilon}(\alpha_i) \approx T_\infty$, and define $f_{1/\varepsilon}$ to be f . See Section A.1 for more details on harmonic rounding.

Let I^∞ be the instance derived from the given 3D-BP instance I by applying harmonic rounding f to the heights of the items. Thus an item $i \in I$ becomes an item of I^∞ with width, depth, height to be $w_i, d_i, f(h_i)$, respectively. Let $\text{OPT}_{3\text{D-BP}}^{T_\infty}(I^\infty)$ denote the minimum number of bins with dimensions $1 \times 1 \times T_\infty$ required to pack all items from I^∞ , and let $\text{OPT}_{3\text{D-SP}}(I^\infty)$ denote the minimum height to pack all items from I^∞ into a strip with unit square base and unbounded height. The following lemma connects packing of I^∞ with I :

► **Lemma 21.** $\text{OPT}_{3\text{D-SP}}(I^\infty) \leq T_\infty \cdot \text{OPT}_{3\text{D-BP}}^{T_\infty}(I^\infty) \leq T_\infty \cdot \text{OPT}_{3\text{D-BP}}(I)$.

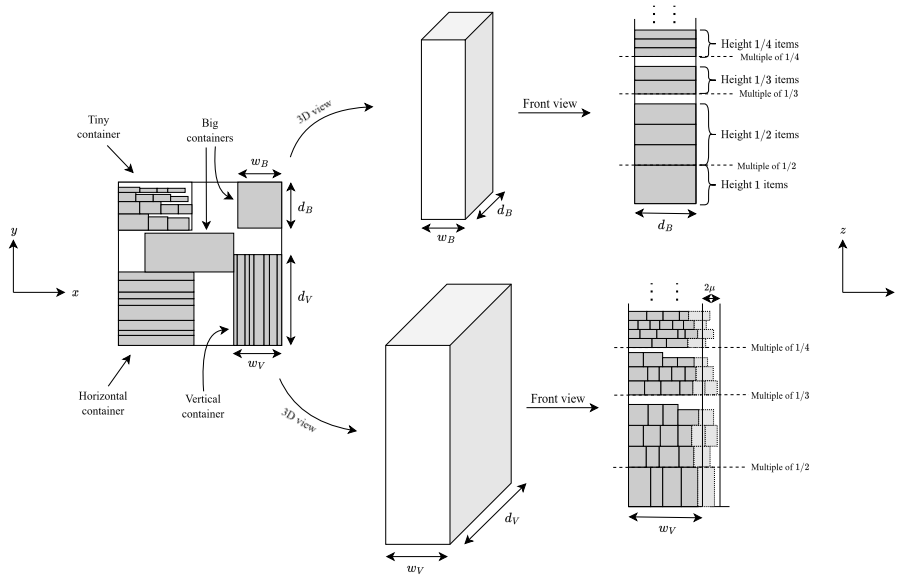
We transform the instance I^∞ for 3D-SP into an instance for 2D-BP, similar to [34], which then uses structural results from a 2D-BP algorithm [35]. Given ε , we round up the item heights of I^∞ to the next multiple of $\frac{\varepsilon v(I^\infty)}{n}$. We create an instance $I_{2\text{D-BP}}$ of 2D-BP, by introducing for each item $i \in I^\infty$ with rounded height $k \frac{\varepsilon v(I^\infty)}{n}$, exactly k rectangles with width w_i and depth d_i . The following lemma bounds the incurred loss.

► **Lemma 22.** $\frac{\varepsilon v(I^\infty)}{n} \text{OPT}_{2\text{D-BP}}(I_{2\text{D-BP}}) \leq (1 + \varepsilon) \text{OPT}_{3\text{D-SP}}(I^\infty)$.

4.1 Essentials of the 2D-BP algorithm

The main ingredient for the 2D-BP algorithm by Jansen and Prädél [35] is a restructuring theorem. It states that each packing can be rearranged into packing with an iterable structure. This rearrangement comes at the cost of introducing more bins to the packing.

First, the items are classified into big, vertical, horizontal, intermediate, and tiny based on a suitably chosen constant $\delta \in [\varepsilon^{O_\varepsilon(1)}, \varepsilon]$. Let $\mu = \delta^4$. An item i is *big* if $w_i \geq \delta$ and $d_i \geq \delta$; *vertical* if $d_i \geq \delta$ but $w_i < \mu$; *horizontal* if $w_i \geq \delta$ but $d_i < \mu$; *tiny* if $d_i < \mu$ and $w_i < \mu$; and *intermediate* if either $w_i \in [\mu, \delta)$ or $d_i \in [\mu, \delta)$. Using standard arguments, we can assume that the area of intermediate items is at most $\varepsilon \cdot \text{area}(I)$.



■ **Figure 4** Left figure depicts a 2D-container-packing, which forms the base of a 3D configuration. Middle figure shows two 3D containers corresponding to two containers (one big and one horizontal) in 2D-container-packing. On the right, the packing ensuring the *tall-not-sliced* property is shown. The light gray rectangles are repacked into additional bins.

Jansen and Prädél [35] showed the existence of a structured packing called *2D-container-packing*. In *2D-container-packing*, one can consider a rounded-up instance \tilde{I} from I_{2D-BP} , where the widths and depths of big items, widths of horizontal items, and depths of vertical items from I_{2D-BP} are rounded to $O(1/\delta^2)$ values. Let $\mathcal{T}, \mathcal{W}, \mathcal{D}$ be the set of different types of (rounded) large items, widths of wide items, and depths of vertical items in \tilde{I} , respectively. Then $|\mathcal{T}|$ is $O(1/\delta^4)$, and $|\mathcal{W}|, |\mathcal{D}|$ are $O(1/\delta^2)$. Furthermore, vertical items are allowed to be sliced along y -axis, horizontal items may be sliced along x -axis, and tiny items may be sliced in both directions. In *2D-container-packing*, each bin of the packing is partitioned into at most $O(1/\delta^4)$ containers and the widths and depths of containers come from $O(1/\delta^4)$ possible values, see Figure 4. The containers are of five types, and only specific types of items from \tilde{I} are allowed to be packed in the corresponding containers: (i) *Big container*: Each such container contains only one (rounded up) big item and has the size of this big item. Each bin contains at most $1/\delta^2$ such containers. (ii) *Horizontal containers*: each such container has a width $w \in \mathcal{W}$ and a depth that is a multiple of δ^4 , and contains only horizontal items with rounded width w . Each line along the y -axis (depth) through the bin intersects at most $O(1/\delta^3)$ horizontal containers. (iii) *Vertical containers*: each such container has a depth $d \in \mathcal{D}$, a height that is a multiple of δ^4 , and contains only vertical items with rounded depth d . Each line along the x -axis (width) through the bin intersects at most $O(1/\delta^3)$ vertical containers. (iv) *Tiny containers*: Contains only tiny items and have a width and depth that is a multiple of δ^4 . Each bin contains at most $O(1/\delta^3)$ of these containers. (v) *Intermediate containers*: These containers contain only intermediate items. There will be an extra $O(\varepsilon)\text{OPT}_{2D-BP}$ bins reserved separately to pack the intermediate items.

Let $\text{OPT}_{2D-BP}^C(\tilde{I})$ be the optimal 2D-container-packing for \tilde{I} . Jansen and Prädél [35] showed the existence of a good *2D-container-packing*.

► **Theorem 23** ([35]). *There exists a rounded up instance \tilde{I} from I_{2D-BP} such that the following holds: $\text{OPT}_{2D-BP}^C(\tilde{I}) \leq (\frac{3}{2} + O(\varepsilon))\text{OPT}_{2D-BP}(I) + O_\varepsilon(1)$. Further, \tilde{I} is one of $n^{O_\varepsilon(1)}$ possible roundings for the items I_{2D-BP} , and given a subset $I' \subseteq \tilde{I}$, it is possible to decide in $O(n) + O_\varepsilon(1)$ time if this set fits into a 2D-container-packing consisting of one bin.*

A 2D-bin configuration $C \in \mathbb{N}^{|\mathcal{T}|+|\mathcal{D}|+|\mathcal{W}|+1}$ is a vector that specifies a set of items that can be placed into one bin of a 2D-container-packing. This vector specifies for each big item type $t \in \mathcal{T}$, the number $n(t, C)$ of items of this type in the container; for each horizontal item width $w \in \mathcal{W}$, the number $D(w, C)$ where the total depth of containers for these items is $D(w, C) \cdot \delta^4$; for each vertical item depth $d \in \mathcal{D}$, the number $V(d, C)$ where the total width of containers for these items is $V(d, C) \cdot \delta^4$; and for tiny items, the number $S(C)$ where the total area of containers for tiny items is $S(C) \cdot \delta^8$. Let \mathcal{C} be the set of all 2D-bin configurations. Then $|\mathcal{C}|$ is $O_\varepsilon(1)$ as the sum of entries has an upper bound of $O(1/\delta^8)$. By Theorem 23, we can verify in polynomial-time if a given vector translates to a configuration C .

To find a 2D-container-packing, the algorithm from [35] utilizes an integer program, to find $\text{OPT}_{2D-BP}^C(\tilde{I})$. However, for our purposes, the linear programming (LP) relaxation of this program is sufficient. For each big item type $t \in \mathcal{T}$, let n_t denote the number of items with this type. Similarly, let V_d denote the total width of vertical items with rounded depth $d \in \mathcal{D}$, and D_w denote the total depth of horizontal items with rounded width $w \in \mathcal{W}$. Further, let S_{area} denote the total area of tiny items in I . We introduce variables x_C that denote for each configuration $C \in \mathcal{C}$, the *amount* of this configuration in the solution, e.g., if $x_C = 2.7$, the configuration C appears 2.7 times in the optimal LP solution. The LP is defined as follows.

$$\begin{aligned} \min \sum_{C \in \mathcal{C}} x_C & \quad \text{s.t.} \quad \sum_{C \in \mathcal{C}} D(w, C) \delta^4 x_C \geq D_w \quad \forall w \in \mathcal{W} \\ \sum_{C \in \mathcal{C}} n(t, C) x_C \geq n_t & \quad \forall t \in \mathcal{T} \quad \sum_{C \in \mathcal{C}} S(C) \delta^8 x_C \geq S_{\text{area}} \\ \sum_{C \in \mathcal{C}} V(d, C) \delta^4 x_C \geq V_d & \quad \forall d \in \mathcal{D} \quad x_C \geq 0 \quad \forall C \in \mathcal{C} \end{aligned}$$

Let $\text{OPT}_{2D-BP}^{\text{LP}}(\tilde{I})$ denote the optimal solution for the LP. A basic solution to this LP uses only $O_\varepsilon(1)$ different configurations, as there are only $O_\varepsilon(1)$ types of containers in total.

► **Lemma 24.** $\text{OPT}_{2D-BP}^{\text{LP}}(\tilde{I}) \leq \text{OPT}_{2D-BP}^C(\tilde{I})$ and in $n^{O_\varepsilon(1)}$ time a basic solution x with $\sum_{C \in \mathcal{C}} x_C = \text{OPT}_{2D-BP}^{\text{LP}}(\tilde{I})$ can be found.

Combining all the transformation steps (Lemma 21, Lemma 22, Theorem 23, and Lemma 24) to the 3D-BP instance I results in Corollary 25.

► **Corollary 25.** $\frac{\varepsilon v(I^\infty)}{n} \text{OPT}_{2D-BP}^{\text{LP}}(\tilde{I}) \leq ((3/2) + O(\varepsilon))\text{OPT}_{3D-SP}(I^\infty) + O_\varepsilon(1) \leq ((3/2)T_\infty + O(\varepsilon))\text{OPT}_{3D-BP}(I) + O_\varepsilon(1)$.

4.2 The algorithm for 3D-BP

In this subsection, we describe how the items are filled into the bins when a solution to the configuration LP is given. An overview of the complete algorithm can be found in Algorithm 1. For each 2D configuration C with $x_C > 0$, we create a 3D configuration of height $x_C \cdot \frac{\varepsilon v(I_{3D-BP}^\infty)}{n}$, whose base is identical to C . From Corollary 25, the sum of the heights of these configurations is at most $(3/2 + O(\varepsilon))\text{OPT}_{3D-SP}(I^\infty) + O_\varepsilon(1)$. The 2D containers of

■ **Algorithm 1** Asymptotic $((3/2)T_\infty + \varepsilon)$ -approximation for 3D-BP with input I and $\varepsilon > 0$.

-
1. Create I^∞ by harmonically rounding the item heights of I .
 2. Create I_{2D-BP} after rounding the heights of the items in I^∞ to multiples of $\frac{\varepsilon v(I^\infty)}{n}$.
 3. **for** Each feasible rounding \tilde{I} of the items I_{2D-BP} **do**
 - ┌ $x \leftarrow$ basic solution of the configuration LP corresponding to \tilde{I} .
 - └ Place I^∞ into 3D containers corresponding to x as discussed in Section 4.2.
 4. **return** the packing that uses the minimum number of bins.
-

C raised by the corresponding height of the configuration will be referred to as 3D containers, and will be denoted by the same type as the corresponding 2D containers. Our goal is to pack the items of I^∞ into these 3D containers while ensuring the *tall-not-sliced* property. We shall call the items with a rounded height larger than ε as *tall*, and the remaining items as *short*. The classification of items into big, vertical, horizontal, tiny, and intermediate depending to the dimensions of their top faces, as defined in the previous subsection continues to hold.

Placing big items. The different slices of a 3D big item might be rounded differently by the 2D-BP algorithm. However, using a linear program we can assign almost all 3D big items into the 3D big containers, without violating the container heights.

► **Lemma 26.** *In polynomial-time, it is possible to compute an assignment of all but $O_\varepsilon(1)$ big items into the 3D big containers, such that for each 3D big container, stacking the assigned items along the height does not exceed the height of the container.*

The $O_\varepsilon(1)$ unassigned big items will be packed using $O_\varepsilon(1)$ additional bins. We now pack the big items assigned to a container, while ensuring the *tall-not-sliced* property. Assume that the base of the container is aligned with the xy -plane at height 0. We sort the tall items in non-increasing order of their heights, and stack the items of height $1/q$ (where $q \in [1/\varepsilon]$) one above the other, starting at the next available z -coordinate that is a multiple of $1/q$. Following this, we stack the short items in an arbitrary order above the tall items. Then it is easy to see that the gaps between the tall items sum up to at most $\sum_{q \in [1/\varepsilon]} 1/q = O(\log(1/\varepsilon))$, and therefore by increasing the height of the container by $O(\log(1/\varepsilon))$, we can pack all the assigned big items ensuring *tall-not-sliced* property.

Placing horizontal items. Each 3D horizontal container is packed with horizontal items whose rounded width matches the container's width. To ensure that all horizontal items are packed, the container height is increased by $O(\log(1/\varepsilon))$, and the depth is extended by δ^4 . Since the container's width is exactly the item width, we simplify the placement by considering only the yz -plane, i.e., the right face of the container/items (see Figure 4). These 2D objects are then packed using a variant of the NFDH algorithm: shelves containing an item of height $1/q > \varepsilon$ are positioned at z -coordinates that are multiples of $1/q$. As a consequence, shelves containing a tall item never overlap an integer height. The total gaps between shelves sum to $O(\log(1/\varepsilon))$. Given the large enough height increase of $O(\log(1/\varepsilon))$, standard analysis of the NFDH algorithm ensures that all horizontal items fit into the containers (see Section A.2). Finally, the depth extension of the containers has to be reverted. Items that are (partly) placed inside the extended depth are repacked into $O(\delta)\text{OPT}_{3D-BP} + O_\varepsilon(1)$ containers by leveraging the fact that each line parallel to the y -axis intersects at most $1/\delta^3$ horizontal containers, and each horizontal item has a depth of at most δ^4 .

► **Lemma 27.** *Enlarging the height of the 3D horizontal containers by $O(\log(1/\varepsilon))$ allows all horizontal items to be placed into their containers and at most $O(\delta)\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1)$ additional bins while maintaining the tall-not-sliced property.*

The vertical items are packed analogously into the 3D vertical containers. The tiny items are packed using a variant of NFDH into the 3D tiny containers and few additional bins.

► **Lemma 28.** *By enlarging the 3D tiny containers by $O(\log(1/\varepsilon))$ along the height, it is possible to compute a packing of a subset of the tiny items ensuring the tall-not-sliced property. The remaining tiny items can be packed into $O(\varepsilon)\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1)$ additional bins.*

We use Theorem 7 to place the medium items into $O(\varepsilon)\text{OPT}_{3\text{D-BP}} + O(1)$ additional bins, as they have a volume of at most $O(\varepsilon)\text{OPT}_{2\text{D-BP}} \cdot \frac{\varepsilon v(I^\infty)}{n} \leq O(\varepsilon)\text{OPT}_{3\text{D-BP}}(I)$. Note that the heights of all the 3D containers were extended by $O(\log(1/\varepsilon))$ in order to obtain a *tall-not-sliced* packing. Since the number of configurations was $O_\varepsilon(1)$, the height of the packing only increases by $O_\varepsilon(1)$. Let H be the height of the stacked extended 3D-configurations returned by the algorithm. We create bins by cutting this packing at integer heights. The items sliced by this process all have a height of at most ε , and, hence, we can pack items sliced at $1/\varepsilon$ heights inside a single bin. Together with the $O(\varepsilon)\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1)$ additional bins used for packing the big, horizontal, vertical and tiny items, the number of bins needed is at most $(1 + \varepsilon)H + O(\varepsilon)\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1) \leq (3/2 + O(\varepsilon))\text{OPT}_{3\text{D-BP}}(I^\infty) + O(\varepsilon)\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1) \leq (3T_\infty/2 + O(\varepsilon))\text{OPT}_{3\text{D-BP}} + O_\varepsilon(1)$, establishing Theorem 5.

5 Algorithms for 3D-MVBB

In this section, we establish Theorem 4. Note that by the discussion in Section 3, Corollary 20 already implies an absolute approximation ratio of $11/2 + \varepsilon$ for 3D-MVBB, improving on the result of [43]. We now improve the approximation factor to $(3 + \varepsilon)$. First, as previously described, we guess the dimensions of the optimal bounding box within factors of $1 + \varepsilon$, and then scale the dimensions of each item by the corresponding (guessed) dimensions of the bounding box, so that all items now fit inside a $1 \times 1 \times 1$ bin.

Absolute approximation. We compute a value of μ such that the volume of the (scaled) items that have at least one of the dimensions in the range $(\mu^6, \mu]$ is bounded by ε , and classify the items into L, I_h, I_w, I_d , and I^{rem} as in Section 3. The items in I^{rem} are further classified into the sets $I_h^{\text{rem}}, I_w^{\text{rem}}$, and I_d^{rem} which can be packed into strips of thickness $O(\varepsilon)$ using the following lemma.

► **Lemma 29.** *The items of I_h^{rem} (resp. $I_w^{\text{rem}}, I_d^{\text{rem}}$) can be packed inside a strip with 1×1 base and height (resp. width, depth) at most 12ε .*

We now consider the packing of the items in $L \cup I_h$ and obtain the following result.

► **Proposition 30.** *In polynomial-time, we can compute a packing of $L \cup I_h$ inside a cubic bin of side length $1 + O(\varepsilon)$.*

The main idea behind the above result is to use Strip Packing with resource augmentation – if the width and depth of the strip are extended by a factor of $1 + O(\varepsilon)$, we can use a result of Bansal et al. [8] for 2D-BP, which returns a packing into the optimal number of bins where the sides of each bin are extended by a factor of $1 + O(\varepsilon)$. For this, we first discretize and guess the positions of all items in L (at most $O_\mu(1)$) inside the bin, and then obtain a set of

104:16 Improved Approximation Algorithms for Three-Dimensional Bin Packing

$O(1/\mu^4)$ 3D configurations that respect the positions of the items of L , into which items of I_h are packed fractionally. Finally, using the techniques described in Section 4, we can obtain an integral packing of I_h by increasing the height of each configuration by the maximum height μ^6 . This results in an overall increase in height of $\mu^6 \cdot O(1/\mu^4) = O(\mu^2) \leq \varepsilon$.

Proposition 30 also directly yields packings of I_w and I_d into bins of size $1 + O(\varepsilon)$. Together with Lemma 29, we obtain a packing of all input items into a bounding box of volume $3 + O(\varepsilon)$, establishing the absolute approximation guarantee of Theorem 4.

APTAS. In order to obtain an APTAS for 3D-MVBB, we notice that the technique described above yields an APTAS for 3D-SP when the base of the strip can be augmented by ε . We run this Strip Packing algorithm with the base of the strip aligned along the xy -, yz - and zx -planes, respectively, thus obtaining three bounding boxes, and return the one with the minimum volume. We show in the full version [39] that this gives an APTAS.

6 3D-BP with rotations

In this section, we present a 5-approximation for 3D-BP when the items are allowed to be rotated by 90° about any axis (Theorem 3).

Proof of Theorem 3. The work of Sharma [54] gave a $(T_\infty^2 + \varepsilon)$ -asymptotic approximation for 3D-BP with rotations, building on the algorithm of Caprara [12]. Let K be a large constant so that the algorithm of Sharma gives an absolute approximation ratio of 5 when $\text{OPT}_{3\text{D-BP}} > K$. From now on, we assume $\text{OPT}_{3\text{D-BP}} = k \leq K$, and that we have correctly guessed the value of k . Our goal is to obtain a packing of all items using at most $5k$ bins. Define $\mu = \frac{1}{12^4 K}$ and let L be the items having all dimensions exceeding μ , which we will refer to as the *large* items from now on.

We create $4k$ groups of items from $I \setminus L$, where each group has a volume of at most $1/4 - 2\mu$. Each such group can be packed into a single bin using Theorem 6 since $4(1/4 - 2\mu) + 8\mu = 1$. If all items of $I \setminus L$ are already packed, we pack the items of L into an additional k bins by brute-force enumeration, thus obtaining a packing of I into $5k$ bins, and are done. Otherwise, since the volume of each item in $I \setminus L$ is bounded by μ , the volume of the items packed into each bin is at least $1/4 - 3\mu$, and we have packed a volume of at least $(1 - 12\mu)k$. Let T denote the set of unpacked items of $I \setminus L$. Note that since $\text{OPT}_{3\text{D-BP}} = k$, we have $v(L \cup T) \leq 12\mu k$.

▷ **Claim 31.** One of the dimensions of each item of L does not exceed $1/12$.

Proof. Since $v(L) \leq 12\mu k \leq 1/12^3$, every large item must have a side of length at most $1/12$. ◁

We orient each large item so that its height is at most $1/12$. Using Theorem 6, the large items can be packed inside a strip with 1×1 base and height $4 \cdot 12\mu k + 8/12 < 3/4$. Finally, the items of T are also packed in a strip of height at most $4 \cdot 12\mu k + 8\mu < 1/4$ using Theorem 6. Thus the items in $L \cup T$ can all be packed inside a single bin. Overall, we obtain a packing of I into $4k + 1 \leq 5k$ bins, and are done. ◀

7 Conclusion

We obtained improved approximation for 3D-BP, 3D-SP, 3D-MVBB. Our framework is quite general and should extend to other cases. E.g., for the case with rotations, we expect that our techniques should be easily extendable to provide similar asymptotic guarantees.

We also expect that the asymptotic approximation algorithm for 3D-BP should extend to d -dimensions ($d > 3$) and provide a $3T_\infty^{d-2}/2$ -approximation. The existence of a PTAS (or hardness) for 3D-MVBB is still open. It is also interesting to obtain improved guarantees in pseudopolynomial-time (when the input numeric data is polynomially bounded in n).

References

- 1 ICRA virtual manufacturing automation competition. https://www.icra2013.org/index0c2d.html?page_id=231.
- 2 OPTIL.io. <https://www.optil.io/optilion/problem/3017>.
- 3 Sara Ali, António Galvão Ramos, Maria Antónia Carravilla, and José Fernando Oliveira. On-line three-dimensional packing problems: A review of off-line and on-line solution approaches. *Computers & Industrial Engineering*, 168:108–122, 2022.
- 4 Sam D Allen, Edmund K Burke, and Graham Kendall. A hybrid placement strategy for the three-dimensional strip packing problem. *European Journal of Operational Research*, 209(3):219–227, 2011. doi:10.1016/j.ejor.2010.09.023.
- 5 Helmut Alt. Computational aspects of packing problems. *Bulletin of EATCS*, 1(118), 2016.
- 6 Helmut Alt and Nadja Scharf. Approximating smallest containers for packing three-dimensional convex objects. *International Journal of Computational Geometry & Applications*, 28(2):111–128, 2018. doi:10.1142/S0218195918600026.
- 7 Nikhil Bansal, Alberto Caprara, and Maxim Sviridenko. A new approximation method for set covering problems, with applications to multidimensional bin packing. *SIAM Journal on Computing*, 39(4):1256–1278, 2010. doi:10.1137/080736831.
- 8 Nikhil Bansal, José R Correa, Claire Kenyon, and Maxim Sviridenko. 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.
- 9 Nikhil Bansal, Xin Han, Kazuo Iwama, Maxim Sviridenko, and Guochuan Zhang. A harmonic algorithm for the 3d strip packing problem. *SIAM Journal on Computing*, 42(2):579–592, 2013. doi:10.1137/070691607.
- 10 Nikhil Bansal and Arindam Khan. Improved approximation algorithm for two-dimensional bin packing. In *SODA*, pages 13–25, 2014. doi:10.1137/1.9781611973402.2.
- 11 Eberhard E Bischoff and Michael D Marriott. A comparative evaluation of heuristics for container loading. *European Journal of Operational Research*, 44(2):267–276, 1990.
- 12 Alberto Caprara. Packing d -dimensional bins in d stages. *Mathematics of Operations Research*, 33(1):203–215, 2008. doi:10.1287/moor.1070.0289.
- 13 Miroslav Chlebík and Janka Chlebíková. Hardness of approximation for orthogonal rectangle packing and covering problems. *Journal of Discrete Algorithms*, 7(3):291–305, 2009. doi:10.1016/j.jda.2009.02.002.
- 14 Henrik I Christensen, Arindam Khan, Sebastian Pokutta, and Prasad Tetali. 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.
- 15 Edward G Coffman, Jr, Michael R Garey, David S Johnson, and Robert Endre Tarjan. Performance bounds for level-oriented two-dimensional packing algorithms. *SIAM Journal on Computing*, 9(4):808–826, 1980. doi:10.1137/0209062.
- 16 Teodor Gabriel Crainic, Guido Perboli, and Roberto Tadei. Extreme point-based heuristics for three-dimensional bin packing. *INFORMS Journal on Computing*, 20(3):368–384, 2008. doi:10.1287/ijoc.1070.0250.
- 17 Teodor Gabriel Crainic, Guido Perboli, and Roberto Tadei. Ts2pack: A two-level tabu search for the three-dimensional bin packing problem. *European Journal of Operational Research*, 195(3):744–760, 2009. doi:10.1016/j.ejor.2007.06.063.
- 18 János Csirik and André Van Vliet. An on-line algorithm for multidimensional bin packing. *Operations Research Letters*, 13(3):149–158, 1993. doi:10.1016/0167-6377(93)90004-Z.

- 19 Florian Diedrich, Rolf Harren, Klaus Jansen, Ralf Thöle, and Henning Thomas. Approximation algorithms for 3d orthogonal knapsack. *Journal of Computer Science and Technology*, 23(5):749–762, 2008. doi:10.1007/s11390-008-9170-7.
- 20 Thai Ha Duong. *Heuristics approaches for three-dimensional strip packing and multiple carrier transportation plans*. PhD thesis, University of Nottingham, 2015.
- 21 Leah Epstein and Rob van Stee. Optimal online bounded space multidimensional packing. In *SODA*, pages 214–223, 2004. URL: <http://dl.acm.org/citation.cfm?id=982792.982823>.
- 22 Oluf Faroe, David Pisinger, and Martin Zachariassen. Guided local search for the three-dimensional bin-packing problem. *INFORMS journal on Computing*, 15(3):267–283, 2003. doi:10.1287/ijoc.15.3.267.16080.
- 23 Carlos E Ferreira, Flavio K Miyazawa, and Yoshiko Wakabayashi. Packing squares into squares. *Pesquisa Operacional*, 19(2):223–237, 1999.
- 24 Lisa Fleischer, Michel X Goemans, Vahab S Mirrokni, and Maxim Sviridenko. Tight approximation algorithms for maximum separable assignment problems. *Mathematics of Operations Research*, 36(3):416–431, 2011. doi:10.1287/moor.1110.0499.
- 25 Waldo Gálvez, Fabrizio Grandoni, Salvatore Ingala, Sandy Heydrich, Arindam Khan, and Andreas Wiese. Approximating geometric knapsack via L-packings. *ACM Transactions on Algorithms*, 17(4):1–67, 2021. doi:10.1145/3473713.
- 26 John A George and David F Robinson. A heuristic for packing boxes into a container. *Computers & Operations Research*, 7(3):147–156, 1980. doi:10.1016/0305-0548(80)90001-5.
- 27 Paul C Gilmore and Ralph E Gomory. Multistage cutting stock problems of two and more dimensions. *Operations Research*, 13(1):94–120, 1965.
- 28 Rolf Harren. Approximation algorithms for orthogonal packing problems for hypercubes. *Theoretical Computer Science*, 410(44):4504–4532, 2009. doi:10.1016/j.tcs.2009.07.030.
- 29 Rolf Harren, Klaus Jansen, Lars Prädél, and Rob Van Stee. A $(5/3 + \varepsilon)$ -approximation for strip packing. *Computational Geometry*, 47(2):248–267, 2014. doi:10.1016/j.comgeo.2013.08.008.
- 30 Rolf Harren, Klaus Jansen, Lars Praedel, Ulrich M Schwarz, and Rob van Stee. Two for one: tight approximation of 2d bin packing. *International Journal of Foundations of Computer Science*, 24(08):1299–1327, 2013. doi:10.1142/S0129054113500354.
- 31 Sören Henning, Klaus Jansen, Malin Rau, and Lars Schmarje. 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.
- 32 Mhand Hifi, Imed Kacem, Stéphane Nègre, and Lei Wu. A linear programming approach for the three-dimensional bin-packing problem. *Electronic Notes in Discrete Mathematics*, 36:993–1000, 2010. doi:10.1016/j.endm.2010.05.126.
- 33 Klaus Jansen, Arindam Khan, Marvin Lira, and K. V. N. Sreenivas. A PTAS for packing hypercubes into a knapsack. In *ICALP*, pages 78:1–78:20, 2022. doi:10.4230/LIPIcs.ICALP.2022.78.
- 34 Klaus Jansen and Lars Prädél. A new asymptotic approximation algorithm for 3-dimensional strip packing. In *SOFSEM*, pages 327–338, 2014. doi:10.1007/978-3-319-04298-5_29.
- 35 Klaus Jansen and Lars Prädél. New approximability results for two-dimensional bin packing. *Algorithmica*, 74:208–269, 2016. doi:10.1007/s00453-014-9943-z.
- 36 Klaus Jansen and Malin Rau. Closing the gap for pseudo-polynomial strip packing. In *ESA*, pages 62:1–62:14, 2019. doi:10.4230/LIPIcs.ESA.2019.62.
- 37 Klaus Jansen and Roberto Solis-Oba. An asymptotic approximation algorithm for 3 d-strip packing. In *SODA*, pages 143–152, 2006. URL: <http://dl.acm.org/citation.cfm?id=1109557.1109575>.
- 38 Zhihong Jin, Takahiro Ito, and Katsuhisa Ohno. The three-dimensional bin packing problem and its practical algorithm. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 46(1):60–66, 2003.

- 39 Debajyoti Kar, Arindam Khan, and Malin Rau. Improved approximation algorithms for three-dimensional bin packing. *arXiv preprint arXiv:2503.08863*, 2025. doi:10.48550/arXiv.2503.08863.
- 40 Claire Kenyon and Eric Rémila. 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.
- 41 C. C. Lee and D. T. Lee. A simple on-line bin-packing algorithm. *Journal of the ACM*, 32(3):562–572, 1985. doi:10.1145/3828.3833.
- 42 Keoin Li and Kam Hoi Cheng. Heuristic algorithms for on-line packing in three dimensions. *Journal of Algorithms*, 13(4):589–605, 1992. doi:10.1016/0196-6774(92)90058-K.
- 43 Keqin Li and Kam-Hoi Cheng. On three-dimensional packing. *SIAM Journal on Computing*, 19(5):847–867, 1990. doi:10.1137/0219059.
- 44 Xueping Li, Zhaoxia Zhao, and Kaike Zhang. A genetic algorithm for the three-dimensional bin packing problem with heterogeneous bins. In *IIE Annual Conference*, page 2039, 2014.
- 45 Andrea Lodi, Silvano Martello, and Daniele Vigo. Heuristic algorithms for the three-dimensional bin packing problem. *European Journal of Operational Research*, 141(2):410–420, 2002. doi:10.1016/S0377-2217(02)00134-0.
- 46 Yiping Lu, Danny Z Chen, and Jianzhong Cha. Packing cubes into a cube is np-complete in the strong sense. *Journal of Combinatorial Optimization*, 29:197–215, 2015. doi:10.1007/s10878-013-9701-1.
- 47 Daniel Mack and Andreas Bortfeldt. A heuristic for solving large bin packing problems in two and three dimensions. *Central European Journal of Operations Research*, 20:337–354, 2012. doi:10.1007/s10100-010-0184-1.
- 48 Batoul Mahvash, Anjali Awasthi, and Satyaveer Chauhan. A column generation-based heuristic for the three-dimensional bin packing problem with rotation. *Journal of the Operational Research Society*, 69(1):78–90, 2018. doi:10.1057/s41274-017-0186-7.
- 49 Silvano Martello, David Pisinger, and Daniele Vigo. The three-dimensional bin packing problem. *Operations Research*, 48(2):256–267, 2000. doi:10.1287/opre.48.2.256.12386.
- 50 Aram Meir and Leo Moser. On packing of squares and cubes. *Journal of Combinatorial Theory*, 5(2):126–134, 1968.
- 51 Flavio Keidi Miyazawa and Yoshiko Wakabayashi. An algorithm for the three-dimensional packing problem with asymptotic performance analysis. *Algorithmica*, 18(1):122–144, 1997. doi:10.1007/BF02523692.
- 52 Flavio Keidi Miyazawa and Yoshiko Wakabayashi. Packing problems with orthogonal rotations. In *LATIN*, pages 359–368, 2004.
- 53 Francisco Parreño, Ramón Alvarez-Valdés, José Fernando Oliveira, and José Manuel Tamarit. A hybrid grasp/vnd algorithm for two-and three-dimensional bin packing. *Annals of Operations Research*, 179:203–220, 2010. doi:10.1007/s10479-008-0449-4.
- 54 Eklavya Sharma. Harmonic algorithms for packing d-dimensional cuboids into bins. In *FSTTCS*, pages 32:1–32:22, 2021. doi:10.4230/LIPIcs.FSTTCS.2021.32.
- 55 Tony Wauters, Jannes Verstichel, and Greet Vanden Berghe. An effective shaking procedure for 2d and 3d strip packing problems. *Computers & Operations Research*, 40(11):2662–2669, 2013. doi:10.1016/j.cor.2013.05.017.
- 56 Lijun Wei, Wee-Chong Oon, Wenbin Zhu, and Andrew Lim. A reference length approach for the 3d strip packing problem. *European Journal of Operational Research*, 220(1):37–47, 2012. doi:10.1016/j.ejor.2012.01.039.

A Tools and techniques

A.1 Harmonic transformation

Lee and Lee [41] introduced harmonic transformation in the context of online bin packing. The harmonic transformation with parameter k is defined by the following function f_k : For $\alpha_i \in (1/(q+1), 1/q]$, $f_k(\alpha_i) := 1/q$, for $q \in [k-1]$. Otherwise, $f_k(\alpha_i) := \frac{k}{k-1}\alpha_i$. Intuitively, the function f_k rounds up $\alpha_i \in (1/k, 1]$ to the nearest larger number of the form $1/q$ where $q \in \mathbb{Z}$.

Now we define harmonic constant T_∞ . Let $t_1 := 1$ and $t_{i+1} = t_i(t_i + 1)$ for $i \in \mathbb{Z}_{\geq 2}$. The sequence $t_i + 1$ is also known as Sylvester's sequence (where each term is the product of the previous terms, plus one). Let $m(k)$ be the integer such that $t_{m(k)} \leq k \leq t_{m(k)+1}$. Now T_k is defined as $\sum_{q=1}^{m(k)} \frac{1}{t_q} + \frac{k}{t_{(m(k)+1) \cdot (k-1)}}$, and $T_\infty := \lim_{k \rightarrow \infty} T_k$. Thus $T_\infty = \sum_{i=1}^{\infty} \frac{1}{t_i} = 1 + \frac{1}{2} + \frac{1}{6} + \dots \approx 1.69103$. Note that $T_k \leq T_\infty + \frac{1}{(k-1)}$.

Lee and Lee [41] showed that that, for any sequence $\alpha_1, \alpha_2, \dots, \alpha_n$, with $\alpha_i \in (0, 1]$ and $\sum_{i=1}^n \alpha_i \leq 1$, we have $\sum_{i=1}^n f_k(\alpha_i) \leq T_k$. In fact, $\lim_{k \rightarrow \infty} \sum_{i=1}^n f_k(\alpha_i) \leq T_\infty \approx 1.691$.

In fact, Bansal, Han, Iwama, Sviridenko, and Zhang [9] showed the above inequality is true even if we define $f_k(\alpha_i) := \alpha_i$ for $\alpha_i \leq 1/k$.

A.2 Next-Fit-Decreasing-Height (NFDH)

The Next-Fit-Decreasing-Height (NFDH) algorithm is a shelf-based approach for packing 2D items into a strip of fixed width w . Given a set of items I , the algorithm first sorts them in decreasing order of height. It then places the items sequentially from left to right on the floor of the strip (or the current shelf) until the next item no longer fits, i.e., adding the next item would cause the total width of items on the shelf to exceed w . At this point, a new shelf is created by drawing a horizontal line at the height of the tallest item on the current shelf. The process then continues on the new shelf, following the same placement rule, until all items have been packed. For more details on the algorithm, we refer to [15, 14].

► **Lemma 32** ([15]). *Given a 2D rectangular box of height h and width w , and a set of 2D items with maximum height h_{\max} and maximum width w_{\max} , it is possible to place any subset of items with a total area of at most $(h - h_{\max})(w - w_{\max})$, into the box using NFDH.*

A.3 Generalized Assignment Problem (GAP)

In the Generalized Assignment Problem, we are given a set of k knapsacks each with an associated capacity $\{c_j\}_{j \in [k]}$, and a set of n items, where each item $i \in [n]$ has size s_{ij} and profit p_{ij} for knapsack j . The goal is to obtain a maximum-profitable packing of a subset of the items into the knapsacks, that respects the knapsack capacities, i.e., the total size of the items packed inside each knapsack do not exceed the capacity of the knapsack. For the general case of GAP, there is a tight $(\frac{e}{e-1} + \varepsilon)$ -approximation [24]. However for the special case when $k = O(1)$, there exists a PTAS.

► **Theorem 33** ([25]). *For any $\varepsilon > 0$, there is an algorithm for GAP with k knapsacks running in $n^{O(k/\varepsilon^2)}$ time, that returns a packing of profit at least $(1 - \varepsilon)\text{OPT}$.*