

# Approximation Algorithms for Orthogonal Packing Problems for Hypercubes

Rolf Harren<sup>1</sup>

*Max-Planck-Institut für Informatik,  
Campus E 1 4, 66123 Saarbrücken, Germany*

---

## Abstract

Orthogonal packing problems are natural multidimensional generalizations of the classical BIN PACKING PROBLEM and KNAPSACK PROBLEM and occur in many different settings. The input consists of a set  $I = \{r_1, \dots, r_n\}$  of  $d$ -dimensional rectangular items  $r_i = (a_{i,1}, \dots, a_{i,d})$  and a space  $Q$ . The task is to pack the items in an orthogonal and non-overlapping manner without using rotations into the given space. In the STRIP PACKING setting the space  $Q$  is given by a strip of bounded basis and unlimited height. The objective is to pack all items into a strip of minimal height. In the KNAPSACK PACKING setting the given space  $Q$  is a single, usually unit sized bin and the items have associated profits  $p_i$ . The goal is to maximize the profit of a selection of items that can be packed into the bin.

We mainly focus on orthogonal KNAPSACK PACKING restricted to hypercubes and our main results are a  $(5/4 + \epsilon)$ -approximation algorithm for two-dimensional HYPERCUBE KNAPSACK PACKING, also known as SQUARE PACKING, and a  $(1 + 1/2^d + \epsilon)$ -approximation algorithm for  $d$ -dimensional HYPERCUBE KNAPSACK PACKING. In addition we consider  $d$ -dimensional HYPERCUBE STRIP PACKING in the case of a bounded ratio between the shortest and longest side of the basis of the strip. We derive an asymptotic polynomial time approximation scheme (*APTAS*) for this problem. Finally, we present an algorithm that packs hypercubes with a total profit of at least  $(1 - \epsilon)\text{OPT}$  into a large bin (the size of the bin depends on  $\epsilon$ ). This problem is known as HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES. A preliminary version was published in [15] but especially for the latter two approximation schemes no details were given due to page limitations.

*Key words:* Orthogonal Knapsack Problem, Square Packing, Hypercube Packing

---

---

*Email address:* rharren@mpi-inf.mpg.de (Rolf Harren).

<sup>1</sup> The results of this work were mainly achieved while visiting the Graduate School of Informatics, Kyoto University, Japan, under a grant by DAAD (German Academic Exchange Service)

## 1 Introduction

The classical BIN PACKING PROBLEM and KNAPSACK PROBLEM are among the most fundamental optimization problems in computer science. The one-dimensional variants and their applications are subject to a great number of articles, see [8] and [24,29]. Not surprisingly, different geometrical generalizations are also very popular. They are defined as follows.

Given a set  $I = \{r_1, \dots, r_n\}$  of rectangular items  $r_i = (a_{i,1}, a_{i,2}, \dots, a_{i,d})$  of side length  $a_{i,j} \leq 1$  and a rectangular space  $Q$ . The objective is to find a *feasible*, i.e., *orthogonal* and *non-overlapping* packing *without using rotations* into  $Q$ . An *orthogonal* packing requires that the items are packed parallel to the axis of the space. Items are *non-overlapping* if their interiors are disjoint. We distinguish three different objectives as the main variants of packing problems. In the BIN PACKING setting the space  $Q$  is an unlimited number of unit sized bins and the goal is to minimize the number of used bins in order to pack the whole list of items. In the STRIP PACKING setting the space  $Q$  is given by a strip of bounded basis and unlimited height. The objective is to pack all items into a strip of minimal height. Finally, in the KNAPSACK PACKING setting the given space  $Q$  is a single, usually unit sized bin and the items have associated profits  $p_i$ . The goal is to maximize the profit of a selection of items that can be packed into the bin.

Throughout this paper we consider the restriction to *squares*, *cubes* and *hypercubes*. We therefore denote the restricted version of the two-dimensional KNAPSACK PROBLEM, which is commonly known as RECTANGLE PACKING, by SQUARE PACKING. Note that this restriction is quite popular in the literature and yields great potential. Bansal, Correa, Kenyon & Sviridenko [2] showed for two-dimensional BIN PACKING that even though it is  $\mathcal{APX}$ -complete in the general case the restriction to squares admits an asymptotic polynomial time approximation scheme ( $\mathcal{APTAS}$ ). Furthermore, their results hold for higher dimensions as well. We now give a comprehensive survey of the known results on the main variants of orthogonal packing problems.

**Related work.** In 1990, Leung et al. [27] proved the  $\mathcal{NP}$ -hardness in the strong sense for the special case of determining whether a set of squares can be packed into a bigger square. Therefore already a very special two-dimensional case and all generalizations are strongly  $\mathcal{NP}$ -hard. In contrast, the  $\mathcal{NP}$ -hardness of *hypercube* BIN PACKING, STRIP PACKING and KNAPSACK PACKING are still open problems for  $d > 2$ .

*Bin Packing.* The classical one-dimensional BIN PACKING PROBLEM has been

studied in great detail. It is known to be  $\mathcal{NP}$ -complete and there is no polynomial time  $(3/2 - \epsilon)$ -approximation algorithm for any  $\epsilon > 0$ , unless  $\mathcal{P} = \mathcal{NP}$ . In contrast, we can achieve much better *asymptotic* results as there is an  $\mathcal{AFPTAS}$  [23].

As already mentioned, Bansal et al. showed the  $\mathcal{APX}$ -completeness for two-dimensional BIN PACKING [2]. Chlebík & Chlebíková [7] gave explicit lower bounds on the asymptotic approximability of  $1/2196$  for two-dimensional BIN PACKING without rotations and of  $1/3792$  for two-dimensional BIN PACKING with rotations of  $90^\circ$ . Caprara was the first to present an algorithm with an asymptotic approximation ratio less than 2 for BIN PACKING without rotations. Indeed, he considered 2-stage packing, in which the items must first be packed into shelves that are then packed into bins, and showed that the asymptotic worst case ratio between two-dimensional BIN PACKING and 2-stage packing is  $T_\infty = 1.691\dots$ . Therefore the asymptotic  $\mathcal{FPTAS}$  for 2-stage packing from Caprara, Lodi & Monaci [5] achieves an asymptotic approximation guarantee arbitrary close to  $T_\infty$ . Recently Bansal, Caprara & Sviridenko [1] presented a general framework to improve subset oblivious approximation algorithms and obtained asymptotic approximation guarantees arbitrarily close to  $1.525\dots$  for packing with or without rotations. These are the currently best-known approximation ratios for these problems. Finally, Bansal et al. considered the restriction to hypercubes and derived an  $\mathcal{APTAS}$  for  $d$ -dimensional HYPERCUBE BIN PACKING [2].

*Strip Packing.* Two-dimensional STRIP PACKING turns out to be relatively easy. Kenyon & Rémila derived an  $\mathcal{AFPTAS}$  for the problem without rotations [25] and Jansen & van Stee gave an  $\mathcal{AFPTAS}$  for the variant where rotations are allowed [20]. The additive constants for these  $\mathcal{AFPTAS}$ s was recently improved by Jansen & Solis-Oba [18] from  $\mathcal{O}(1/\epsilon^2)$  to 1.

In contrast to this, already the three-dimensional STRIP PACKING is  $\mathcal{APX}$ -complete since it includes two-dimensional BIN PACKING as a subproblem. Jansen & Solis-Oba [17] gave a  $(2 + \epsilon)$ -approximation algorithm which was improved independently by Bansal & Sviridenko as well as by Han, Iwama & Zhang to yield an approximation ratio of  $1.691\dots$  [3]. In their joint publication they also give an outline for an  $\mathcal{APTAS}$  for three-dimensional STRIP PACKING for items with square basis that is similar to the one presented in this work in Section 4 and that was previously mentioned in [15].

*Knapsack Packing.* While the extensively studied one-dimensional KNAPSACK PACKING is known to be only weakly  $\mathcal{NP}$ -complete and admits an  $\mathcal{FPTAS}$  [16], the  $d$ -dimensional KNAPSACK PACKING seems to be much more challenging. Caprara & Monaci [6] gave the first approximation algorithm with an approximation ratio of  $3 + \epsilon$  for RECTANGLE PACKING. Jansen & Zhang [22] showed the currently best-known approximation ratio of  $2 + \epsilon$ . In [18], Jansen & Solis-

Oba describe an  $\epsilon$ -augmenting algorithm that finds a packing with profit at least  $(1 - \epsilon)\text{OPT}$  into a rectangle of size  $(1, 1 + \epsilon)$ , where  $\text{OPT}$  denotes the optimal value for packing into a unit bin. This improves upon a previous result from Fishkin, Gerber, Jansen & Solis-Oba [12] where augmentation in both directions was needed.

There are a number of special cases that admit a better approximation. Maximizing the number of packed items, i.e., all items have uniform profit, admits an  $\mathcal{AP}TAS$  as Jansen & Zhang showed [21]. For packing squares, Fishkin, Gerber, Jansen & Solis-Oba showed that if the profit of each item is equal to its area, i.e., the packed area is to be maximized, a  $\mathcal{PTAS}$  is possible [13]. In 2008, Jansen & Solis-Oba settled the approximability of SQUARE PACKING with arbitrary profits with their presentation of a  $\mathcal{PTAS}$  [19]. Finally, there is a result on so-called RECTANGLE PACKING WITH LARGE RESOURCES that is crucial for this work: Given a large bin, a much better approximation ratio can be achieved. Fishkin, Gerber & Jansen [11] showed that a  $(1 - 72\epsilon)$ -optimal solution can be found in time polynomial in  $n$  and  $1/\epsilon$  if one side length of the bin is at least  $1/\epsilon^3$  times larger than the items for any  $0 < \epsilon < 1/72$ .

Just recently the first results on three-dimensional KNAPSACK PACKING have been published. Chlebík & Chlebíková [7] showed that no  $\mathcal{AP}TAS$  for maximizing the number of items, i.e., all items have unit profit, exists unless  $\mathcal{P}=\mathcal{NP}$ . The best general positive result is a  $(7 + \epsilon)$ -approximation algorithm described by Diedrich, Harren, Jansen, Thöle & Thomas [10].

**Our contribution.** Our main result is an approximation algorithm for SQUARE PACKING with an approximation ratio of  $(5/4 + \epsilon)$ . Moreover, we show that this algorithm can be generalized to  $d$ -dimensional HYPERCUBE KNAPSACK PACKING, yielding a  $(1 + 1/2^d + \epsilon)$ -approximation algorithm. In order to generalize our result to higher dimensions, we derive an  $\mathcal{AP}TAS$  for  $d$ -dimensional HYPERCUBE STRIP PACKING in the case of a bounded ratio between the shortest and longest side of the basis of the strip and a result similar to [11] for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES, i.e., packing into a bin that is much larger than the items. This algorithm is not a  $\mathcal{AP}TAS$  since the required size of the bin depends on the desired accuracy of the algorithm. Both results are motivated by their two-dimensional equivalents in [25] and [11], respectively, and thus also stand for themselves. A preliminary version of our work was published in [15].

Note that the  $\mathcal{PTAS}$  from Jansen & Solis-Oba [19] is a recent improvement upon our two-dimensional algorithm. On the other hand already our contribution shows that the best-known approximation ratios for the general case can be significantly improved for the restriction to hypercubes. Furthermore, our approximation ratio improves exponentially with the dimension.

**Applications.** There are a number of important practical applications for the presented problems. A very important industrial application is the CUTTING STOCK PROBLEM. A set of items has to be cut out of a given material. Obviously cutting problems and packing problems correspond to each other. Restrictions to orthogonal packing and packing without rotations make sense in this setting as well if we cut items out of patterned fabric and have to retain the alignment of the pattern. CUTTING STOCK PROBLEMS occur as BIN PACKING, STRIP PACKING or KNAPSACK PACKING PROBLEMS in different dimensions.

The two-dimensional variants of the BIN PACKING PROBLEM and KNAPSACK PROBLEM as well as the three-dimensional KNAPSACK PROBLEM and STRIP PACKING PROBLEM can be understood as an ADVERTISEMENT PLACEMENT PROBLEM [14]. In this problem we have to place given rectangular ads on a certain page or web-page. In the two-dimensional KNAPSACK PACKING setting, this would mean that we have to maximize the profit for a single page. In contrast to this, the BIN PACKING setting requires to place all ads on a minimal number of pages. The third dimension of the KNAPSACK PACKING and STRIP PACKING can be seen as the time on animated flash pages. Here we either have to maximize the profit for a given time slot or to minimize the total time to display all ads.

Further applications can be found in VLSI-DESIGN (MINIMUM RECTANGLE PLACEMENT PROBLEM [2]) and in SCHEDULING ON PARTITIONABLE RESOURCES [28].

**Presentation of the paper.** We begin with some preliminaries in Section 2 and describe our  $(5/4 + \epsilon)$ -approximation algorithm for SQUARE PACKING in detail in Section 3. Before the presentation of the generalization in Section 6 we give an *APTAS* for HYPERCUBE STRIP PACKING in the case of a bounded ratio between the shortest and longest side of the basis of the strip in Section 4 and our result on HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES in Section 5.

## 2 Notations and Preliminaries

Let us start our description with some notations and lemmas. Since the items are hypercubes throughout the paper, we refer to both the items and their sizes, i.e., their side lengths, by  $a_i$ . Let  $I = \{a_1, \dots, a_n\}$  be a set of items. We associate a rectangular space  $Q$  of size  $(q_1, \dots, q_d)$  with  $[0, q_1] \times \dots \times [0, q_d] \subset \mathbb{R}^d$ . A packing  $P$  of  $I$  into  $Q$  is given by the position  $(x_1^i, \dots, x_d^i)$  of the lower

left corner of each item  $a_i \in I$ . The item  $a_i$  is therefore associated with  $[x_1^i, x_1^i + a_i] \times \dots \times [x_d^i, x_d^i + a_i] \subset \mathbb{R}^d$ . We call  $I$  *feasible* if there exists a packing into  $Q$  such that all items are completely contained in  $Q$  and the interior of any two items is disjoint. We denote the volume of  $I$  by  $\text{Vol}(I) = \sum_{i \in I} a_i^d$ . Note that we refer to the area of two-dimensional items by volume as well in order to have a consistent notation.

The height of a packing in the STRIP PACKING setting is denoted by  $h(I)$ , whereas the optimal, i.e., minimal height, is denoted by  $\text{OPT}(I)$ . In the KNAPSACK setting each item has an associated profit  $p_i$  and we denote the profit of a set of items  $I$  by  $p(I) = \sum_{i \in I} p_i$  and the profit of an optimal solution by  $\text{OPT}(I)$ . Usually it will be clear from the context whether we are in the KNAPSACK PACKING or in the STRIP PACKING setting. To avoid confusion, we refer to the optimal height with  $\text{OPT}_{\text{strip}}(I)$  in some cases.

The algorithm that we present in Section 3 to 6 are based on a separation of the input into sets of large and small items. The large items are packed optimally or almost optimally using enumerations and linear programming relaxations whereas the small items are added afterwards depending on the structure of the packing of the large items. In the following we derive some useful lemmas to enumerate packings of the large items and to handle the structure of the free space in a packing of large items.

Suppose the set  $I$  contains large items only, say,  $a_i \geq \delta$  and there are only  $k$  different sizes for some constant  $k$ . Bansal et al. [2] showed how to check the *feasibility* of  $I$  in constant time. In particular, they used this method to generalize the  $\mathcal{O}(\text{polylog}(n))$  time algorithm from de la Vega & Luecker [9] for one-dimensional BIN PACKING on such instances. In the following, we use a similar technique to derive a method to check the feasibility of a constant number of items of arbitrary sizes. We refer to this method as *constant packing*. Our method and the method by Bansal et al. can easily be generalized to rectangular  $d$ -dimensional items instead of hypercubes.

**Lemma 1. Constant packing**

*Given a constant number of items  $I = \{a_1, \dots, a_k\}$ , then we can check the feasibility into a rectangular space  $Q = (q_1, q_2, \dots, q_d)$  in constant time  $\mathcal{O}(2^{dk^2} \cdot dk^2)$ .*

*Proof.* Let  $P$  be a packing of  $I$ . Consider the directions  $i \in \{1, \dots, d\}$  successively, sort the items by ascending  $x_i$ -coordinate, and move them into the direction of the origin until they hit the boundary of the bin or another item. It is easy to see that the  $x_i$ -coordinates of all items are in the set

$$\mathbf{e}^i = \left\{ \sum_{j=1}^k \lambda_j a_j \mid \lambda_j \in \{0, 1\} \text{ for } j = 1, \dots, k \text{ and } \sum_{j=1}^k \lambda_j a_j \leq b_i \right\}.$$

The cardinality of  $\mathfrak{C}^i$  is at most  $2^k$ . Enumerating over all assignments requires at most  $|\mathfrak{C}^1|^k \cdot \dots \cdot |\mathfrak{C}^d|^k \leq ((2^k)^k)^d = (2^k)^{dk} = 2^{dk^2}$  steps. Checking the feasibility for a given assignment requires time  $\mathcal{O}(dk^2)$  by verifying pairwise disjointness and complete containment in the bin.  $\square$

Note that it might be possible to further shift an item  $a_\ell$  in some direction  $i$  in a packing derived by Lemma 1 as an item  $a'$  that blocks  $a_\ell$  in direction  $i$  might be shifted away in another direction  $j > i$ . However, this is irrelevant since in the final packing the  $x_i$ -coordinate of  $a_\ell$  is in  $\mathfrak{C}^i$ .

In order to add small items into a packing of large items we consider the free space in such a packing. Bansal et al. [2] showed the following lemmas to restrict the number of gaps in a packing of a constant number of items.

**Lemma 2. Gaps in a packing [2]**

*Let  $P$  be a packing of  $k$  hypercubes into a cuboid  $Q = (q_1, \dots, q_d)$  such that there is a hypercube touching each of the hyperplanes  $x_i = 0$  for  $i = 1, \dots, d$ . Then the free space in  $Q$  can be partitioned into at most  $(2k)^d$  non-overlapping cuboids in time  $\mathcal{O}(d(2k)^d \log k)$ .*

The partition is done by extending the faces of all items to define a grid with at most  $(2k)^d$  cells. Using interval search we can check in time  $\mathcal{O}(d \log k)$  whether a cell is free.

Note that the *constant packing* method creates suitable packings where each hyperplane  $x_i = 0$  for  $i = 1, \dots, d$  is touched by a hypercube if the enumeration is done in a suitable order.

A stronger statement can be made for the two-dimensional case.

**Lemma 3. Gaps in a two-dimensional packing [2]**

*Let  $P$  be a packing of  $k$  squares into a rectangle  $Q = (q_1, q_2)$  such that there is a square aligned with the bottom line of  $Q$ . Then the free space in  $Q$  can be partitioned into at most  $3k$  rectangular gaps in time  $\mathcal{O}(k \log k)$ .*

This can easily be seen by extending the upper and lower face of each item in the packing and associating each item with the gap above and a gap to the left and to the right—see Figure 1.

Having derived a partition of the free space in a packing into rectangular gaps we use the NEXT FIT DECREASING HEIGHT (NFDH)-algorithm that Meir & Moser [30] described already in 1968 to add small items into these gaps. Their work was generalized by Bansal et al. [2] for  $d$ -dimensional packing. The two-dimensional NFDH, i.e., packing squares into a rectangle  $Q = (q_1, q_2)$ , is given as follows—see Figure 2 for an illustration.

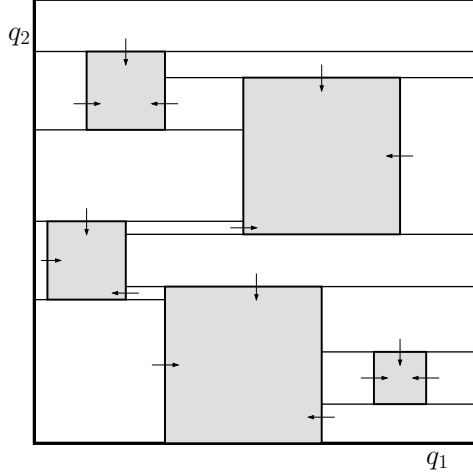


Fig. 1. Partition of the free space into at most  $3k$  gaps. The arrows indicate which gaps are associated to each item.

Sort the items by non-increasing order of size  $a_1 \geq a_2 \geq \dots \geq a_n$ . Stop if  $a_1 > q_1$  or  $a_1 > q_2$ . Otherwise define a layer  $L_1$  by drawing a line at distance  $a_1$  from the bottom line of  $R$  and fill items into  $L_1$ . To do this pack the items successively left-aligned with the preceding item on the bottom line of the layer. When an item  $a_{k_1}$  does not fit into  $L_1$  create a new layer  $L_2$  by drawing a line at distance  $a_{k_1}$  above  $L_1$  and continuing the packing inside  $L_2$ . The algorithm stops if all items are packed or a new line has to be drawn outside  $Q$ .

For  $d$ -dimensional packing into  $Q = (q_1, \dots, q_d)$  we assume that we know how to perform NFDH for  $d - 1$  dimensions. Sort the items again by non-increasing order of size  $a_1 \geq a_2 \geq \dots \geq a_n$ . Stop if  $a_1 > q_i$  for any  $i \in \{1, \dots, d\}$ . Otherwise define a layer  $L_1$  of height  $a_1$  and basis  $(q_1, \dots, q_{d-1})$  inside  $Q$ . Use the  $(d - 1)$ -dimensional NFDH to pack the  $d - 1$  projections of the items into the basis of  $L_1$ . As the height of  $L_1$  is  $a_1$ , this packing can be transformed to a packing of the items into  $L_1$ . When an item  $a_{k_1}$  does not fit into  $L_1$  create a new layer  $L_2$  of height  $a_{k_1}$  above  $L_1$  and continue the packing inside  $L_2$ . The algorithm stops if all items are packed or a new line has to be drawn outside  $Q$ . As the main steps of the algorithm are sorting the items and checking for each item and each dimension whether the item fits into the current layer, we get a running time of  $\mathcal{O}(n \log n + dn)$ .

To apply NFDH on several gaps we consider the gaps one after the other and for each gap we apply NFDH on the items that are not packed into one of the previous gaps. Let

$$\text{SURF}_Q = 2 \sum_{\substack{D \subset \{1, \dots, d\} \\ |D|=d-1}} \prod_{i \in D} q_i$$

be the volume of the surface of the cuboid  $Q = (q_1, \dots, q_d)$ . Note that for

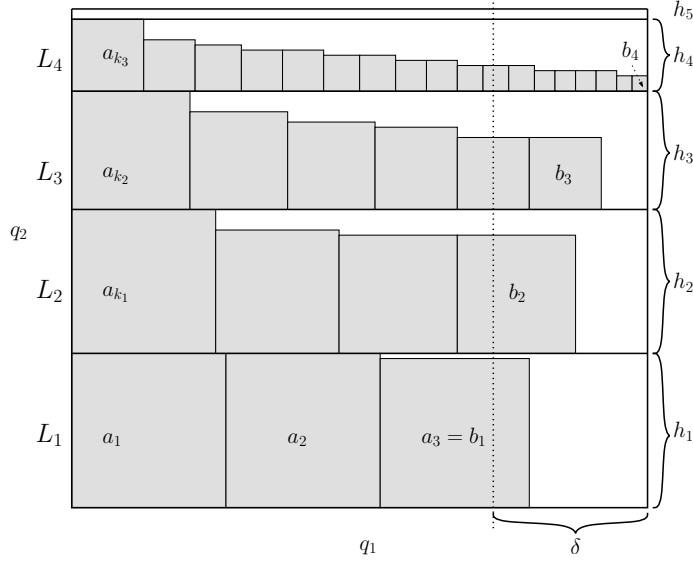


Fig. 2. A packing derived by NFDH.

$d = 3$ ,  $\text{SURF}_Q$  is actually the *area* of the surface of the cuboid  $Q$ . However, in higher dimensions *volume* is the appropriate expression. The following lemma is a generalization of Lemma 3.3 in [2] by Bansal et al.

**Lemma 4. NFDH**

Given a set  $S$  of small items  $a_i \leq \delta$  and a cuboid  $Q = (q_1, \dots, q_d)$ , then NFDH either packs all items from  $S$  into  $Q$  or the total free volume inside  $Q$  is at most  $\delta \text{SURF}_Q/2$ .

If the cuboid  $Q$  is a subset of the unit bin, i.e.,  $q_i \leq 1$ , we get  $\delta \text{SURF}_Q/2 \leq \delta d$ .

The main difference to the lemma by Bansal et al. is that we allow the bin to have arbitrary side lengths. Since  $\delta \text{SURF}_Q/2 \leq \delta \sum_{i=1}^d q_i$  for  $q_i \leq 1$  our lemma yields the lower bound of  $\delta \sum_{i=1}^d q_i$  by Bansal et. al [2] as a corollary.

*Proof.* The proof is by induction on the dimension  $d$ . Therefore we consider the  $i$ -dimensional restriction of  $Q$ , namely  $Q_i = (q_1, \dots, q_i)$  with volume  $\text{VOL}_{Q_i} = \prod_{j=1}^i q_j$  and volume of the surface

$$\text{SURF}_{Q_i} = 2 \sum_{\substack{D \subset \{1, \dots, i\} \\ |D|=i-1}} \prod_{j \in D} q_j.$$

The volume of the surface of  $Q_i$  is related to the volume of  $Q_{i-1}$  and volume

of the surface of  $Q_{i-1}$  as follows.

$$\begin{aligned}
\frac{\text{SURF}_{Q_i}}{2} &= \sum_{\substack{D \subset \{1, \dots, i\} \\ |D|=i-1}} \prod_{j \in D} q_j \\
&= \sum_{\substack{D \subset \{1, \dots, i-1\} \\ |D|=i-2}} q_i \prod_{j \in D} q_j + \prod_{j \in D = \{1, \dots, i-1\}} q_j \\
&= \frac{\text{SURF}_{Q_{i-1}}}{2} q_i + \text{VOL}_{Q_{i-1}}
\end{aligned} \tag{1}$$

Assume that NFDH for two dimensions does not pack all items and the packing consists of  $k$  layers. Let  $h_1, \dots, h_k$  be the heights of the layers  $L_1, \dots, L_k$  and  $h_{k+1}$  be the height of the unused layer above the packing—see Figure 2. Obviously, we have  $h_1 \leq \delta$ . Let  $b_i$  be the smallest item in layer  $L_i$  and let  $a_{k_i}$  be the item that did not fit into layer  $L_i$ , i.e., the first (largest) item in layer  $L_{i+1}$ . Then  $b_i \geq a_{k_i} = h_{i+1}$ , since the items are sorted and the height of each layer is defined by the largest item. Furthermore, each layer  $L_i$  is filled up to a width of at least  $q_1 - \delta$ , since otherwise another item would have been packed. For the filled volume  $V$  we thus get

$$\begin{aligned}
V &\geq \sum_{i=1}^k b_i \cdot (q_1 - \delta) \geq (q_1 - \delta) \sum_{i=2}^{k+1} h_i \geq (q_1 - \delta)(q_2 - \delta) \\
&\geq q_1 q_2 - \delta(q_1 + q_2) = \text{VOL}_{Q_2} - \delta \frac{\text{SURF}_{Q_2}}{2}.
\end{aligned}$$

We used  $q_2 = h_1 + \sum_{i=2}^{k+1} h_i$  and thus  $\sum_{i=2}^{k+1} h_i \geq q_2 - \delta$ .

Assume the claim holds for  $d - 1$  dimensions. Let again  $h_1, \dots, h_k$  be the heights of the layers  $L_1, \dots, L_k$ ,  $h_{k+1}$  be the height of the unused layer above the packing,  $b_i$  be the smallest item in each layer  $L_i$  and  $a_{k_i}$  be the item that did not fit into layer  $L_i$ . Again we have  $b_i \geq a_{k_i} = h_{i+1}$ . The main difference in the calculation is that the induction hypotheses gives a filled volume of at least  $\text{VOL}_{Q_{d-1}} - \delta \text{SURF}_{Q_{d-1}}/2$ . Since  $\sum_{i=2}^{k+1} h_i \geq q_d - \delta$  we get

$$\begin{aligned}
V &\geq \sum_{i=1}^k b_i \cdot \left( \text{VOL}_{Q_{d-1}} - \delta \frac{\text{SURF}_{Q_{d-1}}}{2} \right) \\
&\geq \left( \text{VOL}_{Q_{d-1}} - \delta \frac{\text{SURF}_{Q_{d-1}}}{2} \right) \sum_{i=2}^{k+1} h_i \\
&\geq \left( \text{VOL}_{Q_{d-1}} - \delta \frac{\text{SURF}_{Q_{d-1}}}{2} \right) (q_d - \delta) \\
&\geq \text{VOL}_{Q_d} - \delta \left( \frac{\text{SURF}_{Q_{d-1}}}{2} q_d + \text{VOL}_{Q_{d-1}} \right) = \text{VOL}_{Q_d} - \delta \frac{\text{SURF}_{Q_d}}{2} \quad \text{by (1)}.
\end{aligned}$$

If  $q_i \leq 1$  for  $i = 1, \dots, d$  then  $\text{SURF}_Q/2 \leq d$  as  $\text{SURF}_Q$  consists of  $d$  terms of

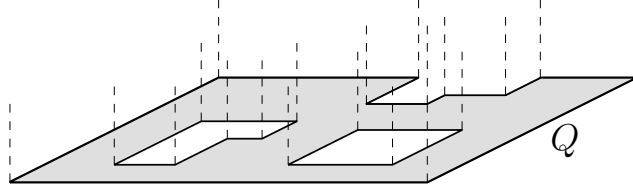


Fig. 3. A rectilinear basis  $Q$  with holes.

the form  $\prod_{j \in D} q_j \leq 1$ . Thus the lemma follows as the free volume inside  $Q$  is at most  $\delta \text{SURF}_Q / 2 \leq \delta d$  or all items are packed.  $\square$

Note that the previous lemma is also valid for very small cuboids  $Q$ . In particular, the lemma also holds if no item is packed, i.e.,  $q_i < a_1$  for any  $i \in \{1, \dots, d\}$ , since  $\text{VOL}_Q = \prod_{i=1}^d q_i \leq \delta \text{SURF}_Q / 2$  in this case.

Now we introduce a shifting technique that we use several times to free given hyperplanes  $H_i$  or a given space  $W$  inside a packing  $P$  without losing too much profit. Fishkin et al. used a similar technique in [11]. We give several different versions of the technique.

An *open strip packing* of a set  $I$  of items into a strip  $T = (Q, h)$  of basis  $Q$ , which is a rectilinear polygon, i.e., all its angles are 90 or 270 degrees, and height  $h$  is a relaxed packing where the items might intersect the top or bottom of  $T$ . We consider bases of these forms so that we can later pack small items into possibly very irregular gaps that are left in packings of big items. To do this we allow  $Q$  to have holes as well, which we will need in Section 6—see Figure 3 for a two-dimensional basis  $Q$  with holes.

**Lemma 5. Shifting technique**

Let  $\delta > 0$  and  $T = (Q, h)$  be a strip of basis  $Q$ , which is a rectilinear polygon, and height  $h \geq 2$ . Given an open strip packing  $P$  of a set  $I$  of hypercubes  $a_i \leq 1$  into  $T$  and  $t \leq \delta h$  hyperplanes  $H_1, \dots, H_t$  that intersect  $T$  orthogonally to the direction of height  $h$ . Then we can derive a packing  $P'$  of a selection  $I' \subseteq I$  into  $T$  with profit  $p(I') \geq (1 - 4\delta)p(I)$  in time  $\mathcal{O}(n \log n)$  such that no item intersects a hyperplane  $H_i$ .

*Proof.* Partition  $T$  into  $\ell = \lfloor h \rfloor \geq h - 1$  slices  $X_1, \dots, X_\ell$  of equal height  $h_{\text{slice}} \geq 1$  by drawing  $\lfloor h \rfloor$  equidistant hyperplanes orthogonally to the direction of height  $h$ —see Figure 4. Let the profit of a slice be the sum of profits of the items intersecting the slice. Find  $t$  slices  $X'_1, \dots, X'_t$  with lowest profit and remove all intersected items. If a plane  $H_i$  intersects with a slice  $X'_j$ , this plane is already free of items and we neglect  $H_i$  and  $X'_j$ . Now for each remaining plane  $H_i$ , move the intersected items left-aligned into a free slice  $X_j$ . Since there are at least as many remaining free slices as remaining planes, we can free all planes  $H_1, \dots, H_t$ . As for each item the intersection with at most two

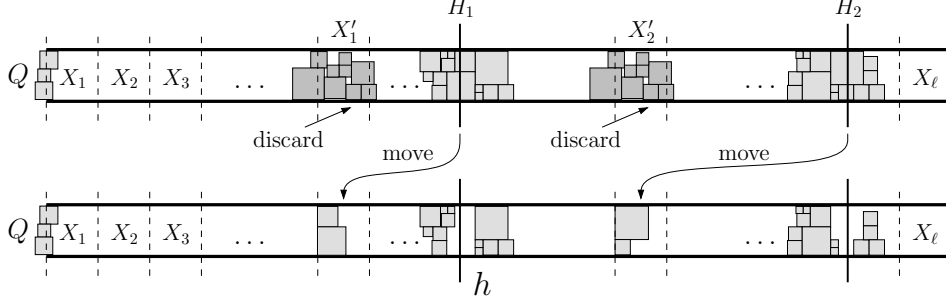


Fig. 4. Freeing planes  $H_1$  and  $H_2$  for  $d = 2$ . Presentation rotated by  $90^\circ$  for space reasons.

slices  $X_i, X_{i+1}$  is nonempty, we have  $p(X_1) + \dots + p(X_\ell) \leq 2p(I)$  and thus  $p(X'_1) + \dots + p(X'_t) \leq t \cdot 2p(I)/\ell$ . Thus the remaining profit is

$$p(I') \geq p(I) - t \frac{2p(I)}{\ell} \geq p(I) - \frac{2\delta h p(I)}{h-1} \geq (1-4\delta)p(I).$$

The last inequality follows by  $2\delta h/(h-1) \leq 4\delta h/h \Leftrightarrow 2 \leq h$  which was a precondition.

Finding the slices  $X'_1, \dots, X'_t$  with lowest profit can be done in time  $\mathcal{O}(n \log n)$  by sorting the items according to their coordinate in direction of  $h$  and sweeping through the items. Removing the items from these slices and moving the items intersecting  $H_1, \dots, H_t$  can be done in time  $\mathcal{O}(n)$ .  $\square$

In the previous lemma we assumed that  $h$  is large, which is the case for STRIP PACKING. In the KNAPSACK setting  $h$  cannot exceed 1 but we will have situations where the items are very small. This makes the ratio between the height of the items and their size sufficiently large. The following lemma explores this fact and allows us to apply the shifting technique on knapsack packings as well. In this setting we only need to free one hyperplane at a time.

**Lemma 6.** *Let  $1/2 \geq \delta > 0$  and  $T = (Q, h)$  be a strip of basis  $Q$ , which is a rectilinear polygon, and height  $h \geq 1/2 + \delta$ . Given an open strip packing  $P$  of a set  $S$  of small hypercubes  $a_i \leq \delta$  into  $T$  and a hyperplane  $H$  that intersects  $T$  orthogonally to the direction of height  $h$ . Then we can derive a packing  $P'$  of a selection  $S' \subseteq S$  into  $T$  with profit  $p(S') \geq (1-4\delta)p(S)$  in time  $\mathcal{O}(n \log n)$  such that no item intersects  $H$ .*

*Proof.* In this case a partition of  $T$  into  $\ell = \lfloor h/\delta \rfloor \geq h/\delta - 1$  slices is derived as a height of  $\delta$  is sufficient to move the items that intersect with  $H$  into a free slice. As  $2/(h/\delta - 1) \leq 4\delta$  for  $h \geq 1/2 + \delta$ , the lemma follows similar to the proof of Lemma 5 with  $p(I') \geq p(I) - 2p(I)/(h/\delta - 1) \geq (1-4\delta)p(I)$ .  $\square$

The following last variant of the shifting technique is needed to free a desig-

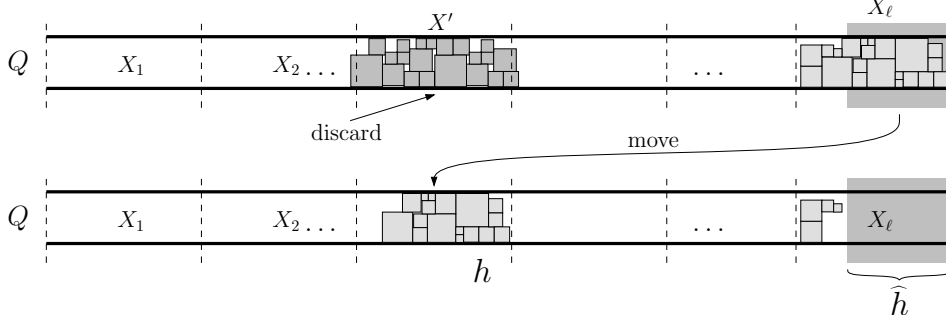


Fig. 5. Freeing the designated substrip  $\widehat{T}$  of height  $\widehat{h} = 2$  for  $d = 2$ . Presentation rotated by  $90^\circ$  for space reasons.

nated space at the top of  $T$  instead of some hyperplanes. We give this variant for the STRIP PACKING setting again, i.e., for items of size at most 1 and a large height  $h$ .

**Lemma 7.** *Let  $1/2 \geq \delta > 0$  and  $T = (Q, h)$  be a strip of basis  $Q$ , which is a rectilinear polygon, and height  $h > 1/\delta$ . Given a packing  $P$  of a set  $I$  of hypercubes  $a_i \leq 1$  into  $T$  and a designated substrip  $\widehat{T}$  of height  $\widehat{h}$  with  $0 < \widehat{h} \leq \delta h - 1$  at the top of  $T$ . Then we can derive a packing  $P'$  of a selection  $I' \subseteq I$  into  $T$  with profit  $p(I') \geq (1 - 4\delta)p(I)$  in time  $\mathcal{O}(n \log n)$  such that no item intersects  $\widehat{T}$ .*

*Proof.* See Figure 5 for an illustration of the lemma and the following proof. Note that  $\delta h > 1$  and thus the valid interval for  $\widehat{h}$  is non-empty. We partition  $T$  into  $\ell = \lfloor h/(\widehat{h} + 1) \rfloor \geq h/(\delta h) - 1 = 1/\delta - 1$  slices of equal height  $h_{\text{slice}} \geq \widehat{h} + 1$ . The extra height of 1 is needed since the items from the designated substrip  $\widehat{T}$  might reach back into the strip. By removing the items of the slice with lowest profit we get enough free space to move all items that intersect  $\widehat{T}$ . Similar to the calculations from Lemma 5 we get

$$p(I') \geq p(I) - \frac{2p(I)}{\ell} \geq p(I) - \frac{2p(I)}{1/\delta - 1} \geq (1 - 4\delta)p(I)$$

as  $2/(1/\delta - 1) \leq 4\delta$  for  $\delta \leq 1/2$ .  $\square$

Finally, we present the crucial result on RECTANGLE PACKING WITH LARGE RESOURCES by Fishkin, Gerber & Jansen [11] that we already mentioned in the introduction.

**Definition 8.** Given a set  $I = \{r_1, \dots, r_n\}$  of rectangles  $r_i = (w_i, h_i)$  with width  $w_i \leq 1$ , height  $h_i \leq 1$ , and associated profit  $p_i > 0$ . We denote the problem of finding a packing  $P$  of a selection  $I' \subseteq I$  into a rectangular bin  $B = (1, b)$  with  $b > 1$  such that the profit is maximized as RECTANGLE PACKING WITH LARGE RESOURCES.

**Theorem 9.** [11] *There is an algorithm for RECTANGLE PACKING WITH LARGE RESOURCES that computes a  $(1 - 72\epsilon)$ -optimal solution if  $b \geq 1/\epsilon^3$  and  $0 < \epsilon \leq 1/35$ . The running time of the algorithm is polynomial in  $n$  and  $1/\epsilon$ .*

We will give a  $d$ -dimensional generalization of this theorem restricted to hypercubes in Section 5.

Unfortunately, Fishkin et al. [11] do not give an explicit formula for the running time of their algorithm. Since we use it and its  $d$ -dimensional generalization as a subroutine in our algorithms, we also cannot bound the running times of our algorithms explicitly. We can only derive bounds for the number of times that we call this subroutine and we will show that this is bounded by a polynomial in  $n$ .

### 3 Square Packing

We now describe our main result for the two-dimensional case, a  $(5/4 + \epsilon)$ -approximation algorithm for SQUARE PACKING. Let us recall the definition first.

**Definition 10.** Given a set  $I = \{a_1, \dots, a_n\}$  of squares  $a_i \leq 1$  with associated profits  $p_i > 0$ . The SQUARE PACKING PROBLEM asks for a packing  $P$  of a selection  $I' \subseteq I$  into the unit square with maximum profit.

**Outline.** The first step of the algorithm is a separation of the items into sets of large, medium and small items. This yields a gap in size between large and small items and a profit of the medium items that is negligible. Since the number of large items in the bin is bounded by a constant, we can enumerate all possible selections and thus assume the knowledge of an optimal packing of large items. After that, we consider three different cases for packing: 1) the large items leave *enough remaining space* to pack the small items, 2) there are *several large items*, and 3) there is only *one very large item*.

We derive almost optimal solutions for the first and third case and an almost  $(k + 1)/k$ -optimal solution for the second case, where  $k$  is the number of large items. By showing that any packing with  $k < 4$  can be reduced to the first or the third case, we derive an overall approximation ratio of  $(5/4 + \epsilon)$ .

Let  $0 < \epsilon \leq 1/93$  and  $\epsilon' = \epsilon/4$ . The following technique divides an optimal solution  $I_{\text{opt}}$  into sets  $L_{\text{opt}}$  of large,  $M_{\text{opt}}$  of medium and  $S_{\text{opt}}$  of small items such that  $p(M_{\text{opt}}) \leq \epsilon' \text{OPT}(I)$  and thus we can neglect the medium items.

**Separation Technique.** Let  $r = \lceil 1/\epsilon' \rceil$ . Consider an optimal solution  $I_{\text{opt}}$  and the sequence  $\alpha_1 = (\epsilon')^3, \alpha_{i+1} = \alpha_i^4/6$  for  $i = 1, \dots, r$ . Let

$$M_i = \{s \in I_{\text{opt}} \mid s \in [\alpha_{i+1}, \alpha_i]\} \quad \text{for } 1 \leq i \leq r.$$

Observe that there is an index  $i^* \in \{1, \dots, r\}$  such that  $p(M_{i^*}) \leq \epsilon' p(I_{\text{opt}}) = \epsilon' \text{OPT}(I)$ . Let  $L_{\text{opt}} = \{s \in I_{\text{opt}} \mid s \geq \alpha_{i^*}\}$  be the set of large,  $M_{\text{opt}} = M_{i^*}$  the set of medium and  $S_{\text{opt}} = \{s \in I_{\text{opt}} \mid s < \alpha_{i^*+1}\}$  the set of small items. Thus  $p(L_{\text{opt}} \cup S_{\text{opt}}) \geq (1 - \epsilon') \text{OPT}(I)$  and it is sufficient to approximate this almost optimal solution. Let  $S = \{s \in I \mid s < \alpha_{i^*+1}\}$ , obviously  $S_{\text{opt}} \subseteq S$  and thus

$$\text{OPT}(L_{\text{opt}} \cup S) \geq (1 - \epsilon') \text{OPT}(I). \quad (2)$$

Since  $s \geq \alpha_{i^*}$  for  $s \in L_{\text{opt}}$ , there are at most  $1/\alpha_{i^*}^2$  items in  $L_{\text{opt}}$ . Let  $I_i = \{s \in I \mid s \geq \alpha_i\}$ . We enumerate over all  $i \in \{1, \dots, r\}$  and  $L \subseteq I_i$  with  $|L| \leq 1/\alpha_i^2$  and use the *constant packing* method from Lemma 1 to check the feasibility of  $L$ . By applying the remainder of the algorithm on every such  $i$  and feasible  $L$  we eventually consider the optimal values  $i^*$  and  $L_{\text{opt}}$ . We therefore assume that we are currently considering optimal values  $i^*$  and  $L_{\text{opt}}$ . Let  $P_{L_{\text{opt}}}$  be a packing of  $L_{\text{opt}}$  derived by the *constant packing* method. Note that the positions of the items in  $P_{L_{\text{opt}}}$  do not necessarily correspond to the positions in an optimal packing. In particular, the free space in  $P_{L_{\text{opt}}}$  might have a different structure than in any optimal packing. We will therefore use volume arguments to prove the desired approximation ratio.

In the following we add small items from a subset  $S' \subseteq S$  into the free space in  $P_{L_{\text{opt}}}$ . To do this we use Lemma 3 to derive a partition of the free space into gaps. To apply NFDH on several gaps we consider the gaps one after the other and for each gap we apply NFDH on the items that are not packed into one of the previous gaps. The following lemma bounds the free volume over all these gaps.

**Lemma 11.** *If NFDH does not pack all items from  $S'$  into the gaps, then the overall free volume  $V$  is at most  $\alpha_{i^*}^2$ .*

*Proof.* By Lemma 3, the number of gaps is bounded by  $3/\alpha_{i^*}^2$  since  $|L_{\text{opt}}| \leq 1/\alpha_{i^*}^2$ . Lemma 4 yields that packing small items from  $S'$  with NFDH into these gaps either packs all items or leaves a free volume of at most  $2\alpha_{i^*+1}$  in each gap. If items remain unpacked, the overall free volume  $V$  is bounded by

$$V \leq \frac{3}{\alpha_{i^*}^2} \cdot 2\alpha_{i^*+1} = \frac{3}{\alpha_{i^*}^2} \cdot \frac{2\alpha_{i^*}^4}{6} = \alpha_{i^*}^2.$$

Note that this is also a lower bound for the volume of an item in  $L_{\text{opt}}$ .  $\square$

As already mentioned we derive packings in three different cases: 1) *enough*

remaining space for the small items ( $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$ ), 2) several large items ( $|L_{\text{opt}}| = k \geq 4$ ), and 3) one very large item ( $a_{\text{max}} \geq 1 - (\epsilon')^3$ ), where  $a_{\text{max}}$  is the largest item in  $L_{\text{opt}}$ .

The following crucial lemma shows that these three cases cover all possibilities. It takes the full advantage of the square shapes of the items, namely that any packing with  $k < 4$  large items can be reduced to either the first or the third case. Our intuition is that it is impossible to fill a unit-size bin with either two or three equally big squares. This also turns out to be the reason for the improving approximation ratio with higher dimensions, e.g., either one very large or at least eight cubes are needed to fill a cube bin almost completely.

**Lemma 12.** *If  $|L_{\text{opt}}| < 4$ , then  $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$  or  $a_{\text{max}} \geq 1 - (\epsilon')^3$ .*

*Proof.* If  $L_{\text{opt}} = \emptyset$  then  $\text{Vol}(L_{\text{opt}}) = 0$ . Suppose that  $|L_{\text{opt}}| \in \{1, 2, 3\}$ . If  $a_{\text{max}} \leq 1/2$ , then  $\text{Vol}(L_{\text{opt}}) \leq 3/4 \leq 1 - \alpha_{i^*}$ . Otherwise  $a_{\text{max}} > 1/2$  and the smaller items have a maximal size of  $1 - a_{\text{max}}$  to be feasible with  $a_{\text{max}}$ . Thus a maximum volume is achieved for two additional items of size  $1 - a_{\text{max}}$ . Let  $x = a_{\text{max}}$  and consider the upper bound  $f(x) = x^2 + 2(1 - x)^2$  for  $\text{Vol}(L_{\text{opt}})$ . We show that  $f(x) \leq 1 - (\epsilon')^3$  for  $x \in [1/2, 1 - (\epsilon')^3]$ .

The upper bound  $f$  is convex, since  $f''(x) = 6$ . Therefore the maxima occur on the left and right sides of the interval. Namely,  $f(1/2) = 3/4$  and

$$\begin{aligned} f(1 - (\epsilon')^3) &= (1 - (\epsilon')^3)^2 + 2(1 - (1 - (\epsilon')^3))^2 \\ &= 1 - 2(\epsilon')^3 + 3(\epsilon')^6 \\ &\leq 1 - (\epsilon')^3 \end{aligned}$$

since  $(\epsilon')^3 \geq 3(\epsilon')^6$ . Thus  $a_{\text{max}} > 1 - (\epsilon')^3$  or  $\text{Vol}(L_{\text{opt}}) \leq f(x) \leq 1 - (\epsilon')^3 \leq 1 - \alpha_{i^*}$ .  $\square$

We now present the three methods for packing according to the different cases above.

**Lemma 13. *Enough Remaining Space***

*If  $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$ , we can find a selection  $S' \subseteq S$  of small items in time  $\mathcal{O}(n)$  such that  $L_{\text{opt}} \cup S'$  is feasible and  $p(L_{\text{opt}} \cup S') \geq (1 - 2\epsilon')\text{OPT}(I)$ .*

*Proof.* Let  $V := 1 - \text{Vol}(L_{\text{opt}}) \geq \alpha_{i^*}$  be the remaining volume in  $P_{L_{\text{opt}}}$ . Find a partition of the free space in  $P_{L_{\text{opt}}}$  into at most  $3/|L_{\text{opt}}|$  gaps in time  $\mathcal{O}(1)$ —see Lemma 3. If these gaps are filled with items from  $S' \subseteq S$  using NFDH, then by Lemma 11 either all items are packed or at most a total space of  $\alpha_{i^*}^2$  is left free.

To find a selection of small items that can be packed into the gaps, we use an instance of the *fractional* KNAPSACK PROBLEM, i.e., one-dimensional KNAPSACK PACKING where fractions of the items can be packed. Note that the well-known greedy algorithm finds an optimal solution with at most one fractional item and can be implemented to run in linear time using WEIGHTED MEDIAN SEARCH [26]. Consider the fractional knapsack instance with knapsack size  $V - \alpha_{i^*}^2 - \alpha_{i^*+1}^2$  and the size of the items in  $S$  be given by their volume. Let  $S'$  be an optimal solution of this instance derived by the greedy algorithm, *including* the possibly fractional item. Since the additional volume of the included fractional item is at most  $\alpha_{i^*+1}^2$ , we get  $\text{Vol}(S') \leq V - \alpha_{i^*}^2$ . Assume that  $S'$  cannot be packed completely into the gaps by NFDH. Then the packed volume is less than  $V - \alpha_{i^*}^2$ , which is a contradiction to a free space of at most  $\alpha_{i^*}^2$  from Lemma 11. Therefore  $S'$  can be packed completely by NFDH.

The non-fractional one-dimensional knapsack instance with items  $S$  given by their volume and knapsack size  $V$  is a relaxation of the packing of the small items. Let  $\text{OPT}_1(S)$  denote the optimum of this one-dimensional instance. Then

$$\text{OPT}_1(S) + p(L_{\text{opt}}) \geq \text{OPT}(L_{\text{opt}} \cup S). \quad (3)$$

Obviously,  $p(S') \geq \text{OPT}_{\text{frac}}(S)$ , where  $\text{OPT}_{\text{frac}}(S)$  denotes the optimum of the fractional knapsack instance that we consider. In a *fractional* knapsack problem, the average optimal profit per volume does not increase if we increase the size of the knapsack or if we restrict the problem to integer solutions. Thus

$$\frac{\text{OPT}_{\text{frac}}(S)}{V - \alpha_{i^*}^2 - \alpha_{i^*+1}^2} \geq \frac{\text{OPT}_1(S)}{V}. \quad (4)$$

Hence

$$\begin{aligned} p(S') &\geq \text{OPT}_{\text{frac}}(S) \\ &\geq \frac{V - \alpha_{i^*}^2 - \alpha_{i^*+1}^2}{V} \text{OPT}_1(S) && \text{by (4)} \\ &\geq \left(1 - \frac{\alpha_{i^*}^2 - \alpha_{i^*+1}^2}{\alpha_{i^*}^2}\right) \text{OPT}_1(S) && \text{since } V \geq \alpha_{i^*} \\ &\geq (1 - \epsilon') \text{OPT}_1(S) && \text{since } \alpha_{i^*} \leq (\epsilon')^3. \end{aligned}$$

The lemma follows with

$$\begin{aligned} p(L_{\text{opt}} \cup S') &= p(S') + p(L_{\text{opt}}) \\ &\geq (1 - \epsilon') \text{OPT}_1(S) + p(L_{\text{opt}}) \\ &\geq (1 - \epsilon') \text{OPT}(L_{\text{opt}} \cup S) && \text{by (3)} \\ &\geq (1 - 2\epsilon') \text{OPT}(I) && \text{by (2)}. \end{aligned}$$

An algorithmic description of the method that is applied in this case is given in Algorithm 1.  $\square$

---

**Algorithm 1** *Enough Remaining Space*

---

**Given:** Packing  $P_L$  of a set  $L$  of large items with  $\text{Vol}(L) \leq 1 - \alpha_{i^*}$

- 1: Derive a partition of the free space in  $P_L$  into gaps using Lemma 3
  - 2: find a selection  $S'$  as a greedy solution to the fractional knapsack instance with knapsack size  $V - \alpha_{i^*}^2 - \alpha_{i^*+1}^2$  and the size of the items in  $S$  be given by their volume including the possibly fractional item
  - 3: pack  $S'$  into the gaps using NFDH
- 

Note that the running time of  $\mathcal{O}(n)$  in the previous lemma only corresponds to *finding* a feasible subset  $S' \subseteq S$ . The corresponding packing can be found in time  $\mathcal{O}(n \log n)$  using NFDH. But this packing only needs to be computed for the final solution.

**Lemma 14. Several Large Items**

If  $|L_{\text{opt}}| = k$ , we can find a selection  $S' \subseteq S$  of small items in time  $\mathcal{O}(n \log n + T_{\text{KP}}(n, \epsilon))$ , where  $T_{\text{KP}}(n, \epsilon)$  is the running time of a *FPTAS* for one-dimensional knapsack packing, such that  $L_{\text{opt}} \cup S'$  is feasible and  $p(L_{\text{opt}} \cup S') \geq (k/(k+1) - 2\epsilon')\text{OPT}(I)$ .

Kellerer, Pferschy & Pisinger [24] mention a best-known running time of  $\mathcal{O}(n \min \{\log n, \log(1/\epsilon)\} + 1/\epsilon^2 \log(1/\epsilon) \cdot \min\{n, 1/\epsilon \log(1/\epsilon)\})$  for a *FPTAS* for one-dimensional knapsack packing.

*Proof.* Let  $\text{Knapsack}(S, K, \epsilon)$  be a solution with accuracy  $\epsilon$  for the one-dimensional knapsack instance with items  $S$  and knapsack size  $K$  found by a *FPTAS* from [24]. The items are given by their volume. Let  $S' = \text{Knapsack}(S, 1 - \text{Vol}(L_{\text{opt}}), \epsilon')$ . Note that  $p(L_{\text{opt}} \cup S') \geq (1 - \epsilon')\text{OPT}(L_{\text{opt}} \cup S) \geq (1 - 2\epsilon')\text{OPT}(I)$ , where the last step follows from Inequality (2). Consider a partition of the free space in  $P_{L_{\text{opt}}}$  into gaps by Lemma 3. Use NFDH to pack as many items as possible of  $S'$  into these gaps. Let the profit be  $P_1$ . If  $S'$  is completely packed,  $P_1 = p(L_{\text{opt}} \cup S') \geq (1 - 2\epsilon')\text{OPT}(I)$ .

Otherwise consider a second packing where we remove an item  $a^*$  with smallest profit from  $L_{\text{opt}}$ . Derive a new partition of the packing  $P_{L_{\text{opt}} \setminus \{a^*\}}$  into gaps and pack  $S'$  with NFDH into these gaps. By the considerations from the proof of Lemma 13,  $S'$  is completely packed (since the size of  $a^*$  is at least as large as the lower bound for the free space from Lemma 11). Let the profit of this second packing be  $P_2$ .

We show that  $\max\{P_1, P_2\} \geq k/(k+1) \cdot p(L_{\text{opt}} \cup S') \geq (k/(k+1) - 2\epsilon')\text{OPT}(I)$ .

Assume w.l.o.g. that  $L_{\text{opt}} = \{a_1, \dots, a_k\}$  and  $a^* = a_k$ . Then

$$P_1 \geq \sum_{i=1}^k p_i \geq k p_k \quad \text{and} \quad P_2 = p(L_{\text{opt}} \cup S') - p_k.$$

For  $p_k \leq p(L_{\text{opt}} \cup S')/(k+1)$  we get  $P_2 \geq p(L_{\text{opt}} \cup S') - p(L_{\text{opt}} \cup S')/(k+1) \geq k/(k+1) \cdot p(L_{\text{opt}} \cup S')$  and for  $p_k \geq p(L_{\text{opt}} \cup S')/(k+1)$  we get  $P_1 \geq k/(k+1) \cdot p(L_{\text{opt}} \cup S')$ .

The claim on the running time follows from the application of the *FPTAS* for one-dimensional knapsack packing, a running time of  $\mathcal{O}(1)$  for finding the gaps in the packings  $P_{L_{\text{opt}}}$  and  $P_{L_{\text{opt}} \setminus \{a^*\}}$ , and a running time of  $\mathcal{O}(n \log n)$  for applying NFDH. An algorithmic description of the method that is applied in this case is given in Algorithm 2.  $\square$

---

**Algorithm 2** *Several Large Items*

---

**Given:** Packing  $P_L$  of a set  $L$  of  $k$  large items

- 1: Let  $S' = \text{Knapsack}(S, 1 - \text{Vol}(L), \epsilon')$
  - 2: derive a partition of the free space in  $P_L$  into gaps using Lemma 3
  - 3: pack  $S'$  into these gaps using NFDH
  - 4: remove an item  $a^*$  with smallest profit from  $L$
  - 5: derive a partition of the free space in  $P_{L \setminus \{a_{\text{max}}\}}$  into gaps using Lemma 3
  - 6: pack  $S'$  into these gaps using NFDH
  - 7: return packing from Step 3 or 6 with highest profit
- 

**Lemma 15. One Very Large Item**

If  $a_{\text{max}} \geq 1 - (\epsilon')^3$ , we can find a selection  $R' \subseteq S \cup L_{\text{opt}} \setminus \{a_{\text{max}}\}$  of items in time  $\mathcal{O}(n \log n + T_{\text{LR}}(n, \epsilon))$ , where  $T_{\text{LR}}(n, \epsilon)$  is the running time of the algorithm from Theorem 9, such that  $\{a_{\text{max}}\} \cup R'$  is feasible and  $p(\{a_{\text{max}}\} \cup R') \geq (1 - 76\epsilon') \text{OPT}(I)$ .

Note that the running time  $T_{\text{LR}}(n, \epsilon)$  is not explicitly given in [11]. Instead the authors claim a running time polynomial in  $n$  and  $1/\epsilon$ .

*Proof.* The proof consists of two parts. First, we show that packing the big item  $a_{\text{max}}$  into the lower left corner of the bin does not change the optimal value too much. Second, we use Theorem 9 for RECTANGLE PACKING WITH LARGE RESOURCES to find an almost optimal packing for the remaining items.

See Figure 6 for the following construction. Consider an optimal packing of  $I_{\text{opt}}$  where  $a_{\text{max}}$  is placed at  $(x, y) \neq (0, 0)$ , i.e.,  $a_{\text{max}}$  is not placed in the lower left corner. Note that the free space to all sides has width (resp. height) at most  $1 - a_{\text{max}} \leq (\epsilon')^3$ . Our goal is to rotate the entire rectangle  $R_{\text{max}} = [0, x + a_{\text{max}}] \times [0, y + a_{\text{max}}]$  by  $180^\circ$  as illustrated in Figure 6(a). Hereby we move  $a_{\text{max}}$  into the lower left corner of the bin. Rotations of packings are

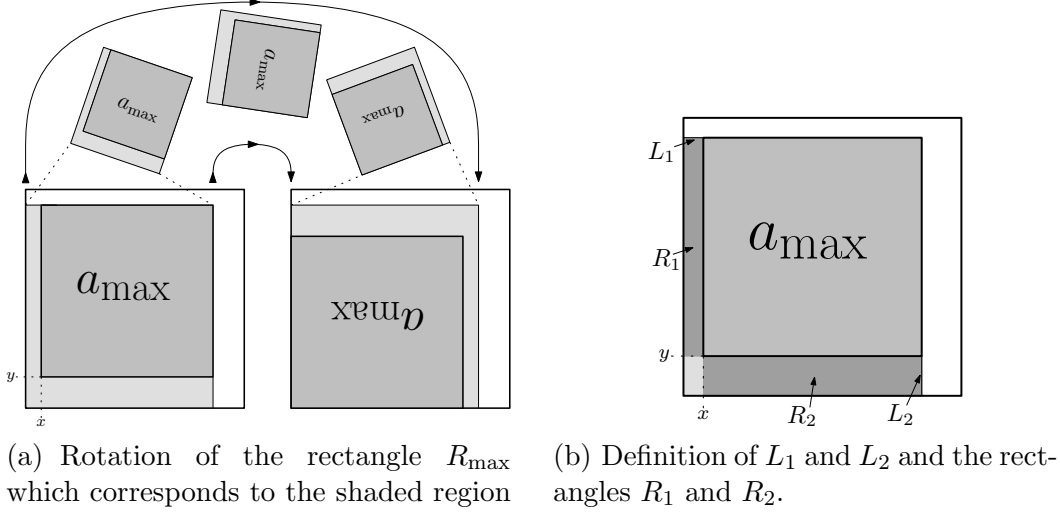


Fig. 6. Optimal solution with  $a_{\max}$  at position  $(x, y) \neq (0, 0)$

possible since all items are squares. To accomplish the rotation we have to ensure that no item in the packing intersects with line  $L_1$  from  $(0, y + a_{\max})$  to  $(x, y + a_{\max})$  or with line  $L_2$  from  $(x + a_{\max}, 0)$  to  $(x + a_{\max}, y)$ —see Figure 6(b). As some of the intersecting items might have high profit we cannot remove them directly, but need to apply the shifting technique of Lemma 6. Consider the *open strip packings* inside the rectangles  $R_1 = [0, x] \times [y, y + a_{\max}]$  to the left of  $a_{\max}$  and  $R_2 = [x, x + a_{\max}] \times [0, y]$  below  $a_{\max}$ —see Figure 6(b). For the application of the shifting technique on  $R_1$  we use  $\delta_1 := x \leq (\epsilon')^3$ . All conditions of Lemma 6 are satisfied since all items that intersect with  $R_1$  have size at most  $x$  and the height of  $R_1$  is  $h_1 = a_{\max} \geq 1/2 + (\epsilon')^3$ . Similarly, the shifting technique can be applied on  $R_2$  with  $\delta_2 := y \leq (\epsilon')^3$ . We obtain a packing without any item intersecting lines  $L_1$  and  $L_2$ . Lemma 6 allows us to bound the total loss by  $8(\epsilon')^3 \text{OPT}(L_{\text{opt}} \cup S) \leq \epsilon' \text{OPT}(L_{\text{opt}} \cup S)$ . The described rotation by  $180^\circ$  is possible without further loss of profit. We denote the packing into the remaining space around  $a_{\max}$  as *L-shaped* packing. Let  $R := S \cup L_{\text{opt}} \setminus \{a_{\max}\}$  be the set of remaining items that have to be packed into this space. Obviously,

$$\text{OPT}_{\text{L-shape}}(R) + p(a_{\max}) \geq (1 - \epsilon') \text{OPT}(L_{\text{opt}} \cup S), \quad (5)$$

where  $\text{OPT}_{\text{L-shape}}(R)$  denotes the optimal profit for L-shaped packings of the remaining items. We use  $p(a_{\max})$  to denote the profit of  $a_{\max}$  as the notion of  $p_{\max}$  could be misinterpreted.

Now assume  $a_{\max}$  is placed in the lower left corner of the bin. We approximate an optimal L-shaped packing by reassembling the L-shaped space around  $a_{\max}$  into a strip shaped bin to be able to apply Theorem 9. To do this we consider an optimal L-shaped packing of profit  $\text{OPT}_{\text{L-shape}}(R)$  and show that a packing with almost the same profit into the reassembled space exists. Draw a line  $L_3$

from  $(a_{\max}, a_{\max})$  to  $(a_{\max}, 1)$  and use the rectangle  $R_3 = [0, a_{\max}] \times [a_{\max}, 1]$  to apply the shifting technique with  $\delta_3 := 1 - a_{\max} \leq (\epsilon')^3$  to free this line. Again all items have size at most  $\delta_3$  and  $h_3 = a_{\max} \geq 1/2 + (\epsilon')^3$ . Rotate the right part of the remaining space by  $90^\circ$  and match it to the upper part—see Figure 7. This results in a strip-like shape of size  $(1 + a_{\max}, 1 - a_{\max})$ . Let  $\text{OPT}_{\text{strip}}(R)$  denote the optimal value for packing items from  $R$  into this strip-like bin. With Lemma 6 we get

$$\text{OPT}_{\text{strip}}(R) \geq (1 - \epsilon')\text{OPT}_{L\text{-shape}}(R). \quad (6)$$

It remains to approximate  $\text{OPT}_{\text{strip}}(R)$ . Scaling the strip and the remaining items in  $R$  by  $1/(1 - a_{\max})$  gives an instance of **RECTANGLE PACKING WITH LARGE RESOURCES** with a strip of size  $(b, 1)$  where  $b = (1 + a_{\max})/(1 - a_{\max}) \geq 1/(\epsilon')^3$  (as  $a_{\max} \geq 1 - (\epsilon')^3$ ). Thus by Theorem 9 we can find a packing of a subset  $R' \subseteq R$  with profit  $p(R') \geq (1 - 72\epsilon')\text{OPT}_{\text{strip}}(R)$  into the strip. Rescaling the packing gives a packing of  $R'$  into  $(1 + a_{\max}, 1 - a_{\max})$  without losing further profit.

By applying the shifting technique again at  $L_4$ , i.e., the position of  $L_3$  in the reassembled space, the solution can be adopted to the original shape. We get a final profit of

$$\begin{aligned} p &\geq (1 - \epsilon')p(R') + p(a_{\max}) && \text{freeing } L_4 \\ &\geq (1 - \epsilon')(1 - 72\epsilon')\text{OPT}_{\text{strip}}(R) + p(a_{\max}) && \text{Theorem 9} \\ &\geq (1 - \epsilon')(1 - 72\epsilon')(1 - \epsilon')\text{OPT}_{L\text{-shape}}(R) + p(a_{\max}) && \text{by (6)} \\ &\geq (1 - \epsilon')(1 - 72\epsilon')(1 - \epsilon')(1 - \epsilon')\text{OPT}(L_{\text{opt}} \cup S) && \text{by (5)} \\ &\geq (1 - \epsilon')(1 - 72\epsilon')(1 - \epsilon')(1 - \epsilon')(1 - \epsilon')\text{OPT}(I) && \text{by (2)} \\ &\geq (1 - 76\epsilon')\text{OPT}(I). \end{aligned}$$

The claim on the running time follows from the application of the algorithm for **RECTANGLE PACKING WITH LARGE RESOURCES** and a running time of  $\mathcal{O}(n \log n)$  for the shifting technique—see Lemma 6. An algorithmic description of the method that is applied in this case is given in Algorithm 3.  $\square$

---

**Algorithm 3** *One Very Large Item*

---

**Given:** Sets  $L$  and  $S$  where  $L$  is feasible and contains  $a_{\max} \geq 1 - (\epsilon')^3$

- 1: Pack  $a_{\max}$  into the lower left corner of the bin
  - 2: let  $R := S \cup L \setminus \{a_{\max}\}$
  - 3: find an almost optimal packing of a selection  $R' \subseteq R$  into the rectangle  $(1 + a_{\max}, 1 - a_{\max})$  using Theorem 9
  - 4: use the shifting technique to free  $L_4$
  - 5: insert the two parts of the packing into the free space around  $a_{\max}$
- 

The complete algorithm  $A_{\text{KP}}$  is summed up below.

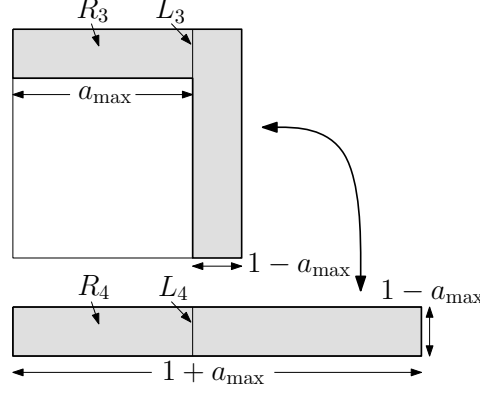


Fig. 7. Shifting the remaining space

---

**Algorithm 4**  $(5/4 + \epsilon)$ -algorithm  $A_{\text{KP}}$  for SQUARE PACKING

---

- 1: **for all**  $i \in \{1, \dots, r\}$  and  $L \subseteq \{s \in I \mid s \geq \alpha_i\}$  with  $|L| \leq 1/\alpha_i^2$  **do**
  - 2:     check feasibility of  $L$  with *constant packing method*
  - 3:     **if**  $L$  is feasible **then**
  - 4:         let  $P_L$  be the packing of  $L$  from Step 2
  - 5:         **case**  $\text{Vol}(L) \leq 1 - \alpha_i$ : pack almost optimally with Algorithm 1
  - 6:         **case**  $a_{\max} \geq 1 - (\epsilon')^3$ : pack almost optimally with Algorithm 3
  - 7:         **case**  $|L| \geq 4$ : pack with Algorithm 2
  - 8:     return the packing with the highest profit that was encountered
- 

**Theorem 16.** *Algorithm  $A_{\text{KP}}$  is a polynomial time algorithm for SQUARE PACKING with performance ratio  $(5/4 + \epsilon)$ .*

*Proof.* Algorithm  $A_{\text{KP}}$  iterates over all  $i \in \{1, \dots, r\}$  and  $L \subseteq \{s \in I \mid s \geq \alpha_i\}$  with  $|L| \leq 1/\alpha_i^2$ . Eventually the algorithm considers the iteration where  $i = i^*$  and  $L = L_{\text{opt}}$ . Lemma 12 shows that  $|L| \geq 4$  or  $\text{Vol}(L) \leq 1 - \alpha_i$  or  $a_{\max} \geq 1 - (\epsilon')^3$ . Therefore one of the Algorithms 1, 2 or 3 from Lemmas 13, 14 or 15 is applied on  $L$ . Since  $1 - 2\epsilon' \geq 1 - 76\epsilon' \geq 4/5 - 2\epsilon'$  for  $\epsilon' \leq 1/370$ , the profit is at least  $(4/5 - 2\epsilon')\text{OPT}(I)$ —see Lemma 14. The theorem follows as  $\frac{1}{4/5 - 2\epsilon'} \leq 5/4 + \epsilon$ .  $\square$

*Remark.* The running time of the algorithm  $A_{\text{KP}}$  is dominated by the enumeration on  $i$  and  $L$  and the unknown polynomial running time for RECTANGLE PACKING WITH LARGE RESOURCES from [11]. For each  $i$  a total of

$$\binom{n}{1/\alpha_i^2} = \mathcal{O}(n^{1/\alpha_i^2})$$

sets have to be checked for feasibility in time  $\mathcal{O}(2^{2/\alpha_i^4} \cdot 1/\alpha_i^4)$ —see Lemma 1. Afterwards one of the Algorithms 1, 2 or 3 from Lemmas 13, 14 or 15 is applied on the feasible sets. This requires time  $\mathcal{O}(n \log n + T_{\text{KP}}(n, \epsilon) + T_{\text{LR}}(n, \epsilon))$ , where  $T_{\text{KP}}(n, \epsilon)$  is the running time of a  $\mathcal{FPTAS}$  for one-dimensional knapsack packing

and  $T_{\text{LR}}(n, \epsilon)$  is the running time of the algorithm from Theorem 9. For any fixed  $\epsilon$  we iterate  $i$  over a constant number of values and  $\alpha_i$  is also constant. Thus we get a polynomial running time in  $n$  for any such fixed  $\epsilon$ .

Let us get some further insight into the magnitude of the running time. With  $\epsilon' = \epsilon/4$  we have

$$\alpha_i = \frac{(\epsilon')^{3 \cdot 4^{i-1}}}{6^{\sum_{j=0}^{i-1} 4^j}} = \frac{\epsilon^{3 \cdot 4^{i-1}}}{4^{3 \cdot 4^{i-1}} \cdot 6^{\sum_{j=0}^{i-1} 4^j}}$$

for  $i \geq 1$  which can easily be seen by induction since  $\alpha_1 = (\epsilon')^{3 \cdot 4^0} / 6^{\sum_{j=0}^0 4^j}$  and

$$\alpha_{i+1} = \frac{\alpha_i^4}{6} = \left( \frac{(\epsilon')^{3 \cdot 4^{i-1}}}{6^{\sum_{j=0}^{i-1} 4^j}} \right)^4 \cdot \frac{1}{6} = \frac{(\epsilon')^{3 \cdot 4^i}}{6^{\sum_{j=1}^i 4^j}} \cdot \frac{1}{6} = \frac{(\epsilon')^{3 \cdot 4^i}}{6^{\sum_{j=0}^i 4^j}}.$$

We iterate  $i$  from 1 to  $\lceil 1/\epsilon' \rceil = \lceil 4/\epsilon \rceil$ . Let  $t = \lceil 4/\epsilon \rceil$  be the number of iteration steps. Then we have a bound of  $t \cdot \mathcal{O}(n^{1/\alpha_t^2})$  on the total number of sets to be enumerated.

We stop our investigation at this point as we are not able to give a closed formula for the running time. As enumerations and separation parameters get even more involved for higher dimension, we will not comment in detail on the running times in the following sections but rather state that it is polynomial in  $n$  for any fixed  $\epsilon$ .

## 4 Hypercube Strip Packing

The algorithm for HYPERCUBE STRIP PACKING that we describe in this section is primarily based on methods from Kenyon & Rémila [25] for STRIP PACKING and from Bansal et al. [2] for HYPERCUBE BIN PACKING. As already mentioned in the introduction, Bansal et al. [3] describe a similar solution for three-dimensional STRIP PACKING where the basis of all items are squares. It is easy to see that our algorithm can be adapted such that only the basis of the items have to be hypercubes.

In general strip packing problems it is possible to scale the strip and the items in each dimension differently such that the basis of the strip is of unit size. This is obviously not possible when packing hypercubes. In order to be able to use our algorithm for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES we consider a different approach where the basis  $Q = (q_1, \dots, q_{d-1})$  has side lengths  $1 \leq q_i \leq C$  for some constant  $C$ . Thus we allow an arbitrary basis where the ratio between the shortest and longest side is bounded by

$C$ . Considering only strips with unit basis would simplify the description and analysis of our algorithm. But we would not be able to develop an appropriate algorithm for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES. For a fixed dimension  $d \geq 2$  and a fixed constant  $C \geq 1$  the problem is stated as follows:

**Definition 17.** Given a set  $I = \{a_1, \dots, a_n\}$  of  $d$ -dimensional cubes  $a_i \leq 1$  and a  $(d - 1)$ -dimensional basis  $Q = (q_1, \dots, q_{d-1})$  with  $1 \leq q_i \leq C$ . The HYPERCUBE STRIP PACKING PROBLEM *in the case of a maximal ratio between the shortest and longest side length of the basis of  $C$*  asks for a packing  $P$  of  $I$  into a strip of basis  $Q$  and unlimited height such that the overall height of the packing is minimized.

**Outline.** The algorithm can essentially be divided into the three steps of *separation, packing the large items* and *adding the small items*. Separating the items into sets of large, medium and small items yields a gap in size between large and small items that is crucial for adding the small items using NFDH into the gaps of a packing of the large items. The total volume of the medium items will be negligible such that they can be packed on top of the packing of the large and small items using NFDH again. Before packing the large items, a rounding step is applied that reduces the number of different sizes to a constant. This allows us to solve the *fractional* STRIP PACKING optimally. With a technique from [25] an integer solution is derived from this fractional solution. Finally, the small and medium items are added with NFDH.

We denote the volume of the basis  $Q = (q_1, \dots, q_{d-1})$  by  $\text{VOL}_Q = \prod_{i=1}^{d-1} q_i$  and recall the definition for the volume of the surface of the cuboid  $Q$  from Section 2 as

$$\text{SURF}_Q = 2 \sum_{\substack{D \subset \{1, \dots, d-1\} \\ |D|=d-2}} \prod_{i \in D} q_i.$$

Since the side lengths  $1 \leq q_i \leq C$  of the basis are bounded, we receive bounds for the volume  $1 \leq \text{VOL}_Q \leq C^{d-1}$  and the volume of the surface  $2(d - 1) \leq \text{SURF}_Q \leq 2(d - 1)C^{d-2}$  as well. Let  $\eta > 0$  be a fixed accuracy such that  $\eta \leq \min(1/(2C^{2d-2}), 1/3)$  and let  $\eta' = \eta/3$ .

**Separation.** Let  $r = \lceil 1/\eta' \rceil$  and consider the sequence

$$\alpha_1 = \min\left(\frac{2\eta'\text{VOL}_Q}{\text{SURF}_Q}, \eta'\right),$$

$$\alpha_{i+1} = \alpha_i^{(d-1)^2} \cdot \frac{\eta'}{2^{d-2}\text{VOL}_Q^{d-2}\text{SURF}_Q} \quad \text{for } 1 \leq i \leq r.$$

As  $\text{VOL}_Q/\text{SURF}_Q \geq 1/(2(d-1)C^{d-2})$  we have  $\alpha_1 \geq \eta'/((d-1)C^{d-2})$ . With the upper bounds for  $\text{VOL}_Q$  and  $\text{SURF}_Q$  we get

$$\alpha_{i+1} \geq \alpha_i^{(d-1)^2} \cdot \frac{\eta'}{2^{d-2} (C^{d-1})^{d-2} 2(d-1)C^{d-2}} \quad \text{for } 1 \leq i \leq r.$$

Thus for any fixed  $d$ ,  $C$  and  $\eta'$  and any  $i \in \{1, \dots, r\}$  there exists a constant lower bound  $\ell_i$  for  $\alpha_i$ . Obviously we have  $\alpha_i \geq \ell_r$  for  $i \in \{1, \dots, r\}$ .

Let

$$M_i = \{s \in I \mid s \in [\alpha_{i+1}, \alpha_i]\} \quad \text{for } 1 \leq i \leq r.$$

Since  $r = \lceil 1/\eta' \rceil$ , there is an index  $i^* \in \{1, \dots, r\}$  with  $\text{Vol}(M_{i^*}) \leq \eta' \text{Vol}(I)$ . Let  $L = \{s \in I \mid s \geq \alpha_{i^*}\}$  be the set of large,  $M = M_{i^*}$  the set of medium and  $S = \{s \in I \mid s < \alpha_{i^*+1}\}$  the set of small items.

**Rounding.** The rounding technique we use here was introduced by de la Vega & Lueker in [9] for one-dimensional BIN PACKING and Bansal et al. [2] applied this technique on HYPERCUBE BIN PACKING. The main idea is to derive a set  $L_{\text{sup}}$  such that for every item in  $L$  there is an item in  $L_{\text{sup}}$  that has at least the same size. In other words, if a packing for  $L_{\text{sup}}$  is known, a packing for  $L$  can be derived by substituting all corresponding items.

Let  $n_L$  be the number of items in  $L$ . Sort the items in  $L$  by non-increasing size and partition them into  $K = \lceil \text{VOL}_Q/(\eta' \alpha_{i^*}^d) \rceil$  groups  $G_1, \dots, G_{K-1}$  of  $m = \lceil n_L/K \rceil$  consecutive items and group  $G_K$  of the remaining  $m' \leq \lceil n_L/K \rceil$  items. Note that  $K \leq \lceil C^{d-1}/(\eta' \ell_r^d) \rceil$ . Denote the largest item in each group as the threshold item  $a_1^*, \dots, a_K^*$  and round all items in group  $G_i$  to the size of threshold item  $a_i^*$ . Denote the set of new items as  $L_{\text{sup}}$ . The different sizes that occur in  $L_{\text{sup}}$  are  $a_1^*, \dots, a_K^*$ , the sizes of the threshold items. The following lemma shows that this rounding does not affect the optimal solution too much.

**Lemma 18.** *The rounded instance  $L_{\text{sup}}$  satisfies*

$$\begin{aligned} \text{OPT}(L_{\text{sup}}) &\leq \text{OPT}(L) + \eta' \text{OPT}(L) + 1 && \text{and} \\ \text{Vol}(L_{\text{sup}}) &\leq \text{Vol}(L) + \eta' \text{Vol}(L) + 1. \end{aligned}$$

*Proof.* Construct another set of items  $L_{\text{inf}}$  by rounding the items of each group to the size of the threshold item of the *succeeding* group. The items of the last group  $G_K$  are disposed. Obviously  $\text{OPT}(L_{\text{inf}}) \leq \text{OPT}(L) \leq \text{OPT}(L_{\text{sup}})$ .  $L_{\text{sup}}$  and  $L_{\text{inf}}$  are similar except group  $G_1$ , which is empty in  $L_{\text{inf}}$ , and  $G_K$ , which might contain less items in  $L_{\text{sup}}$  than in  $L_{\text{inf}}$ . By ignoring the additional items of group  $G_K$  in  $L_{\text{inf}}$  and stacking up the  $m$  items of group  $G_1$  above the

remaining packing we get

$$\text{OPT}(L_{\text{sup}}) \leq \text{OPT}(L_{\text{inf}}) + m \leq \text{OPT}(L) + m \leq \text{OPT}(L) + \lceil \eta' \text{OPT}(L) \rceil.$$

Using the volume bound  $\text{OPT}(L) \geq \text{Vol}(L)/\text{VOL}_Q$ , the last part of the inequality follows with  $\text{Vol}(L) \geq n_L \alpha_{i^*}^d$  and  $m = \lceil n_L/K \rceil = \lceil n_L \eta' \alpha_{i^*}^d / \text{VOL}_Q \rceil \leq \lceil \eta' \text{OPT}(L) \rceil$ . The second inequality follows similarly with  $m = \lceil n_L/K \rceil = \lceil n_L \eta' \alpha_{i^*}^d / \text{VOL}_Q \rceil \leq \lceil n_L \eta' \alpha_{i^*}^d \rceil \leq \lceil \eta' \text{Vol}(L) \rceil$  since  $\text{VOL}_Q \geq 1$ .  $\square$

**Packing  $L_{\text{sup}}$ .** In order to pack the rounded set of large items  $L_{\text{sup}}$  efficiently, we consider the HYPERCUBE STRIP PACKING in the *fractional* variant. In this setting, the items can be cut horizontally, i.e., orthogonally to dimension  $d$ , and the pieces can be packed separately. Since this can only improve the packing,  $\text{OPT}_{\text{frac}}(L_{\text{sup}}) \leq \text{OPT}(L_{\text{sup}})$  holds. Now we show how to compute an optimal fractional solution with a linear program and how to convert this fractional solution into a non-fractional solution with almost the same height.

At first we introduce the central notion of a *packing pattern*. A packing pattern  $T_i = (T_{i1}, \dots, T_{iK})$  corresponds to a horizontal cut through a packing of  $L_{\text{sup}}$ .  $T_{ij}$  denotes the number of items of size  $a_j^*$  in this cut. The number of items in a cut is bounded by  $\text{VOL}_Q/\alpha_{i^*}^{d-1}$  and there are only  $K$  different item sizes. Thus  $K^{\text{VOL}_Q/\alpha_{i^*}^{d-1}}$  is a rough upper bound for the number of packing patterns. Therefore we can compute the set of all possible packing patterns  $\mathfrak{T} = \{T_1, \dots, T_v\}$  with  $v \leq K^{\text{VOL}_Q/\alpha_{i^*}^{d-1}}$  in constant time. Let  $h(G_j)$  denote the total height of items in group  $G_j$ . In the following linear program LP the variables  $x_i$  correspond to the cumulative height of packing pattern  $T_i$ .

$$\begin{aligned} \text{LP :} \quad & \min \sum_{i=1}^v x_i \\ & \sum_{i=1}^v T_{ij} x_i = h(G_j) && \text{for } 1 \leq j \leq K \\ & x_i \geq 0 && \text{for } 1 \leq i \leq v. \end{aligned}$$

**Claim 19.** LP is precisely fractional STRIP PACKING.

*Proof.* Consider a fractional packing  $P$  of  $L_{\text{sup}}$  with height  $h$ . Sweep  $P$  with an intersecting hyperplane and add the passed height to the corresponding packing pattern. Let  $x_i \geq 0$  denote the cumulative height for packing pattern  $T_i$  in  $P$ . We describe the packing  $P$  with the vector  $(x_1, \dots, x_v)$ . Obviously  $h = \sum_{i=1}^v x_i$  and  $\sum_{i=1}^v T_{ij} x_i = h(G_j)$  for  $1 \leq j \leq K$ .

On the other hand we can derive a fractional packing of height  $h = \sum_{i=1}^v x_i$  from a solution  $(x_1, \dots, x_v)$ . Let  $k \leq v$  denote the number of  $x_i > 0$  and w.l.o.g. let these be  $x_1, \dots, x_k$ . Construct a strip with  $k$  layers  $L_1, \dots, L_k$  of heights

$x_1, \dots, x_k$ . In each layer  $L_i$ , construct  $T_{ij}$  columns with a base size  $a_j^*$  according to packing pattern  $T_i$ —see Figure 8(a). Fill the items of group  $G_i$  greedily into the designated columns. If an item does not fit completely, cut it orthogonally to dimension  $d$  such that the current column is filled completely and proceed with the next column. All items can be packed since  $h(G_j) \leq \sum_{i=1}^v T_{ij}x_i$  for  $1 \leq j \leq K$ .  $\square$

**Running time and limitation of the number of layers.** As the linear program has constant size ( $v \leq K^{\text{VOL}_Q/\alpha_i^{d-1}}$  variables and  $K$  equations) a basic feasible solution can be found in constant time (see [4] for an introduction to linear programming). Such a basic feasible solution of a linear program with  $K$  equality constraints has at most  $K$  values  $x_i$  that are larger than 0. Therefore the number  $k$  of layers in the packing described above is bounded by  $K$ .

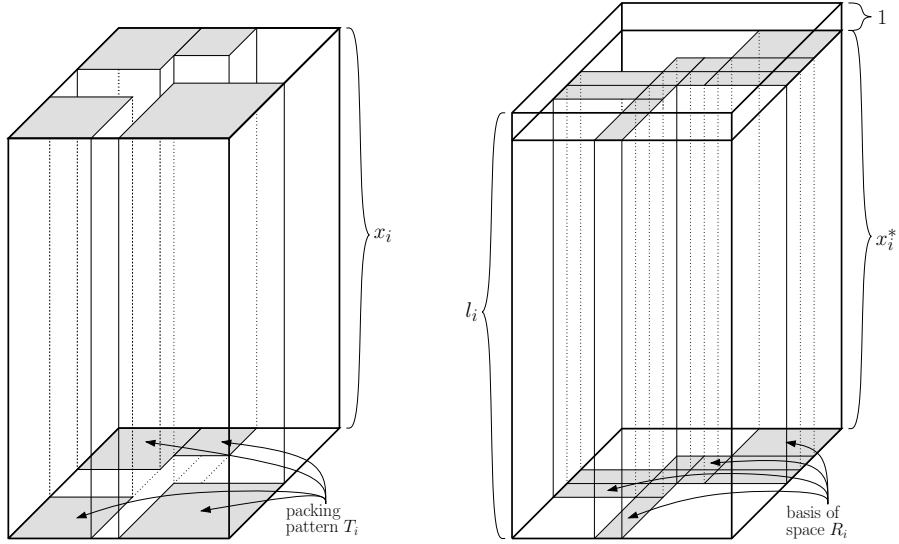
**Generation of a non-fractional packing.** Similar to the proof of the equivalence of the linear program and *fractional* STRIP PACKING we construct a non-fractional packing from a solution of the linear program. By increasing the heights of the layers we are able to pack the items without cutting them.

Let  $(x_1^*, \dots, x_q^*)$  be an optimal solution of the linear program with  $k \leq K$  variables  $x_i^* > 0$ . Let these be w.l.o.g.  $x_1^*, \dots, x_k^*$ . The height of the optimal fractional packing is  $\text{OPT}_{\text{frac}}(L_{\text{sup}}) = \sum_{i=1}^k x_i^*$ . We construct a non-fractional packing of height  $h_{\text{sup}} = \text{OPT}_{\text{frac}}(L_{\text{sup}}) + k$ .

Create a strip with  $k$  layers  $L_1, \dots, L_k$  of heights  $l_1 := x_1^* + 1, \dots, l_k := x_k^* + 1$ . In each layer  $L_i$ , construct  $T_{ij}$  columns with a base size  $a_j^*$  according to packing pattern  $T_i$ . Fill the items of group  $G_i$  greedily into the designated columns. If an item does not fit completely, close that column and proceed with the next column. Assume that an item  $a_i$  of group  $G_j$  cannot be packed. Since  $a_i \leq 1$ , each column for group  $G_j$  has to be filled up to the corresponding height  $x_j^*$ . The contradiction follows with  $\sum_{i=1}^v T_{ij}x_i^* \geq h(G_j)$ . Since  $k \leq K$  we derived a packing  $P_{L_{\text{sup}}}$  for  $L_{\text{sup}}$  of height

$$h_{\text{sup}} \leq \text{OPT}_{\text{frac}}(L_{\text{sup}}) + K. \quad (7)$$

**Adding the small items.** The small items are added with NFDH into certain gaps of the packing  $P_{L_{\text{sup}}}$ . To do this consider the layers  $L_1, \dots, L_k$  of the packing  $P_{L_{\text{sup}}}$ . The corresponding packing pattern  $T_i$  gives a partition of the free space of the basis of the layer into at most  $(2\text{VOL}_Q/\alpha_i^{d-1})^{d-1}$  cuboid gaps—see Lemma 2. Consider these gaps as the basis of a space  $R_i$  with height  $x_i^*$ —see Figure 8(b). Pack the small items with NFDH into these spaces  $R_i$  for  $i = 1, \dots, k$ . The following lemma bounds the free volume over all these spaces.



(a) Definition of the columns according to packing pattern  $T_i$  for the construction of a packing from a fractional solution  $(x_1, \dots, x_v)$

(b) The space  $R_i$  for packing the small items corresponds to the gaps in the packing pattern  $T_i$  with height  $x_i^*$ .

Fig. 8. Packing of a layer  $L_i$ . The vertical direction has been scaled down to illustrate that the increase of the height of each layer can be neglected asymptotically

**Lemma 20.** *If NFDH does not pack all items from  $S'$  into the spaces  $R_1, \dots, R_k$ , then for  $i = 1, \dots, k$  the free volume  $V_i$  in space  $R_i$  is bounded by*

$$V_i \leq \eta' \text{VOL}_Q x_i^* + 1.$$

*Proof.* We estimate the free volume in each space  $R_i$  for the case that not all items are packed by NFDH. The space  $R_i$  can be partitioned into at most  $(2\text{VOL}_Q/\alpha_i^{d-1})^{d-1}$  cuboid gaps by expanding the gaps in the basis to height  $x_i^*$ . Let  $\text{SURF}_{L_i}$  be the volume of the surface of layer  $L_i$ . Lemma 4 gives a bound of  $\alpha_i^{*+1}\text{SURF}_{L_i}/2$  for the free volume in each gap, since each gap is contained in  $L_i$  and thus  $\text{SURF}_{L_i}$  is an upper bound for the volume of the surface of the gap. As for  $L_i = (r_1, \dots, r_d) := (q_1, \dots, q_{d-1}, x_i^*)$  we have

$$\begin{aligned} \frac{\text{SURF}_{L_i}}{2} &= \sum_{\substack{D \subset \{1, \dots, d\} \\ |D|=d-1}} \prod_{j \in D} r_j \\ &= \left( \sum_{\substack{D \subset \{1, \dots, d-1\} \\ |D|=d-2}} \prod_{j \in D} r_j \right) \cdot r_d + \prod_{j=1, \dots, d-1} r_j \\ &= \frac{x_i^* \text{SURF}_Q}{2} + \text{VOL}_Q, \end{aligned} \tag{8}$$

we can estimate the free volume  $V_i$  after packing the small items with NFDH

into the space  $R_i$  by Lemma 2 and Lemma 4.

$$\begin{aligned}
V_i &\leq \left( \frac{2\text{VOL}_Q}{\alpha_{i^*}^{d-1}} \right)^{d-1} \cdot \alpha_{i^*+1} \left( \frac{x_i^* \text{SURF}_Q}{2} + \text{VOL}_Q \right) \\
&= \frac{2^{d-2} \text{VOL}_Q^{d-1} (x_i^* \text{SURF}_Q + 2\text{VOL}_Q)}{\alpha_{i^*}^{(d-1)^2}} \cdot \frac{\alpha_{i^*}^{(d-1)^2} \eta'}{2^{d-2} \text{VOL}_Q^{d-2} \text{SURF}_Q} \\
&= \eta' \text{VOL}_Q x_i^* + \frac{\eta' 2 \text{VOL}_Q^2}{\text{SURF}_Q} \\
&\leq \eta' \text{VOL}_Q x_i^* + 1. \tag{9}
\end{aligned}$$

The last inequality holds since  $\eta' < \eta \leq 1/(2C^{2d-2})$  and  $\text{SURF}_Q/(2\text{VOL}_Q^2) \geq 1/(2(C^{d-1})^2) = 1/(2C^{2d-2})$ .  $\square$

Open a new layer  $L_S$  above the current packing if the spaces  $R_i$  are not sufficient to pack all small items. Pack the remaining small items with NFDH into the new layer. We denote the height of this layer by  $h_S$  and derive the same bound on the free volume in this layer as for the spaces  $R_i$ .

**Lemma 21.** *The free volume  $V_S$  in layer  $L_S$  is bounded by*

$$V_S \leq \eta' \text{VOL}_Q h_S + 1.$$

*Proof.* In this case we can not apply Lemma 4 directly as all remaining small items are packed into  $L_S$ . Instead we draw a hyperplane  $H$  at distance  $\alpha_{i^*+1}$  from the top of  $L_S$ . Since NFDH did not pack all remaining small items below  $H$ , Lemma 4 and Inequality (8) give a bound of  $\alpha_{i^*+1}((h_S - \alpha_{i^*+1})\text{SURF}_Q/2 + \text{VOL}_Q)$  on the free space below  $H$ . The volume above  $H$  is  $\alpha_{i^*+1}\text{VOL}_Q$ . Since  $\text{VOL}_Q \geq 1$ ,  $\text{SURF}_Q \geq 1$  and  $\eta' < \text{SURF}_Q/(2\text{VOL}_Q^2)$  (see end of proof of Lemma 20), the free volume  $V_S$  is bounded by

$$\begin{aligned}
V_S &\leq \alpha_{i^*+1} \left( \frac{(h_S - \alpha_{i^*+1})\text{SURF}_Q}{2} + 2\text{VOL}_Q \right) \\
&\leq \frac{\alpha_{i^*}^{(d-1)^2} \eta'}{2^{d-2} \text{VOL}_Q^{d-2} \text{SURF}_Q} \frac{h_S \text{SURF}_Q}{2} + \frac{\alpha_{i^*}^{(d-1)^2} \eta'}{2^{d-2} \text{VOL}_Q^{d-2} \text{SURF}_Q} 2\text{VOL}_Q \\
&\leq \eta' h_S + 1 \\
&\leq \eta' \text{VOL}_Q h_S + 1.
\end{aligned}$$

$\square$

**Adding the medium items and substituting the large items.** Open another new layer  $L_M$  and pack the medium items  $M$  with NFDH into  $L_M$ . We denote the height of  $L_M$  by  $h_M$ . Finally, substitute the items of  $L_{\text{sup}}$  by the original items  $L$  and denote the final packing of  $I$  by  $P$  and the height

by  $h_{\text{total}}$ . Similar to Lemma 21 we get the following bound for the free volume in layer  $L_M$ .

**Lemma 22.** *The free volume  $V_M$  in layer  $L_M$  is bounded by*

$$V_M \leq \eta' \text{VOL}_Q h_M + 2.$$

*Proof.* The proof of Lemma 21 holds with  $\alpha_{i^*}$  instead of  $\alpha_{i^*+1}$  for  $i^* > 1$ . Otherwise  $i^* = 1$  and we have  $\alpha_{i^*} = \alpha_1 \leq 2\eta' \text{VOL}_Q / \text{SURF}_Q$ . Thus

$$\begin{aligned} V_S &\leq \alpha_{i^*} \left( \frac{(h_S - \alpha_{i^*+1}) \text{SURF}_Q}{2} + 2 \text{VOL}_Q \right) \\ &\leq \frac{2\eta' \text{VOL}_Q}{\text{SURF}_Q} \frac{h_S \text{SURF}_Q}{2} + \frac{2\eta' \text{VOL}_Q}{\text{SURF}_Q} 2 \text{VOL}_Q \\ &\leq \eta' \text{VOL}_Q h_S + 2. \end{aligned}$$

The last step follows since  $\eta' \leq \text{SURF}_Q / (2 \text{VOL}_Q^2)$  (see end of proof of Lemma 20).  $\square$

---

**Algorithm 5** *APTAS*  $A_{\text{Strip}}$  for HYPERCUBE STRIP PACKING

---

- 1: Compute  $i^* \in \{1, \dots, r\}$  such that  $\text{Vol}(M_{i^*}) \leq \eta' \text{Vol}(I)$ ,
  - 2: sort  $L$  by non-increasing height and create a rounded instance  $L_{\text{sup}}$ ,
  - 3: compute an optimal fractional strip packing for  $L_{\text{sup}}$ ,
  - 4: create a non-fractional strip packing with  $k \leq K$  layers for  $L_{\text{sup}}$  with height  $h_{\text{sup}}$ ,
  - 5: pack the small items  $S$  into the spaces  $R_1, \dots, R_k$  and possibly into an additional layer of height  $h_S$  above the packing,
  - 6: pack the medium items into an additional layer of height  $h_M$  above the packing,
  - 7: substitute all items in  $L_{\text{sup}}$  by the corresponding items in  $L$  to derive packing  $P$ ,
  - 8: denote the derived height by  $h_{\text{total}} = h_{\text{sup}} + h_S + h_M$ .
- 

**Analysis.** We distinguish two cases for the analysis, depending on whether an additional layer was opened for the small items.

**Lemma 23.** *If an additional layer was opened for the small items then*

$$h_{\text{total}} \leq (1 + 3\eta') \text{OPT}(I) + \frac{1}{1 - \eta'} (2K + 4).$$

*Proof.* The proof makes use of a volume argument based on Lemma 20 which is applicable in this case as NFDH did not pack all small items from  $S'$  into the spaces.

Let the volume of packing pattern  $T_i$  be  $c_i = \sum_{j=1}^K T_{ij}(a_j^*)^{d-1}$ . The following inequality holds for the volume of  $L$  (Lemma 18).

$$\text{Vol}(L) \geq \text{Vol}(L_{\text{sup}}) - \eta' \text{Vol}(L) - 1 = \sum_{i=1}^k c_i x_i^* - \eta' \text{Vol}(I) - 1.$$

Using Lemmas 20 and 21, we can estimate the packed volume of  $S$  by

$$\begin{aligned} \text{Vol}(S) &\geq \sum_{i=1}^k \left( \text{VOL}_{R_i} - (\eta' \text{VOL}_Q x_i^* + 1) \right) + \text{VOL}_{L_S} - (\eta' \text{VOL}_Q h_S + 1) \\ &= \sum_{i=1}^k \left( (\text{VOL}_Q - c_i) x_i^* - \eta' \text{VOL}_Q x_i^* - 1 \right) + \text{VOL}_Q h_S - \eta' \text{VOL}_Q h_S - 1. \end{aligned}$$

Here  $\text{VOL}_{R_i} = (\text{VOL}_Q - c_i) x_i^*$  denotes the volume of the space  $R_i$ . With Lemma 22 we get

$$\text{Vol}(M) \geq \text{VOL}_Q h_M - \eta' \text{VOL}_Q h_M - 2.$$

Summed up we get

$$\begin{aligned} \text{Vol}(I) &= \text{Vol}(L) + \text{Vol}(S) + \text{Vol}(M) \\ &\geq \sum_{i=1}^k c_i x_i^* - \eta' \text{Vol}(I) - 1 \\ &\quad + \sum_{i=1}^k \left( (\text{VOL}_Q - c_i) x_i^* - \eta' \text{VOL}_Q x_i^* - 1 \right) + \text{VOL}_Q h_S - \eta' \text{VOL}_Q h_S - 1 \\ &\quad + \text{VOL}_Q h_M - \eta' \text{VOL}_Q h_M - 2 \\ &= (\text{VOL}_Q - \eta' \text{VOL}_Q) \left( \sum_{i=1}^k x_i^* + h_S + h_M \right) - \sum_{i=1}^k 1 - \eta' \text{Vol}(I) - 4 \\ &= (\text{VOL}_Q - \eta' \text{VOL}_Q) \left( \sum_{i=1}^k (l_i - 1) + h_S + h_M \right) - k - \eta' \text{Vol}(I) - 4 \\ &= (\text{VOL}_Q - \eta' \text{VOL}_Q) h_{\text{total}} - (\text{VOL}_Q - \eta' \text{VOL}_Q + 1) k - \eta' \text{Vol}(I) - 4 \\ &\geq (\text{VOL}_Q - \eta' \text{VOL}_Q) h_{\text{total}} - 2 \text{VOL}_Q k - \eta' \text{Vol}(I) - 4 \\ &\geq (\text{VOL}_Q - \eta' \text{VOL}_Q) h_{\text{total}} - 2 \text{VOL}_Q K - \eta' \text{Vol}(I) - 4. \end{aligned}$$

The second to last step follows with  $\text{VOL}_Q \geq 1$  and thus  $\text{VOL}_Q - \eta' \text{VOL}_Q + 1 \leq 2 \text{VOL}_Q$ . Since  $\text{OPT}(I) \geq \text{Vol}(I) / \text{VOL}_Q$  and  $\text{VOL}_Q \geq 1$  we have

$$\begin{aligned} (1 + \eta') \text{Vol}(I) &\geq (\text{VOL}_Q - \eta' \text{VOL}_Q) h_{\text{total}} - 2 \text{VOL}_Q K - 4 \\ \Leftrightarrow h_{\text{total}} &\leq \frac{1 + \eta'}{\text{VOL}_Q - \eta' \text{VOL}_Q} \text{Vol}(I) + \frac{1}{\text{VOL}_Q - \eta' \text{VOL}_Q} (2 \text{VOL}_Q K + 4) \\ &\leq \frac{1 + \eta'}{1 - \eta'} \text{OPT}(I) + \frac{1}{1 - \eta'} (2K + 4). \end{aligned}$$

The claim follows with  $(1 + \eta') / (1 - \eta') \leq 1 + 3\eta'$  for  $\eta' \leq 1/3$ .  $\square$

**Lemma 24.** *If no additional layer was opened for the small items then*

$$h_{\text{total}} \leq (1 + 3\eta')\text{OPT}(I) + K + 4.$$

*Proof.* Obviously  $h_{\text{total}} = h_{\text{sup}} + h_M$ . As in the proof of the previous lemma we have

$$\text{Vol}(M) \geq \text{VOL}_Q h_M - \eta' \text{VOL}_Q h_M - 2.$$

To derive an upper bound for  $h_M$  we use  $\text{Vol}(M) \leq \eta' \text{Vol}(I)$ . We get

$$\begin{aligned} (\text{VOL}_Q - \eta' \text{VOL}_Q) h_M &\leq \text{Vol}(M) + 2 \leq \eta' \text{Vol}(I) + 2 \leq \eta' \text{VOL}_Q \text{OPT}(I) + 2 \\ \Rightarrow h_M &\leq \frac{\eta'}{1 - \eta'} \text{OPT}(I) + \frac{2}{\text{VOL}_Q - \eta' \text{VOL}_Q} \leq 2\eta' \text{OPT}(I) + 3, \end{aligned}$$

where we use  $\eta' \leq 1/3$  and  $1/(\text{VOL}_Q - \eta' \text{VOL}_Q) \leq 3/2$  since  $\text{VOL}_Q \geq 1$  and thus  $\text{VOL}_Q - \eta' \text{VOL}_Q \geq 2/3$ . It follows that

$$\begin{aligned} h_{\text{total}} &= h_{\text{sup}} + h_M \\ &\leq \text{OPT}_{\text{frak}}(L_{\text{sup}}) + K + 2\eta' \text{OPT}(I) + 3 && \text{by Inequality (7)} \\ &\leq \text{OPT}(L_{\text{sup}}) + K + 2\eta' \text{OPT}(I) + 3 \\ &\leq (1 + 3\eta') \text{OPT}(I) + K + 4 && \text{by Lemma 18.} \end{aligned}$$

□

Using  $\eta' = \eta/3$ , Lemma 23 and Lemma 24 yield

$$h_{\text{total}} \leq (1 + \eta) \text{OPT}(I) + Z \tag{10}$$

with  $Z = \frac{1}{1-\eta/3}(2K + 4)$  being an upper bound for the additive constant.

Before we state the final theorem we introduce upper bounds for  $K$  and  $Z$ , the number of groups in the rounding step and the additive constant, respectively, and for the number of layers that we used. These bounds are crucial for the algorithms in the following sections. All bounds only depend on  $d$ ,  $C$  and  $\eta' = \eta/3$  and are therefore constant for fixed  $d$ ,  $C$  and  $\eta$ . Recall that we denoted the lower bound for the constants  $\alpha_i$  by  $\ell_i$  and that  $\ell_i$  depends only on  $d$ ,  $C$  and  $\eta'$ . Let  $K_{\text{Strip}} := \lceil C^{d-1}/(\eta' \ell_r^d) \rceil$  with  $r = \lceil 1/\eta' \rceil$ . As  $\text{VOL}_Q \leq C^{d-1}$  we have  $K = \lceil \text{VOL}_Q/(\eta' \alpha_{i^*}^d) \rceil \leq K_{\text{Strip}}$ . Thus  $K_{\text{Strip}}$  is an upper bound for the number of groups in the rounding step. Then  $Z_{\text{Strip}} := \frac{1}{1-\eta'}(2K_{\text{Strip}} + 4) \geq Z$  is an upper bound for the additive constant and  $L_{\text{Strip}} := K_{\text{Strip}} + 2$  is an upper bound for the number of layers (including the additional layers for the small and medium items).

Using these constant bounds, Inequality (10) finally yields the following theorem for any fixed dimension  $d \geq 2$  and any maximal ratio between the shortest and longest side length of the basis  $C \geq 1$ .

**Theorem 25.** For any fixed accuracy  $0 < \eta \leq \min(1/(2C^{2d-2}), 1/3)$  and any given set  $I = \{a_1, \dots, a_n\}$  of  $d$ -dimensional cubes  $a_i \leq 1$  and  $(d-1)$ -dimensional basis  $Q = (q_1, \dots, q_{d-1})$  with  $1 \leq q_i \leq C$ , Algorithm  $A_{\text{Strip}}$  has running time polynomial in  $n$  and outputs a packing  $P$  of  $I$  into a strip with basis  $Q$  and height  $h \leq (1 + \eta)\text{OPT}(I) + Z_{\text{Strip}}$ .  $P$  consist of at most  $L_{\text{Strip}}$  layers, each corresponding to a packing pattern (including an empty one).

## 5 Hypercube Knapsack Packing with Large Resources

The algorithm presented in this section is a generalization of the algorithm from Fishkin et al. [11] on RECTANGLE PACKING WITH LARGE RESOURCES. In the original two-dimensional setting, the bin had unit size in one direction and a very large size in the other direction. In order to use our algorithm for the algorithm for HYPERCUBE KNAPSACK PACKING we have to relax this restriction. Instead we assume that the bin has a volume that is sufficiently large.

For any fixed dimension  $d \geq 2$  let  $\mu \in (0, 1/(4d + 33)]$  be a fixed accuracy and  $\mu' = \mu/(4d + 34)$ . Let  $C = 1/\mu'$  and consider the accuracy parameter  $\eta = 1/(2C^{2d-2}) = (\mu')^{2d-2}/2$  for Algorithm  $A_{\text{Strip}}$ . Furthermore, let  $Z_{\text{Strip}}$ ,  $L_{\text{Strip}}$  and  $K_{\text{Strip}}$  denote the bounds for the additive constant, the number of layers and the number of groups in the rounding step, respectively, as defined at the end of the previous section, using  $C = 1/\mu'$  and accuracy  $\eta$ . Thus with  $\ell_r$  being a lower bound for  $\alpha_r$  for  $r = \lceil 1/\eta' \rceil$  and  $\eta' = \eta/3$  we have

$$\begin{aligned} K_{\text{Strip}} &= \left\lceil \frac{C^{d-1}}{\eta' \ell_r^d} \right\rceil = \left\lceil \frac{1}{(\mu')^{d-1}} \cdot \frac{6}{(\mu')^{2d-2}} \cdot \frac{1}{\ell_r^d} \right\rceil = \left\lceil \frac{6}{(\mu')^{3d-3} \ell_r^d} \right\rceil \quad \text{and} \\ Z_{\text{Strip}} &= \frac{1}{1 - \eta'} (2K_{\text{Strip}} + 4) = \frac{1}{1 - (\mu')^{2d-2}/6} (2K_{\text{Strip}} + 4) \quad \text{and} \\ L_{\text{Strip}} &= K_{\text{Strip}} + 2. \end{aligned}$$

Note that all values above depend only on  $d$  and  $\mu$  as the lower bound  $\ell_r$  also depends only on  $d$ ,  $C$  and  $\eta$  (and thus on  $d$  and  $\mu$ ). Thus  $Z_{\text{Strip}}$ ,  $L_{\text{Strip}}$  and  $K_{\text{Strip}}$  are constant for any fixed values  $d$  and  $\mu$ .

*Remark.* Usually we would like to use the algorithm  $A_{\text{Strip}}$  in a black-box fashion, not caring that the internal accuracy needs to be adjusted. The need to define the internal accuracy parameter  $\eta$  for  $A_{\text{Strip}}$  arises because the following algorithm for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES heavily relies on the structure of the packings that  $A_{\text{Strip}}$  derives.

**Definition 26.** Given a set  $I = \{a_1, \dots, a_n\}$  of  $d$ -dimensional cubes  $a_i \leq 1$  with profits  $p_i > 0$  and a  $d$ -dimensional bin  $B = (b_1, \dots, b_d)$  with side lengths

$1 \leq b_1 \leq \dots \leq b_d$  and volume  $\text{VOL}_B = \prod_{i=1}^d b_i \geq Z_{\text{Strip}}/(\mu')^d$ . The HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES asks for a packing  $P$  of a selection  $I' \subseteq I$  into the bin  $B$  with maximal profit.

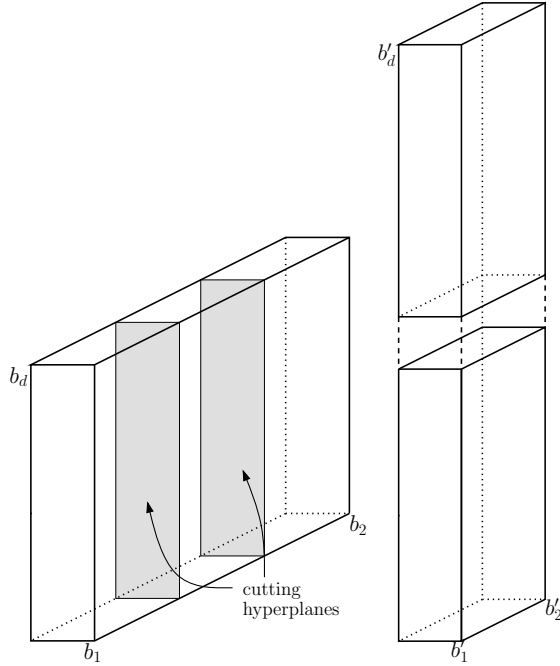
The assumption that  $\text{VOL}_B = \prod_{i=1}^d b_i \geq Z_{\text{Strip}}/(\mu')^d$  is similar to the assumption for RECTANGLE PACKING WITH LARGE RESOURCES by Fishkin, Gerber & Jansen [11]. As already mentioned in the introduction, the authors showed that a  $(1 - 72\epsilon)$ -optimal solution can be found if one side length of the bin is at least  $1/\epsilon^3$  times larger than the items for any  $0 < \epsilon < 1/72$ . Thus if we scale the bin such that the shorter length is 1 they assumed that  $\text{VOL}_B \geq 1/\epsilon^3$ .

**Outline.** As already mentioned, the main difference between the multi-dimensional setting and the two-dimensional setting is that we have a lower bound on the volume instead of a certain direction of the bin. By cutting the bin at certain hyperplanes and reassembling it to a strip-like bin we can even out this difference. Hence applying methods from Fishkin et al. [11] is easy on the newly assembled bin. First, we consider a packing of an optimal solution  $I_{\text{opt}}$  by Algorithm  $A_{\text{Strip}}$ . Second, we guess the structure of such a packing and derive upper bounds for the selection of items. With a number of one-dimensional knapsack instances we select the items for packing. Third, we pack the selected items similar to the guessed structure and the approach of  $A_{\text{Strip}}$ .

**Reassembling the bin.** We want to change the shape of the bin to a strip-like form without losing too much profit. As the definitions of the constants above indicate, we are interested in a bin where the side length of the basis are bounded by  $C = 1/\mu'$ . See Figure 9 for an illustration of the following reassembling.

Consider an optimal solution  $I_{\text{opt}}$  with a packing  $P_{\text{opt}}$  into the bin  $B$ . Let  $k$  denote the smallest index such that  $b_k > 1/\mu'$  and let  $t$  be the smallest integer with  $b_k \leq (t + 1)/\mu'$ . Cut  $B$  in direction  $k$  in  $t + 1$  equally sized parts. To do this  $t$  cutting hyperplanes are needed. Let  $b'_k = b_k/(t + 1)$  denote the length of the parts in direction  $k$ . Obviously  $1/(2\mu') < b'_k \leq 1/\mu'$ . Let  $\delta = \mu'$  for the application of the shifting technique from Lemma 5. The shifting technique is applicable as  $b_k > 1/\mu' > 2$ ,  $a_i \leq 1$  for all items  $a_i$  and  $t \leq \mu' b_k$  by definition of  $t$ . The loss in profit due to the cutting is at most  $4\mu' \text{OPT}(I)$ . Reassemble the parts in direction  $d$  and continue with the next direction, except direction  $d$  itself.

In the newly assembled bin, all side lengths but  $b'_d$  are bounded by  $1/\mu'$ . Since



(a) Cutting the bin at  $t$  (b) Reassembling the bin in direction  $d$

Fig. 9. Reassembling the bin (direction  $d$  is scaled down for space reasons)

the overall volume did not change we get

$$b'_d = \frac{\text{VOL}_B}{\prod_{i=1}^{d-1} b'_i} \geq \frac{\text{VOL}_B}{(1/\mu')^{d-1}} \geq \frac{Z_{\text{Strip}}}{\mu'}. \quad (11)$$

The overall loss in profit for reassembling the bin is at most  $4(d-1)\mu'\text{OPT}(I)$ . We denote the new bin by  $B' = (b'_1, \dots, b'_d)$  with optimal profit  $\text{OPT}'(I)$ . Let  $J'$  be a set of items that is feasible for  $B'$ . To derive a packing of  $J \subseteq J'$  into  $B$  from a packing of  $J'$  into  $B'$  we cut  $B'$  in direction  $d$  into pieces of size  $b_d$ . These pieces can easily be reassembled to form the bin  $B$ . As we have  $a_i \leq 1$  for all items  $a_i$  and  $b'_d > 2$ , the shifting technique of Lemma 5 can be applied with  $\delta = 1/b_d$  since  $t = b'_d/b_d - 1$  cuts are necessary. Thus the loss is bounded by  $4/b_d p(J') \leq 4\mu' p(J')$  (as  $b_d \geq b_{d-1} \geq \dots \geq b_1$  and  $\prod_{i=1}^d b_i \geq 1/(\mu')^d$ ). We are therefore interested in approximating an optimal packing for  $B'$ . By the considerations above the following inequalities hold:

$$\text{OPT}'(I) \geq (1 - 4(d-1)\mu')\text{OPT}(I) \quad (12)$$

$$p(J) \geq (1 - 4\mu')p(J'). \quad (13)$$

The following steps are generalizations of the two-dimensional algorithm in [11]. In order to apply Algorithm  $A_{\text{Strip}}$  we consider the bin as a strip. Therefore we denote the basis of the bin by  $Q' = (b'_1, \dots, b'_{d-1})$  and the volume and the volume of the surface of  $Q'$  by  $\text{VOL}_{Q'}$  and  $\text{SURF}_{Q'}$ , respectively. As mentioned

above we use the accuracy parameter  $\eta = (\mu')^{2d-2}/2$  for  $A_{\text{Strip}}$  and  $\eta' = \eta/3$ .

**Parameterization of an optimal solution packed by  $A_{\text{Strip}}$ .** Packing an optimal solution  $I_{\text{opt}}$  with  $A_{\text{Strip}}$  gives a packing  $P_{\text{Strip}}$  of height

$$\begin{aligned} h(P_{\text{Strip}}) &\leq (1 + \eta)\text{OPT}_{\text{Strip}}(I_{\text{opt}}) + Z_{\text{Strip}} && \text{by Theorem 25} \\ &\leq (1 + \eta)b'_d + \mu'b'_d \\ &\leq (1 + 2\mu')b'_d, \end{aligned}$$

since  $Z_{\text{Strip}} \leq \mu'b'_d$  by Inequality (11),  $\text{OPT}_{\text{Strip}}(I_{\text{opt}}) \leq b'_d$  and  $\eta \leq \mu'$ .

Moreover, we already know about the packing  $P_{\text{Strip}}$  that there are at most  $L_{\text{Strip}}$  layers, each associated with a packing pattern and possibly including empty ones for the small and medium items.

Now we express the structure of  $P_{\text{Strip}}$  in parameters. There are  $k \leq L_{\text{Strip}}$  layers and each layer corresponds to a packing pattern  $T_i = (T_{i1}, \dots, T_{i\ell})$  with  $\ell \leq K_{\text{Strip}}$ , of which we already know that they can be computed in constant time and their number is bounded by a constant. Furthermore, each layer has height  $l_i = x_i^* + 1$  that is given by the solution of the linear program. We are interested in the solution of the linear program  $x_i^*$  as a parameter, which is in  $\mathbb{R}$ . By rounding to the next-largest number in

$$R = \left\{ \frac{t\mu'}{L_{\text{Strip}}} \cdot b'_d \mid t = 1, \dots, \lceil L_{\text{Strip}}/\mu' \rceil \right\},$$

the heights  $x_i^*$  are discretized such that the overall height increases by at most

$$k \cdot \frac{\mu'}{L_{\text{Strip}}} b'_d \leq L_{\text{Strip}} \cdot \frac{\mu'}{L_{\text{Strip}}} b'_d = \mu'b'_d.$$

Let  $\tilde{P}$  denote the resulting packing with a height of

$$\tilde{h} \leq h(P_{\text{Strip}}) + \mu'b'_d \leq (1 + 3\mu')b'_d. \quad (14)$$

**Guessing the structure of  $\tilde{P}$ .** Our aim is to guess the structure of  $\tilde{P}$  that is expressed in parameters. By guessing we mean to apply a (huge) enumeration on all possible parameters and execute the whole remaining algorithm on each combination of parameters. Eventually we consider the parameters of an optimal solution packed by  $A_{\text{Strip}}$ . The whole enumeration can be done in polynomial time if the number of parameters is constant and each parameter is chosen from a polynomial number of possibilities. It is easy to see that this is the case for the following parameters.

Guess the separation index  $i^*$ , a selection of  $\ell \leq K_{\text{Strip}}$  threshold items  $a_1^* \geq \dots \geq a_\ell^*$ , the number of layers  $k \leq L_{\text{Strip}}$ , the selection of  $k$  packing patterns  $T_1, \dots, T_k$  and their associated layer heights  $x_i^* \in R$ . We assume that all these values from an optimal solution are known (alternatively: we are in the iteration step where these values are selected). Let the sets of large, medium and small items,  $L$ ,  $M$  and  $S$ , respectively, be defined by  $i^*$  as in the previous section.

**Selection of items.** Let  $\text{Knapsack}(S, K, \epsilon)$  be a solution with accuracy  $\epsilon$  for the one-dimensional knapsack instance with items  $S$  and knapsack size  $K$  found by a  $\mathcal{FPITAS}$  from [24]. We want to approximate the groups  $G_i$  in algorithm  $A_{\text{Strip}}$  with one-dimensional knapsack instances. To do this assume that  $a_1^* > \dots > a_\ell^*$ , i.e., the items of the groups  $G_i$  have different sizes. This assumption is valid as for threshold items  $a_i^* = \dots = a_j^*$  ( $i < j$ ) of equal size we can merge the corresponding groups of  $G_i, \dots, G_j$  and also combine the corresponding counters in all packing patterns  $T_l$  for  $l \in \{1, \dots, k\}$ . Let  $L_i = \{a_i \in L \mid a_i \in (a_{i+1}^*, a_i^*]\}$  denote the items between the sizes of the threshold items.  $\tilde{h}_i = \sum_{j=1}^k T_{ji} x_j^*$  is the overall height in  $\tilde{P}$  of the items in  $L_i \cap L_{\text{opt}}$  and gives the knapsack size for the selection  $L'_i = \text{Knapsack}(L_i, \tilde{h}_i, \mu')$ , where the items are given by their heights  $a_i$ .

For the small items we consider the volume bound  $\tilde{V}_S = \sum_{i=1}^k \text{VOL}_{R_i} = \sum_{i=1}^k (\text{VOL}_{Q'} - c_i) x_i^*$ , where  $c_i = \sum_{j=1}^{\ell} T_{ij} (a_j^*)^{d-1}$  is again the volume of the packing pattern  $T_i$ . We select  $S' = \text{Knapsack}(S, \tilde{V}_S, \mu')$ , where the items are given by their volume  $a_i^d$ .

The volume bound of  $\tilde{V}_M = \eta' \text{Vol}(B') \leq \mu' \text{Vol}(B')$  for the medium items is derived by the separation step. We select  $M' = \text{Knapsack}(M, \tilde{V}_M, \mu')$ , where the items are given by their volume  $a_i^d$  again.

Note that  $L_i \cap L_{\text{opt}}$  corresponds to the group  $G_i$  in  $A_{\text{Strip}}$ , whereas our approximation  $L'_i$  does not correspond to it. In particular,  $L'_i$  is not made up of a fixed number of items. Nevertheless we can approximate  $I_{\text{opt}}$  with  $L' = L'_1 \cup \dots \cup L'_k$ ,  $M'$  and  $S'$  as the following lemma shows.

**Lemma 27.** *The profit of the selection  $I' = L' \cup M' \cup S'$  is*

$$p(I') \geq (1 - \mu') \text{OPT}'(I).$$

*Proof.* We get  $p(L'_i) \geq (1 - \mu') p(L_i \cap L_{\text{opt}})$  and thus  $p(L') \geq (1 - \mu') p(L_{\text{opt}})$ .  $p(M') \geq (1 - \mu') p(M_{\text{opt}})$  and  $p(S') \geq (1 - \mu') p(S_{\text{opt}})$  complete the claim.  $\square$

**Packing the selection  $I'$ .** We demonstrate that  $L'$ ,  $M'$  and  $S'$  are suitable approximations of  $I_{\text{opt}}$  by giving a packing  $P'$  using the guessed parameters for  $\tilde{P}$ . We proceed as in Algorithm  $A_{\text{Strip}}$  by packing  $L'$  and adding  $S'$  into the gaps of the packing. In order to pack  $S'$  and  $M'$  completely we possibly need to enlarge the packing. We start with the packing of  $L'$ .

Use the guessed parameters to create a strip with the same structure as  $\tilde{P}$ . Thus create  $k$  layers with heights  $l_i = x_i^* + 1$  and columns with  $(d - 1)$ -dimensional basis in each layer that correspond to the packing pattern. Add the sets  $L'_i$  greedily into the corresponding columns. If the space in a column does not suffice, continue with the next column. All items can be packed since the overall height of  $L'_i$  is bound by  $\tilde{h}_i = \sum_{j=1}^k T_{ji}x_j^*$ , but a height of  $l_i = x_i^* + 1$  is available in each column. Denote the packing by  $P'$ . The remaining space in  $P'$  that is available for the packing of  $S'$  corresponds to the remaining space in  $\tilde{P}$ . It is the space  $R_i$  in each layer that is defined by the free space in packing pattern  $T_i$  and the height  $x_i^*$ . Since we used this remaining space as the volume bound but NFDH still leaves space free, we have to pack remaining items into an additional layer. Another additional layer is needed to pack the medium items  $M'$  with NFDH. These two additional layers make up the difference between the structure of  $\tilde{P}$  and  $P'$ .

Thus add  $S'$  with NFDH into the free spaces  $R_1, \dots, R_k$  and pack possibly remaining items into an additional layer  $L'_S$  above  $P'$ . Pack the medium items  $M'$  into another additional layer  $L'_M$ . The following lemma shows that the heights  $h'_S$  and  $h'_M$  of the additional layers are small.

**Lemma 28.** *If NFDH is used to pack the small items  $S'$  into the free space as well as into an additional layer  $L'_S$  and the medium items  $M'$  into another additional layer  $L'_M$  above  $P'$ , then the heights of the additional layers are bounded as follows*

$$\begin{aligned} h'_S &\leq 3\mu'b'_d \\ h'_M &\leq 2\mu'b'_d. \end{aligned}$$

*Proof.* With Lemma 22 we get

$$\text{Vol}(M') \geq \text{Vol}(L'_M) - \eta' \text{Vol}_{Q'} h'_M - 2 = \text{Vol}_{Q'} h'_M - \eta' \text{Vol}_{Q'} h'_M - 2$$

Using  $\text{Vol}(M') \leq \eta' \text{Vol}(B')$  we get the desired result for  $h'_M$ :

$$\begin{aligned} h'_M(\text{Vol}_{Q'} - \eta' \text{Vol}_{Q'}) &\leq \text{Vol}(M') + 2 \leq \eta' \text{Vol}(B') + 2 \leq \eta' \text{Vol}_{Q'} b'_d + 2 \\ \Rightarrow h'_M &\leq \frac{\eta'}{1 - \eta'} b'_d + \frac{2}{\text{Vol}_{Q'} - \eta' \text{Vol}_{Q'}} \\ &\leq 2\eta' b'_d \leq 2\mu' b'_d. \end{aligned}$$

In the second to last step we used  $\text{VOL}_{Q'} \geq 1$ ,  $\eta' < 1/2$  and  $b'_d \geq Z_{\text{Strip}}/\mu' \geq 8/\eta'$  to get  $1/(\text{VOL}_{Q'} - \eta'\text{VOL}_{Q'}) \leq 4 \leq \eta'b'_d/2$ . Furthermore with  $\eta' \leq 1/3$  we get  $\eta'/(1 - \eta') \leq (3/2)\eta'$ .

To bound the height of layer  $L'_S$ , let  $V'_S$  be the total volume of items from  $S'$  that are packed into this layer. With Lemma 21 we have

$$V'_S \geq \text{Vol}(L'_S) - \eta'\text{VOL}_{Q'}h'_S - 1 = \text{VOL}_{Q'}h'_S - \eta'\text{VOL}_{Q'}h'_S - 1.$$

Accordingly we get the following bound

$$h'_S \leq \frac{V'_S + 1}{\text{VOL}_{Q'} - \eta'\text{VOL}_{Q'}}. \quad (15)$$

In the next step we bound  $V'_S$ , i.e., the volume of items from  $S'$  that are packed into the additional layer. From Lemma 20 we get that either all items are packed (and thus  $h'_S = 0$ ) or the total free volume  $V'_i$  in a layer  $L_i$  for  $i \in \{1, \dots, k\}$  after packing the small items from  $S'$  with NFDH is  $V'_i \leq \eta'\text{VOL}_{Q'}x_i^* + 1$ . As the overall free volume was used as an upper bound for the volume of  $S'$  in the selection step we get

$$V'_S \leq \sum_{i=1}^k V'_i \leq \sum_{i=1}^k (\eta'\text{VOL}_{Q'}x_i^* + 1) \leq \eta'\text{VOL}_{Q'}\tilde{h} + k. \quad (16)$$

From Inequality (15) and Inequality (16) we get

$$\begin{aligned} h'_S &\leq \frac{\eta'\text{VOL}_{Q'}\tilde{h} + k + 1}{\text{VOL}_{Q'} - \eta'\text{VOL}_{Q'}} \\ &\leq \frac{\eta'}{1 - \eta'}\tilde{h} + \frac{k + 1}{\text{VOL}_{Q'} - \eta'\text{VOL}_{Q'}}. \end{aligned}$$

To further bound the second term we use  $\text{VOL}_{Q'} \geq 1$  and  $k \leq L_{\text{Strip}}$  and get

$$\frac{k + 1}{\text{VOL}_{Q'}(1 - \eta')} \leq \frac{L_{\text{Strip}} + 1}{1 - \eta'} = \frac{K_{\text{Strip}} + 3}{1 - \eta'} \leq Z_{\text{Strip}}.$$

The last steps follow by definition of  $L_{\text{Strip}}$ ,  $K_{\text{Strip}}$  and  $Z_{\text{Strip}}$ . With  $\tilde{h} \leq 1 + 3\mu'b'_d$  by Inequality (14) and with  $Z_{\text{Strip}} \leq \mu'b'_d$  by Inequality (11) we get

$$h'_S \leq \frac{\eta'}{1 - \eta'}\tilde{h} + Z_{\text{Strip}} \leq \frac{\eta'}{1 - \eta'}(1 + 3\mu')b'_d + \mu'b'_d \leq 3\mu'b'_d.$$

Here the last step follows with  $\eta'/(1 - \eta') \leq (3/2)\eta'$  for  $\eta' \leq 1/3$  and  $\mu' \leq 1/9$  and  $\eta' \leq \mu'$ .  $\square$

We summarize:  $P'$  is a packing of  $L'$ ,  $M'$  and  $S'$  with height  $h' \leq \tilde{h} + h'_M + h'_S \leq (1 + 8\mu')b'_d$  (see Inequality 14 and Lemma 28) and  $P'$  can be computed

without the knowledge of  $I_{\text{opt}}$ , by guessing the parameters of  $\tilde{P}$ . The profit of  $I' = L' \cup M' \cup S'$  is  $p(I') \geq (1 - \mu')\text{OPT}'(I)$  (see Lemma 27).

**Reduction of height  $h'$  and packing into  $B$ .** Use the shifting technique of Lemma 7 with  $\hat{h} = 8\mu'b'_d$  and  $\delta = 8\mu' \leq 1/2$  (as  $\mu' \leq 1/16$ ) to free the overhang of at most  $8\mu'b'_d$ . As we consider the full height  $h = (1 + 8\mu')b'_d$  we have  $h > b'_d > 1/\delta$ . Furthermore we have  $\delta h = 8\mu'b'_d + 64(\mu')^2b'_d \geq 8\mu'b'_d + 1$  since  $b'_d \geq 1/(\mu')^d$  and  $d \geq 2$ . Thus  $\hat{h} = 8\mu'b'_d \leq \delta h - 1$ . Finally, transfer the packing into  $B'$  into a packing into  $B$  as described before. Let  $I''$  be the set of items after reducing the height and let  $I'''$  be the set of items that is finally packed into  $B$ . The profit of  $I'''$  can be estimated as follows.

$$\begin{aligned}
p(I''') &\geq (1 - 4\mu')p(I'') && \text{by (13)} \\
&\geq (1 - 4\mu')(1 - 4 \cdot 8\mu')p(I') && \text{reducing } h' \\
&\geq (1 - 4\mu')(1 - 32\mu')(1 - \mu')\text{OPT}'(I) && \text{by Lemma 27} \\
&\geq (1 - 4\mu')(1 - 32\mu')(1 - \mu')(1 - 4(d - 1)\mu')\text{OPT}(I) && \text{by (12)} \\
&\geq (1 - (4d + 33)\mu')\text{OPT}(I).
\end{aligned}$$

We insert  $\mu' = \mu/(4d + 34)$  and sum up our result in Algorithm 6 and the following theorem.

---

**Algorithm 6** Approximation algorithm  $A_{\text{LR}}$  for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES

---

- 1: Reassemble bin  $B$  to a strip-like bin  $B'$
  - 2: **for all**  $i \in \{1 \dots, r\}$ , selections of threshold items  $a_1^* \geq \dots \geq a_\ell^*$ , (for  $\ell \leq K_{\text{Strip}}$ ), numbers of layers  $k \leq L_{\text{Strip}}$ , selections of packing patterns  $T_1, \dots, T_k$  and layer heights  $x_1^*, \dots, x_k^* \in R$  **do**
  - 3:     compute  $L'_j, M'$  and  $S'$  for  $j = 1, \dots, k$
  - 4:     pack  $L'_j, M'$  and  $S'$  into a strip of height  $h'$
  - 5:     **if**  $h' \leq (1 + 8\mu)b'_d$  **then**
  - 6:         reduce the height of the packing with the shifting technique
  - 7:         change the packing into a packing for  $B$
  - 8:         keep the solution
  - 9: select the solution with highest profit that was encountered
- 

Let  $V_{\text{LR}} = Z_{\text{Strip}}/(\mu')^d$  denote the lower bound for the volume of the bin. As defined we have

$$V_{\text{LR}} = \frac{Z_{\text{Strip}}}{(\mu')^d} = \frac{1}{1 - (\mu')^{2d-2}/6} \left( 2 \left\lceil \frac{6}{(\mu')^{3d-3}\ell_r^d} \right\rceil + 4 \right) \frac{1}{(\mu')^d}, \quad (17)$$

with  $\ell_r$  being a lower bound for  $\alpha_r$  for  $r = \lceil 1/\eta' \rceil$ ,  $\eta' = \eta/3$  and  $\eta = (\mu')^{2d-2}/2$ . Thus  $V_{\text{LR}}$  is a constant only depending on  $d$  and  $\mu$  since  $\mu' = \mu/(4d + 34)$ .

For  $\mu \leq 1/(4d + 33)$  we have  $\mu' \leq 1/((4d + 33)(4d + 34))$  and  $1/(1 - (4d + 33)\mu') \leq 1 + (4d + 34)\mu' = 1 + \mu$  and we thus get the following theorem for any fixed dimension  $d \geq 2$ .

**Theorem 29.** *For any fixed accuracy  $0 < \mu \leq 1/(4d + 33)$  and any given set  $I = \{a_1, \dots, a_n\}$  of  $d$ -dimensional cubes  $a_i \leq 1$  with profits  $p_i > 0$  and  $d$ -dimensional bin  $B = (b_1, \dots, b_d)$  with side lengths  $1 \leq b_1 \leq \dots \leq b_d$  and volume  $\text{VOL}_B \geq V_{\text{LR}}$ , where  $V_{\text{LR}}$  is a constant only depending on  $d$  and  $\mu$  as defined above, Algorithm  $A_{\text{LR}}$  is a polynomial time algorithm for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES with approximation ratio  $1 + \mu$ .*

## 6 Hypercube Knapsack Packing

Now we are ready to present our main result, a  $(1 + 1/2^d + \rho)$ -approximation algorithm for HYPERCUBE KNAPSACK PACKING. In the SQUARE PACKING algorithm we considered three different cases, packing with *enough remaining space*, packing with *several large items* and packing with *only one large item*. The latter case was motivated by the observation that three squares cannot fill a unit bin almost completely unless one of the squares is huge. This observation is generalized to a number of  $2^d - 1$  hypercubes in the  $d$ -dimensional case. For a fixed dimension  $d \geq 2$  the problem is stated as follows:

**Definition 30.** Given a set  $I = \{a_1, \dots, a_n\}$  of  $d$ -dimensional cubes  $a_i \leq 1$  with profits  $p_i > 0$ . The HYPERCUBE KNAPSACK PACKING PROBLEM asks for a packing  $P$  of a selection  $I' \subseteq I$  into the unit cube with maximal profit.

**Outline.** First, we give new parameters for the separation step such that the first two cases hold for  $d$ -dimensional packing. Second, we observe that for a number of up to  $2^d - 1$  hypercubes, either the remaining space is big enough or there is only one very large item. Finally, we show how to handle the third case, applying Algorithm  $A_{\text{LR}}$  from the previous section.

**Separation Technique.** Let  $0 < \rho \leq 1/(2^d + 1)$  and  $\rho' = \rho/4$  and let  $V_{\text{LR}}$  be the lower bound for the volume of the bin from Section 5 using  $\mu = \rho'$  and  $d$  for the definition in (17). Thus  $V_{\text{LR}}$  is a constant only depending on  $d$  and  $\rho$  since  $\rho' = \rho/4$ . Note that  $V_{\text{LR}} = Z_{\text{Strip}}/(\mu')^d \geq 2^d$  as  $Z_{\text{Strip}} \geq 1$  and  $\mu' \leq 1/2$ . We apply the usual separation technique with  $r = \lceil 1/\rho' \rceil$ ,  $\alpha_1 = 1/V_{\text{LR}}$  and  $\alpha_{i+1} = \alpha_i^{d^2+d}/(d2^d)$  for  $i = 0, \dots, r$ . As in Section 3 we consider an optimal solution  $I_{\text{opt}}$  and assume the knowledge of an optimal separation parameter

$i^*$  and an optimal packing  $P_{L_{\text{opt}}}$  of a set  $L_{\text{opt}}$  from now on. As in the two-dimensional case we have

$$\text{OPT}(L_{\text{opt}} \cup S) \geq (1 - \rho')\text{OPT}(I). \quad (18)$$

Since  $|L_{\text{opt}}| \leq 1/\alpha_{i^*}^d$ , the free space in  $P_{L_{\text{opt}}}$  can be partitioned into at most  $(2/\alpha_{i^*}^d)^d$  gaps—see Lemma 2. Similar to Lemma 11 we get the following bound on the free volume for adding items from a subset  $S' \subseteq S$  with NFDH into these gaps.

**Lemma 31.** *If NFDH does not pack all items from  $S'$  into the gaps, then the overall free volume  $V$  is at most  $\alpha_{i^*}^d$ .*

*Proof.* Lemma 4 yields that packing small items from  $S'$  with NFDH into these gaps either packs all items or leaves a free volume of at most  $d\alpha_{i^*+1}$  in each gap. If items remain unpacked, the overall free volume  $V$  is bounded by

$$V \leq \frac{2^d}{\alpha_{i^*}^{d^2}} \cdot d\alpha_{i^*+1} = \frac{d2^d}{\alpha_{i^*}^{d^2}} \cdot \frac{\alpha_{i^*}^{d^2+d}}{d2^d} \leq \alpha_{i^*}^d$$

Again this is also a lower bound for the volume of an item in  $L_{\text{opt}}$ .  $\square$

We consider the following three cases: 1) *enough remaining space* for the small items ( $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$ ), 2) *several large items* ( $|L_{\text{opt}}| = k \geq 2^d$ ), and 3) *one very large item* ( $a_{\text{max}} \geq 1 - 1/V_{\text{LR}}$ ), where  $a_{\text{max}}$  is the largest item in  $L_{\text{opt}}$ .

We want to show a generalized version of Lemma 12, i.e., these three cases cover all possibilities. Before we do this, we introduce the function  $f_d(x) = x^d + (2^d - 2)(1 - x)^d$  and show the following claim.

**Claim 32.** *We have  $f_d(\frac{3}{4}) \leq f_{d-1}(\frac{3}{4})$  for  $d \geq 3$ .*

*Proof.*

$$\begin{aligned} & f_d\left(\frac{3}{4}\right) \leq f_{d-1}\left(\frac{3}{4}\right) \\ \Leftrightarrow & \left(\frac{3}{4}\right)^d + (2^d - 2)\left(\frac{1}{4}\right)^d \leq \left(\frac{3}{4}\right)^{d-1} + (2^{d-1} - 2)\left(\frac{1}{4}\right)^{d-1} \\ \Leftrightarrow & \frac{3^d + 2^d - 2}{4^d} \leq \frac{3^{d-1} + 2^{d-1} - 2}{4^{d-1}} \\ \Leftrightarrow & 3^d + 2^d - 2 \leq (3^{d-1} + 2^{d-1} - 2)4 \\ & = 3 \cdot 3^{d-1} + 3^{d-1} + 2 \cdot 2^{d-1} + 2 \cdot 2^{d-1} - 8 \\ & = 3^d + 2^d + 2^d + 3^{d-1} - 8 \end{aligned}$$

and this is true since  $d \geq 3$  and thus  $2^d \geq 8$  and  $3^{d-1} \geq 9$ .  $\square$

Now we are ready to give the generalized version of Lemma 12.

**Lemma 33.** *If  $|L_{\text{opt}}| < 2^d$ , then  $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$  or  $a_{\text{max}} \geq 1 - 1/V_{\text{LR}}$ .*

*Proof.* Recall that  $V_{\text{LR}} \geq 2^d$  and thus  $\alpha_1 = 1/V_{\text{LR}} \leq 1/2^d$ . If  $L_{\text{opt}} = \emptyset$ , then  $\text{Vol}(L_{\text{opt}}) = 0$ . Suppose that  $|L_{\text{opt}}| \in \{1, \dots, 2^d - 1\}$ . If  $a_{\text{max}} \leq 1/2$ , then  $\text{Vol}(L_{\text{opt}}) \leq 1 - 1/2^d \leq 1 - \alpha_{i^*}$ . Otherwise  $a_{\text{max}} > 1/2$  and the smaller items have a maximum size of  $1 - a_{\text{max}}$  to be feasible with  $a_{\text{max}}$ . Thus a maximum volume is achieved for  $2^d - 2$  additional items of size  $1 - a_{\text{max}}$ . For  $x = a_{\text{max}}$  the function  $f_d(x) = x^d + (2^d - 2)(1 - x)^d$  that we introduced before is an upper bound for the volume  $\text{Vol}(L_{\text{opt}})$ . We now show that  $f_d(x) \leq 1 - 1/V_{\text{LR}}$  for  $x \in [1/2, 1 - 1/V_{\text{LR}}]$ .

The upper bound  $f_d$  is convex since the second derivative  $f_d(x)'' = (d - 1)dx^{d-2} + (d - 1)d(2^d - 2)(1 - x)^{d-2}$  is positive on the domain  $[1/2, 1]$  as  $d \geq 2$ . Therefore the maxima occur on the left and right sides of the interval. Namely,  $f_d(1/2) = 1 - 1/2^d$  and  $f_d(1 - 1/V_{\text{LR}})$ . To calculate the latter value we use that  $f_d(\lambda x + (1 - \lambda)y) \leq \lambda f_d(x) + (1 - \lambda)f_d(y)$  for convex functions. We have  $f_d(3/4) \leq f_2(3/4) = 11/16$  by Claim 32 and  $f_d(1) = 1$ . Thus for  $\lambda = 4/V_{\text{LR}}$  we get

$$\begin{aligned} f_d\left(1 - \frac{1}{V_{\text{LR}}}\right) &= f_d\left(\frac{4}{V_{\text{LR}}} \cdot \frac{3}{4} + \left(1 - \frac{4}{V_{\text{LR}}}\right) \cdot 1\right) \\ &\leq \frac{4}{V_{\text{LR}}} f_d\left(\frac{3}{4}\right) + \left(1 - \frac{4}{V_{\text{LR}}}\right) f_d(1) \\ &= \frac{4}{V_{\text{LR}}} \frac{11}{16} + \left(1 - \frac{4}{V_{\text{LR}}}\right) \cdot 1 \\ &\leq 1 - \frac{1}{V_{\text{LR}}}. \end{aligned}$$

Thus  $a_{\text{max}} > 1 - 1/V_{\text{LR}}$  or  $\text{Vol}(L_{\text{opt}}) \leq 1 - 1/V_{\text{LR}} \leq 1 - \alpha_{i^*}$ .  $\square$

With Lemma 31 the first two cases can be handled similarly to the two-dimensional case—see Lemma 13, using a knapsack size of  $V - \alpha_{i^*}^d - \alpha_{i^*+1}^d$ , and Lemma 14. We get the following corollaries.

**Corollary 34. *Enough Remaining Space***

*If  $\text{Vol}(L_{\text{opt}}) \leq 1 - \alpha_{i^*}$ , we can find a selection  $S' \subseteq S$  of small items in polynomial time such that  $L_{\text{opt}} \cup S'$  is feasible and  $p(L_{\text{opt}} \cup S') \geq (1 - 2\rho')\text{OPT}(I)$ .*

**Corollary 35. *Several Large Items***

*If  $|L_{\text{opt}}| = k$ , we can find a selection  $S' \subseteq S$  of small items in polynomial time such that  $L_{\text{opt}} \cup S'$  is feasible and  $p(L_{\text{opt}} \cup S') \geq (k/(k + 1) - 2\rho')\text{OPT}(I)$ .*

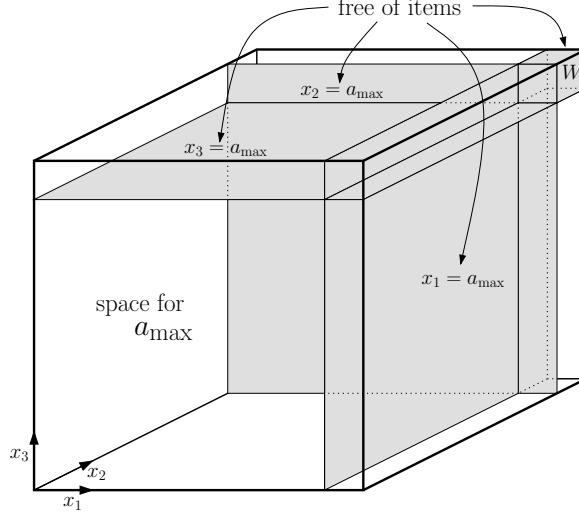


Fig. 10. A *well-structured* packing for  $d = 3$

As a last step we show how an almost optimal packing can be derived for  $a_{\max} \geq 1 - 1/V_{\text{LR}}$ . First, we show that a special packing structure does not change the optimal value significantly. Second, we use the shifting technique and some rotations to apply Theorem 29 for HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES to find an almost optimal packing for the remaining items. As for  $a_{\max} = 1$  no other item fits into the bin we can disregard this case. For the remainder of this section we assume that  $1 > a_{\max} \geq 1 - 1/V_{\text{LR}}$ .

**Well-structured Packing.** A packing  $P$  is called *well-structured* if the following conditions are satisfied:

- 1) the largest item  $a_{\max}$  is located in the origin  $(0, \dots, 0)$ ,
- 2) for each item in the packing, the intersection of the interior of the item and the hypercube space  $W$  of size  $1 - a_{\max}$  at position  $(a_{\max}, \dots, a_{\max})$  is empty and
- 3) for each item in the packing, the intersection of the interior of the item and any hyperplane defined by the facets of  $a_{\max}$  are empty.

See Figure 10 for a well-structured packing. Let  $\text{OPT}_{\text{WS}}(I)$  denote the optimal profit of a well-structured packing. The following lemma shows that it is sufficient to approximate well-structured packings.

**Lemma 36.** *We have  $\text{OPT}_{\text{WS}}(I) \geq (1 - \rho')\text{OPT}(I)$ .*

*Proof.* We show how an optimal packing can be transformed to a well-structured packing losing not more than  $\rho'\text{OPT}(I)$  of the profit. Consider an optimal packing of  $I_{\text{opt}}$  where  $a_{\max}$  is placed at  $(x_1^*, \dots, x_d^*) \neq (0, \dots, 0)$ , i.e.,  $a_{\max}$  is not placed in the origin. Note that the free space to all sides has

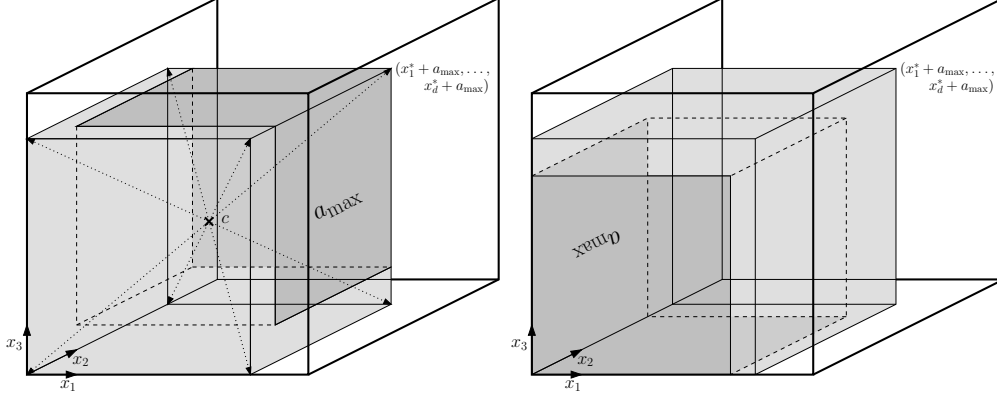


Fig. 11. Point reflection of  $[0, x_1^* + a_{\max}] \times \cdots \times [0, x_d^* + a_{\max}]$  at its center  $c = ((x_1^* + a_{\max})/2, \dots, (x_d^* + a_{\max})/2)$  for  $d = 3$

length at most  $1 - a_{\max} \leq 1/V_{\text{LR}}$ . Our goal is to perform a point reflection of the entire cuboid  $[0, x_1^* + a_{\max}] \times \cdots \times [0, x_d^* + a_{\max}]$  at its center  $c = ((x_1^* + a_{\max})/2, \dots, (x_d^* + a_{\max})/2)$  as illustrated in Figure 11. Reflections of packings are possible since all items are hypercubes. To accomplish the reflection we have to ensure that no item in the packing intersects with parts of the hyperplanes  $x_i = x_i^* + a_{\max}$ . The left side of Figure 11 shows which parts of the hyperplanes have to be free. As some of the intersecting items might have high profit we cannot remove them directly, but need to apply the shifting technique of Lemma 6. To ease the presentation we are going to free the complete hyperplanes  $x_i = x_i^* + a_{\max}$  instead of just parts of them—see Figure 12. This is the reason why we introduced the shifting technique for rectilinear basis  $Q$  allowing holes.

Consider the *open strip packing* in the space around  $a_{\max}$  for  $x_i \in [x_i^*, x_i^* + a_{\max}]$  for each direction  $i = 1, \dots, d$ —see Figure 12. The height of the open strip packing is  $a_{\max}$  and the size of the small items is at most  $1 - a_{\max}$ . The shifting technique from Lemma 6 is applicable for  $\delta = 1 - a_{\max}$  as  $a_{\max} \geq 1 - 1/V_{\text{LR}} \geq 1 - 1/2^d \geq 3/4$  and thus  $\delta \leq 1/4$  and  $a_{\max} \geq 1/2 + \delta$ . We obtain a packing without any item intersecting the hyperplane  $x_i = x_i^* + a_{\max}$ . As items are only moved along the current direction  $i$ , the  $x_j$ -coordinates are not changed for  $j \neq i$ . Therefore, after applying the shifting technique for all directions  $i = 1, \dots, d$ , all hyperplanes  $x_i = x_i^* + a_{\max}$  are free of items. The total loss is bounded by  $4d(1 - a_{\max})\text{OPT}(I)$  (Lemma 6). The described point reflection is possible without further loss of profit.

So far we derived an almost optimal packing with  $a_{\max}$  placed at the origin. The hyperplanes  $x_i = a_{\max}$  can be freed similar to the previous step losing not more than  $4d(1 - a_{\max})\text{OPT}(I)$  again. Finally, the hypercube  $W$  of size  $1 - a_{\max}$  at position  $(a_{\max}, \dots, a_{\max})$  is freed with the shifting technique of Lemma 7. To do this consider the strip  $T$  of size  $(1 - a_{\max}, \dots, 1 - a_{\max}, 1)$  at position  $(a_{\max}, \dots, a_{\max}, 0)$  ( $T$  includes the hypercube  $W$  at its top). By

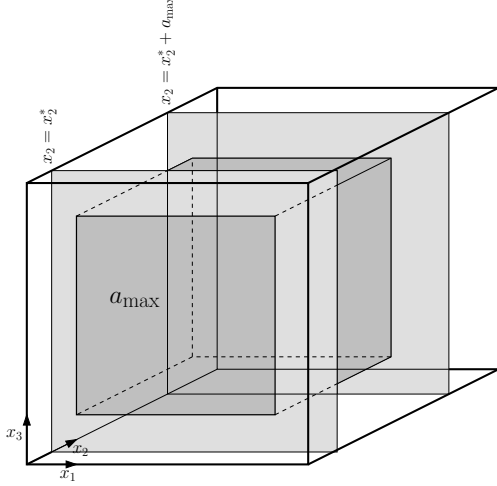


Fig. 12. The open strip packing between  $x_2 = x_2^*$  and  $x_2 = x_2^* + a_{\max}$ . Items intersecting  $x_2 = x_2^* + a_{\max}$  are moved

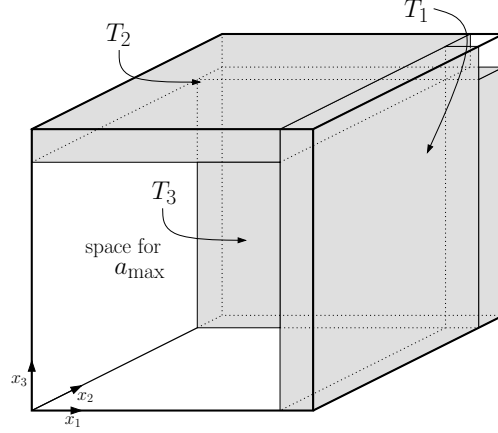


Fig. 13. Division of the remaining space into  $T_1, \dots, T_d$  for  $d = 3$

scaling  $T$  and all items that intersect  $T$  with  $1/(1 - a_{\max})$  we get a strip of size  $(1, \dots, 1, 1/(1 - a_{\max}))$  and the items have size at most 1. For  $\delta = 2(1 - a_{\max}) \leq 1/2$  (as  $1 - a_{\max} \leq 1/2^d$  and  $d \geq 2$ ) we have  $h = 1/(1 - a_{\max}) \geq 1/\delta = 1/(2(1 - a_{\max}))$  and for the height  $\hat{h}$  of  $W$  we have  $\hat{h} = 1 \leq \delta h - 1 = 2 - 1$ . Therefore we can apply Lemma 7 on this scaled strip and free  $W$ . Since  $1 - a_{\max} \leq 1/V_{\text{LR}}$  and  $V_{\text{LR}} = Z_{\text{Strip}}/(\mu')^d \geq (4d + 34)^d/\mu^d = (4d + 34)^d/(\rho')^d$  we have  $1 - a_{\max} \leq (\rho')^d/(4d + 34)^d$ . Thus with Lemma 7 we get an estimation for the overall loss of profit of at most  $(2 \cdot 4d(1 - a_{\max}) + 4 \cdot 2(1 - a_{\max}))\text{OPT}(I) = (8d + 8)(1 - a_{\max})\text{OPT}(I) \leq \rho'\text{OPT}(I)$ .  $\square$

We showed that it is sufficient to approximate well-structured packings. Therefore pack  $a_{\max}$  into the origin. In the following we show how to approximate a packing of  $R := S \cup L_{\text{opt}} \setminus \{a_{\max}\}$ , which is the set of remaining items, into the free space around  $a_{\max}$ .

**Applying Algorithm  $A_{\text{LR}}$ .** We cut and rotate the remaining space around  $a_{\max}$  such that it builds a cuboid bin that is much bigger than the remaining items. Then we apply Algorithm  $A_{\text{LR}}$  for HYPERCUBE KNPSACK PACKING FOR LARGE RESOURCES, and by cutting again and reassembling to the original position, a valid solution is derived.

Observe that the remaining space in the bin, with the exception of a hypercube  $W$  of size  $1 - a_{\max} \leq 1/V_{\text{LR}}$  in the opposite corner of the origin, can be divided into  $d$  differently rotated spaces  $T_1, \dots, T_d$ , each of size  $(1 - a_{\max}, a_{\max}, \dots, a_{\max}, 1)$ —see Figure 13. In a well-structured packing all items are completely included in one of these spaces. Rotate all spaces into the same

orientation and assemble them to a bin  $Q$  of size  $(1 - a_{\max}, a_{\max}, \dots, a_{\max}, d)$ . Scaling  $Q$  and the remaining items in  $R$  by  $1/(1 - a_{\max})$  gives an instance of HYPERCUBE KNAPSACK PACKING WITH LARGE RESOURCES where all items have size at most 1. The volume of the scaled bin is  $(a_{\max}/(1 - a_{\max})^{d-2}(d/(1 - a_{\max}))) \geq 1/(1 - a_{\max}) \geq V_{\text{LR}}$  (since  $a_{\max} \geq 1 - 1/V_{\text{LR}}$ ). So we can apply Algorithm  $A_{\text{LR}}$  and find a packing for a selection  $R' \subseteq R$  of items with profit  $p(R') \geq (1 - \rho')\text{OPT}_{\text{LR}}(R)$ , where  $\text{OPT}_{\text{LR}}$  denotes the optimal value for packing into the scaled bin. To avoid misinterpretation we again use  $p(a_{\max})$  to denote the profit of  $a_{\max}$ . Obviously

$$\text{OPT}_{\text{LR}}(R) + p(a_{\max}) \geq \text{OPT}_{\text{WS}}(L_{\text{opt}} \cup S). \quad (19)$$

Reassembling  $Q$  into the original shape requires an application of the shifting technique of Lemma 5 with  $d-1$  cutting hyperplanes. Note that all items have size at most 1. We use  $\delta = 1 - a_{\max}$  and thus we have  $h = d/(1 - a_{\max}) > 2$  (as  $d \geq 2$  and  $0 < a_{\max} < 1$ ) and  $t = d - 1 \leq \delta h = d$ . Thus by Lemma 5 the loss of profit is bounded by  $4(1 - a_{\max})p(R') \leq \rho'p(R')$  (as  $1 - a_{\max} \leq 1/V_{\text{LR}} \leq (\mu')^d \leq (\rho')^d$ ,  $\rho' \leq 1/2^d$  and  $d \geq 2$ ). Note that the packing that we actually derived is not well-structured since the spaces  $T_i$  intersect parts of the extensions of facets of  $a_{\max}$ . We get a final profit of

$$\begin{aligned} p &\geq (1 - \rho')p(R') + p(a_{\max}) && \text{reassembling the bin} \\ &\geq (1 - \rho')(1 - \rho')\text{OPT}_{\text{LR}}(R) + p(a_{\max}) && \text{Theorem 29} \\ &\geq (1 - \rho')(1 - \rho')\text{OPT}_{\text{WS}}(L_{\text{opt}} \cup S) && \text{by (19)} \\ &\geq (1 - \rho')(1 - \rho')(1 - \rho')\text{OPT}(L_{\text{opt}} \cup S) && \text{Lemma 36} \\ &\geq (1 - \rho')(1 - \rho')(1 - \rho')(1 - \rho')\text{OPT}(I) && \text{by (18)} \\ &\geq (1 - 4\rho')\text{OPT}(I). \end{aligned}$$

This proves the following lemma.

**Lemma 37. One Very Large Item**

*If  $a_{\max} \geq 1 - 1/V_{\text{LR}}$ , we can find a selection  $R' \subseteq S \cup L_{\text{opt}} \setminus \{a_{\max}\}$  of items in polynomial time, such that  $\{a_{\max}\} \cup R'$  is feasible and  $p(\{a_{\max}\} \cup R') \geq (1 - 4\rho')\text{OPT}(I)$ .*

An algorithmic description of the method that is applied in this case is given in Algorithm 7.

The complete algorithm  $A_{\text{KP}}$  is given below.

---

**Algorithm 7** *One Very Large Item*

---

**Given:** Sets  $L$  and  $S$  where  $L$  is feasible and  $L$  contains  $a_{\max} \geq 1 - 1/V_{\text{LR}}$

- 1: pack  $a_{\max}$  into the origin of the bin
  - 2: let  $R := S \cup L \setminus \{a_{\max}\}$
  - 3: find an almost optimal packing of a selection  $R' \subseteq R$  into the bin  $(1 - a_{\max}, a_{\max}, \dots, a_{\max}, d)$  using Algorithm  $A_{\text{LR}}$
  - 4: use the shifting technique to cut the bin into  $d$  parts
  - 5: insert the parts of the packing into the free space around  $a_{\max}$
- 

---

**Algorithm 8**  $(1 + 1/2^d + \rho)$ -algorithm  $A_{\text{KP}}$  for HYPERCUBE KNAPSACK PACKING

---

- 1: **for all**  $i \in \{1, \dots, r\}$  and  $L \subseteq \{s \in I \mid s \geq \alpha_i\}$  with  $|L| \leq 1/\alpha_i^d$  **do**
  - 2:     check feasibility of  $L$  with *constant packing* method
  - 3:     **if**  $L$  is feasible **then**
  - 4:         let  $P_L$  be the packing of  $L$  from Step 2
  - 5:         **case**  $A(L) \leq 1 - \alpha_i$ : pack almost optimally with Algorithm 1
  - 6:         **case**  $a_{\max} \geq 1 - 1/V_{\text{LR}}$ : pack almost optimally with Algorithm 7
  - 7:         **case**  $|L| \geq 2^d$ : pack with Algorithm 2
  - 8:     return the packing with the highest profit that was encountered
- 

**Theorem 38.**  $A_{\text{KP}}$  is a polynomial time algorithm for HYPERCUBE KNAPSACK PACKING with performance ratio  $(1 + 1/2^d + \rho)$ .

*Proof.* Algorithm  $A_{\text{KP}}$  iterates over all  $i \in \{1, \dots, r\}$  and  $L \subseteq \{s \in I \mid s \geq \alpha_i\}$  with  $|L| \leq 1/\alpha_i^d$ . Eventually it considers the iteration where  $i = i^*$  and  $L = L_{\text{opt}}$ . Lemma 33 shows that  $|L| \geq 2^d$  or  $\text{Vol}(L) \leq 1 - \alpha_i$  or  $a_{\max} \geq 1 - 1/V_{\text{LR}}$ . Therefore one of the Algorithms 1, 2 or 7 from Corollary 34, 35 and Lemma 37 is applied on  $L$ . Since  $1 - 2\rho' \geq 1 - 4\rho' \geq 2^d/(2^d + 1) - 2\rho'$  for  $\rho' \leq 1/(2^{d+1} + 2)$ , the profit is at least  $(2^d/(2^d + 1) - 2\rho')\text{OPT}(I)$ . In the following we show that

$$\frac{1}{\frac{2^d}{2^d+1} - 2\rho'} \leq 1 + 1/2^d + \rho$$

which proves the theorem. Recall that  $\rho \leq 1/(2^d + 1)$  and  $\rho' = \rho/4$ . We have

$$\begin{aligned} & \frac{1}{\frac{2^d}{2^d+1} - 2\rho'} \leq 1 + 1/2^d + \rho \\ \Leftrightarrow & \quad 1 \leq \left(\frac{2^d+1}{2^d} + \rho\right) \left(\frac{2^d}{2^d+1} - \rho/2\right) \\ \Leftrightarrow & \quad \frac{\rho^2}{2} \leq \rho \left(\frac{2^d}{2^d+1} - \frac{2^d+1}{2^{d+1}}\right) \\ \Leftrightarrow & \quad \rho \leq \frac{2^{d+1}}{2^d+1} - \frac{2^d+1}{2^d}. \end{aligned}$$

This is true since

$$\begin{aligned} \rho \leq \frac{1}{2^d + 1} \quad \text{and} \quad \frac{2^{d+1}}{2^d + 1} - \frac{2^d + 1}{2^d} &= \frac{2^{d+1}2^d - (2^d + 1)(2^d + 1)}{(2^d + 1)2^d} \\ &= \frac{2^{2d+1} - 2^{2d} - 2^{d+1} - 1}{(2^d + 1)2^d} \\ &\geq \frac{2^d}{(2^d + 1)2^d} = \frac{1}{2^d + 1}. \end{aligned}$$

The last step holds since  $2^{2d+1} - 2^{2d} = 2^{2d}$  and  $2^{2d} - 2^{d+1} \geq 2^{d+1}$  as  $d \geq 2$ .  $\square$

## 7 Acknowledgments

I thank the anonymous referees for many valuable comments and suggestions regarding the presentation of this paper. Furthermore, I would like to thank Kazuo Iwama for his hospitality during my stay in Japan. I am grateful to Ingo Wegener for his encouragement. Rob van Stee strongly supported me with the final write-up.

This work would not have been done without the unconditional support of a very special person. Annalena, I thank you for this support, your understanding, your encouragement and your love.

## References

- [1] Nikhil Bansal, Alberto Caprara, and Maxim Sviridenko. Improved approximation algorithms for multidimensional bin packing problems. In *FOCS: Proc. 47nd IEEE Symposium on Foundations of Computer Science*, pages 697–708, 2006.
- [2] Nikhil Bansal, Jose 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.
- [3] Nikhil Bansal, Xin Han, Kazuo Iwama, Maxim Sviridenko, and Guochuan Zhang. Harmonic algorithm for 3-dimensional strip packing problem. In *SODA: Proc. 18th ACM-SIAM Symposium on Discrete Algorithms*, pages 1197–1206, 2007.
- [4] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*, volume 6. Athena Scientific, 1997.

- [5] Alberto Caprara, Andrea Lodi, and Michele Monaci. Fast approximation schemes for two-stage, two-dimensional bin packing. *Mathematics of Operations Research*, 30(1):150–172, 2005.
- [6] Alberto Caprara and Michele Monaci. On the two-dimensional knapsack problem. *Operations Research Letters*, 32(1):5–14, 2004.
- [7] Miroslav Chlebík and Janka Chlebíková. Inapproximability results for orthogonal rectangle packing problems with rotations. In *CIAC: Proc. 6th Conference on Algorithms and Complexity*, pages 199–210, 2006.
- [8] Edward G. Coffman, Jr., Michael R. Garey, and David S. Johnson. Approximation algorithms for bin packing: A survey. In *Approximation Algorithms for NP-hard Problems*, pages 46–93. PWS Publishing Company, 1997.
- [9] Wenceslas Fernandez de la Vega and George S. Lueker. Bin packing can be solved within  $(1 + \epsilon)$  in linear time. *Combinatorica*, 1(4):349–355, 1981.
- [10] Florian Diedrich, Rolf Harren, Klaus Jansen, Ralf Thöle, and Thomas Henning. Approximation algorithms for 3d orthogonal knapsack. In *TAMC: Proc. 4th International Conference on Theory and Applications of Models of Computation*, pages 34–45, 2007.
- [11] Aleksei V. Fishkin, Olga Gerber, and Klaus Jansen. On efficient weighted rectangle packing with large resources. In *ISAAC: Proc. 16th International Symposium on Algorithms and Computation*, pages 1039–1050, 2005.
- [12] Aleksei V. Fishkin, Olga Gerber, Klaus Jansen, and Roberto Solis-Oba. On packing squares with resource augmentation: Maximizing the profit. In *CATS: Proc. 11th Australasian Symposium on Theory of Computing*, pages 61–67, 2005.
- [13] Aleksei V. Fishkin, Olga Gerber, Klaus Jansen, and Roberto Solis-Oba. Packing weighted rectangles into a square. In *MFCS: Proc. 30th International Symposium on Mathematical Foundations of Computer Science*, pages 352–363, 2005.
- [14] Ari Freund and Joseph Naor. Approximating the advertisement placement problem. *Journal of Scheduling*, 7(5):365–374, 2004.
- [15] Rolf Harren. Approximating the orthogonal knapsack problem for hypercubes. In *ICALP: Proc. 33rd International Colloquium on Automata, Languages and Programming*, pages 238–249, 2006.
- [16] Oscar H. Ibarra and Chul E. Kim. Fast approximation for the knapsack and sum of subset problems. *Journal of ACM*, 22(4):463–468, 1975.
- [17] Klaus Jansen and Roberto Solis-Oba. An asymptotic approximation algorithm for 3d-strip packing. In *SODA: Proc. 17th ACM-SIAM Symposium on Discrete Algorithms*, pages 143–152, 2006.

- [18] Klaus Jansen and Roberto Solis-Oba. New approximability results for 2-dimensional packing problems. In *MFCS: Proc. 32nd International Symposium on Mathematical Foundations of Computer Science*, pages 103–114, 2007.
- [19] Klaus Jansen and Roberto Solis-Oba. A polynomial time approximation scheme for the square packing problem. In *IPCO: Proc. 13th Conference on Integer Programming and Combinatorial Optimization*, page to appear, 2008.
- [20] Klaus Jansen and Rob van Stee. On strip packing with rotations. In *STOC: Proc. 37th ACM Symposium on Theory of Computing*, pages 755–761, 2005.
- [21] Klaus Jansen and Guochuan Zhang. Maximizing the number of packed rectangles. In *SWAT: 9th Scandinavian Workshop on Algorithm Theory*, pages 362–371, 2004.
- [22] Klaus Jansen and Guochuan Zhang. Maximizing the total profit of rectangles packed into a rectangle. *Algorithmica*, 47(3):323–342, March 2007.
- [23] Narendra Karmarkar and Richard M. Karp. An efficient approximation scheme for the one-dimensional bin packing problem. In *FOCS: Proc. 23rd IEEE Symposium on Foundations of Computer Science*, pages 312–320, 1982.
- [24] Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer, 2004.
- [25] Claire Kenyon and Eric Rémy. A near optimal solution to a two-dimensional cutting stock problem. *Mathematics of Operations Research*, 25(4):645–656, 2000.
- [26] Bernhard H. Korte and Jens Vygen. *Combinatorial Optimization - Theory and Algorithms; Second Edition*. Springer, 2002.
- [27] Joseph Y.-T. Leung, Tommy W. Tam, C. S. Wong, Gilbert H. Young, and Francis Y.L. Chin. Packing squares into a square. *Journal of Parallel and Distributed Computing*, 10(3):271–275, 1990.
- [28] Keqin Li and Kam Hoi Cheng. Static job scheduling in partitionable mesh connected systems. *Journal of Parallel and Distributed Computing*, 10(2):152–159, 1990.
- [29] Silvano Martello and Paolo Toth. *Knapsack Problems: Algorithms and Computer Implementations*. Wiley, 1990.
- [30] A. Meir and L. Moser. On packing of squares and cubes. *Journal of Combinatorial Theory*, 5(2):126–134, 1968.