Approximation Algorithms for Two-Dimensional Geometric Packing Problems

Doctoral Dissertation submitted to the
Faculty of Informatics of the Università della Svizzera Italiana
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

presented by
Waldo Gálvez

under the supervision of
Prof. Fabrizio Grandoni

October 2019

Dissertation Committee

Prof. Antonio Carzaniga	Università della Svizzera Italiana, Switzerland
Prof. José Correa	Universidad de Chile, Chile
Prof. Alberto Marchetti-Spaccamela	Università "La Sapienza" Roma, Italy
Prof. Stefan Wolf	Università della Svizzera Italiana, Switzerland

Dissertation accepted on TBA October 2019

Research Advisor	PhD Program Director
Prof. Fabrizio Grandoni	**Prof. Walter Binder / Prof. Olaf Schenk**

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

Waldo Gálvez
Lugano, TBA October 2019

TBA

TBA

Abstract

There are a lot of natural problems arising in real life that can be modeled as discrete optimization problems. Unfortunately many of them are believed to be *hard to solve efficiently* (i.e. they cannot be solved in polynomial time unless P=NP). An *approximation algorithm* is one of the ways to tackle these hard optimization problems. These algorithms have polynomial running time and guarantee a feasible solution whose value is within a *proven factor* of the optimal solution value. The field of approximation algorithms has grown fast over the last few decades, and many techniques have been developed to handle these hard problems. However, there are still many problems for which substantial progress is needed. The ultimate goal for any optimization problem is an approximation algorithm with a performance guarantee along with a matching hardness of approximation result.

In this thesis we address two fundamental geometric packing problems: *Strip Packing* and *two-dimensional Geometric Knapsack*. In the Strip Packing problem we are given a set of rectangles and the goal is to place them into a rectangular region of fixed width W so that they do not overlap while minimizing the total height of the spanned region. On the other hand, in the two-dimensional Geometric Knapsack problem we are given a set of rectangles with associated profits and a square region of fixed height and width N, and the goal is to select and pack inside the region a subset of the rectangles of maximum profit so that they do not overlap. Both problems are NP-hard and have many interesting real-world applications, so they have been studied through the lens of approximation algorithms in the past.

We start by describing our results on the Strip Packing problem, where we develop improved approximation algorithms for some important special cases. In the first case we show a pseudo-polynomial time (PPT) $\left(\frac{4}{3}+\varepsilon\right)$-approximation which improves and simplifies the previous best $\left(\frac{7}{5}+\varepsilon\right)$-approximation from Nadiradze and Wiese [2016]. In the second case we show that there exists a tight $\left(\frac{3}{2}+\varepsilon\right)$-approximation for the problem in the special case where no rectangle is "large" in both dimensions (compared to the dimensions of the optimal solution). Both these results try to give new insights in order to approach the important

open problem of improving the approximability of Strip Packing.

In the second part we describe our results for the two-dimensional Geometric Knapsack problem and some of their known variants. For this problem we improve upon the best known approximation ratios for the cases with and without 90 degree rotations, and give refined approximation algorithms for the case of uniform weights. These are the first algorithms that break the approximation barrier of 2 for the aforementioned problems. As an important development we introduce the notion of L-packings which turns out to be crucial to achieve the previously mentioned results in the settings without rotations, and may be of independent interest to address related problems.

Acknowledgements

TBA

Contents

Contents **xi**

1 Introduction **1**
1.1 Basic definitions . . . 2
1.2 Two-dimensional Rectangle Packing Problems . . . 4
1.3 Outline of the Thesis . . . 6

2 Standard Tools **9**
2.1 Useful Procedures under Size/Area Guarantees . . . 9
2.1.1 Steinberg Algorithm . . . 9
2.1.2 Next-Fit Decreasing Height . . . 10
2.2 Container Packings . . . 11
2.2.1 Rounding the Container Dimensions . . . 12
2.2.2 Packing Rectangles into Containers . . . 14
2.3 Turning Fractional Packings into Feasible Solutions . . . 17
2.4 Including Small Rectangles into a Container Packing . . . 20
2.5 Packing Rectangles with Resource Augmentation . . . 22

3 Improved Approximation Algorithms for Strip Packing **23**
3.0.1 Prior work . . . 23
3.1 Strip Packing without Large Rectangles . . . 25
3.1.1 Preliminaries . . . 25
3.1.2 Hardness of Approximation . . . 26
3.1.3 Existence of a Structured Solution . . . 27
3.1.4 Algorithm . . . 33
3.2 A PPT $(4/3+\varepsilon)$-approximation for Strip Packing . . . 35
3.2.1 Preliminaries . . . 37
3.2.2 Overview of the Algorithm . . . 38
3.2.3 A Repacking Lemma . . . 42
3.2.4 A refined Structural Lemma . . . 45

3.2.5 The Final Algorithm 49
3.2.6 Extension to the Case with Rotations 50
3.3 Conclusions 50

4 On the Two-Dimensional Geometric Knapsack problem 53
4.0.1 Description of the Results 54
4.0.2 Other related work 57
4.1 Preliminaries 58
4.1.1 Item classification 58
4.1.2 Corridors, Spirals and Rings 59
4.2 Our approach: Search for L&C Packings 62
4.2.1 Main algorithm 63
4.3 A PTAS for L-packings 64
4.4 2DK without Rotations 72
4.4.1 A Simple Improved Approximation for Cardinality 2DK . . 73
4.4.2 Weighted Case without Rotations 75
4.5 2DK with Rotations 84
4.5.1 Cardinality 2DKR 84
4.5.2 Weighted 2DKR 90
4.6 Open Problems 95

A Two-Dimensional Geometric Knapsack with Resource Augmentation 97
A.1 Containers for Short-High Items 100
A.2 Fractional Packing with $O(1)$ Containers 101
A.2.1 Existence of an Integral Packing 102
A.3 Rounding Down Horizontal and Vertical Containers 103
A.4 Packing Short-Narrow Items 104

B Proof of Lemma 59 without Extra Assumptions 107
B.1 Definition of Grid and Corridor Partition 108
B.2 Partitioning Corridors into Boxes 112
B.3 Partitioning Boxes into Containers 113
B.4 Handling Small Items 113
B.5 L&C Packings 114

C Improved Approximation Algorithm for Cardinality 2DK without rotations 117
C.1 Overview of the Refined Packing 118
C.2 Design of Candidate Solutions 119
C.3 Bounding the Approximation Factor 132

Bibliography **135**

Chapter 1

Introduction

Many important optimization problems are known to be NP-hard (Garey and Johnson [1979]), and hence it is widely believed that an efficient algorithm (i.e. an algorithm with polynomial running time) to solve them exactly is unlikely to exist. Since these problems capture a lot of applications from the real world, it has become crucial to find ways to deal with them and get solutions in a reasonable time at the expense of efficiency in some sense. To this end, the major research sub-fields that have arisen are the following:

1. ***Exact and Parameterized Algorithms.*** The main goal in this field is to solve the problems exactly as fast as possible. This direction has successfully led to algorithms with super-polynomial running time but significantly faster than the naive brute-force approach, and also to sharpen lower bounds on the required running time to solve the problems under some complexity assumptions (usually stronger than P≠NP). More recently, a common approach is to isolate some aspects of the input as *parameter(s)* (for example the size of the solution, the diameter of the input graph, the maximum degree of the input graph, etc.) in order to obtain a running time guarantee which is super-polynomial only in the parameter(s). In practice, algorithms designed under this paradigm work efficiently on instances with small parameter-sized inputs.

2. ***Restricted Instances/Relaxed models.*** Another two successful approaches (similar in spirit to parameterization) is trying to solve NP-hard problems only for special classes of instances, such as planar graphs, perfect graphs or bounded cost/weight ratios (Hadlock [1975]; Grötschel et al. [2012]; Zukerman et al. [2001]), or to give extra power to the algorithm while comparing it to the optimum solution without those extra guarantees. In

the latter framework we can include resource augmentation, fractional solutions, etc. (Correa and Kenyon [2004]; Heydrich and Wiese [2017]; Lau et al. [2011]). These approaches come handy because, for many problems, algorithms for special cases as well as algorithms with extra power lead to improved algorithms for the general problem due to the deeper understanding of the inner structure of the instances.

3. ***Approximation Algorithms.*** The goal in this field is to develop efficient algorithms (i.e. with polynomial running time) that, although not necessarily outputting optimal solutions, compute solutions with a proven *performance guarantee* in the worst case. This field has provided powerful algorithmic techniques such as LP Rounding, Primal-Dual, Randomized Rounding and Metric Embeddings among others (Hochbaum [1997]; Vazirani [2001]; Williamson and Shmoys [2011]). In some cases, although the algorithms have polynomial running time, big exponents and hidden constants may do the algorithms impractical, but such bounds help to develop a further mathematical understanding of the inherent hardness of these problems. This acts as a stepping stone in the long run to come up with new algorithmic approaches.

Recently these approaches have been mixed to achieve better performance guarantees even for problems which are provably inapproximable. In particular, in the field of geometric optimization and packing problems, many results have been developed obtaining improved approximation algorithms with weaker running time guarantees (Adamaszek and Wiese [2014]; Jansen and Thöle [2010]; Wiese [2017]), improved approximations for restricted instances and relaxed models (Grandoni et al. [2015, 2017]; Wiese [2018]), and even approximation and exact algorithms with weaker running time guarantees for relaxed models (Antoniadis et al. [2017]; Pilipczuk et al. [2017]).

1.1 Basic definitions

In the following we present some basic definitions that will be used along this thesis. Let us start the formal definition of approximation algorithm.

Definition 1 (Williamson and Shmoys [2011]). *Given an optimization problem $\mathscr{P}$, an **α-approximation algorithm** for $\mathscr{P}$ is an algorithm with polynomial running time that, for each instance of the problem, outputs a solution whose value is within a factor α of the value of an optimal solution.*

The **approximation ratio** $\alpha \geq 1$ of an approximation algorithm is defined as

$$\max_{I \in \mathscr{I}} \max \left\{ \frac{OPT(I)}{APX(I)}, \frac{APX(I)}{OPT(I)} \right\},$$

where $\mathscr{I}$ is the set of instances of the problem, $APX(I)$ is the value of the solution computed by the approximation algorithm when run on instance I and $OPT(I)$ is the value of an optimal solution for instance I. The best kind of approximation algorithms one can hope to design are called *Polynomial Time Approximation Schemes* which we proceed to define.

Definition 2. *A **Polynomial Time Approximation Scheme** (PTAS) for an optimization problem $\mathscr{P}$ is a family of algorithms $\{\mathscr{A}_\varepsilon\}_{\varepsilon>0}$ such that, for each $\varepsilon > 0$, $\mathscr{A}_\varepsilon$ is a $(1+\varepsilon)$-approximation algorithm for $\mathscr{P}$. The running time of these algorithms is polynomial in the input size for any fixed ε.*

We can consider the following interesting relaxations of Definition 1: if we allow quasi-polynomial running time $O(n^{\text{polylog} n})$ instead of polynomial time, then similarly to Definition 2 a family of algorithms with arbitrarily good approximation ratios is called a **Quasi-Polynomial Time Approximation Scheme** (QPTAS). The development of this kind of algorithms is useful as it is commonly seen as a hint that an algorithm with polynomial running time achieving the same approximation guarantee exists. It is often as well insightful to consider the *asymptotic approximation ratios* as many pathological lower bound instances occur when the optimal value is small. We say that an algorithm is an asymptotic α-approximation if there exists some value $n_0 \in \mathbb{N}$ such that the algorithm is an α-approximation restricted to instances I where $OPT(I) \geq n_0$. A family of asymptotic approximation algorithms with arbitrarily good approximation ratios is called an **Asymptotic Polynomial Time Approximation Scheme** (APTAS)

We can also consider a *parameterized problem* which consists of a problem plus a parameter k that bounds a certain part of the input. One typically assumes that k is much smaller than the input size $|I|$. Choosing the correct parameter may allow us to find algorithms with a very accurate control on the dependence of the running time as a function of the parameter. This is the main notion that the field of Parameterized Complexity tries to capture.

Definition 3 (Cygan et al. [2015])**.** *A parameterized optimization problem $\mathscr{P}$ is called **Fixed-Parameter Tractable** (FPT) if there exists an algorithm $\mathscr{A}$, a computable function $f : \mathbb{N} \to \mathbb{N}$ and a constant c such that, given an instance (I, k), algorithm $\mathscr{A}$ correctly solves the problem in time bounded by $f(k) \cdot |I|^c$.*

In the special case of numerical problems (for example problems involving sizes or profits), there exists the notion of *pseudo-polynomial running time* which takes into account not only the length of the instance but also the values involved. Although being a weaker complexity notion it is a reasonable unit of measurement for some problems.

Definition 4 (Vazirani [2001]). *Given an optimization problem $\mathscr{P}$, an algorithm for $\mathscr{P}$ is said to have* ***pseudo-polynomial running time*** *(PPT) if, for any instance I of $\mathscr{P}$, the running time of the algorithm applied to I is poly($|I_u|$), where I_u corresponds to instance I encoded in unary.*

Equivalently, an algorithm runs in PPT if its running time is bounded by $O(\text{poly}(n \cdot W))$ where n is the size of the instance and W is the absolute value of the largest integer in the instance. If an NP-hard problem can be solved in PPT we say it is **weakly NP-hard**, and if it does not admit a PPT algorithm unless P=NP we say it is **strongly NP-hard**.

Additionally, we can search for *hardness of approximation* results, which means that there cannot exist an approximation algorithm with performance guarantee better than a given threshold assuming P≠NP or other widely believed conjecture relating known complexity classes. A problem is said to be APX-hard if it does not admit a PTAS unless P=NP. The ultimate goal for any NP-hard optimization problem is to find an approximation algorithm for it plus a matching hardness of approximation result.

1.2 Two-dimensional Rectangle Packing Problems

An important family of problems in the field that has received increasing attention in the last years is the family of *Rectangle Packing* problems. In general, instances of these problems consist of a family of rectangles, each one characterized by an integral height and an integral width (and possibly an integral profit), plus a region in the two-dimensional plane, and the objective is to pack the rectangles (or a subset of them) inside the given region in an axis-parallel way such that they do not overlap while optimizing some given criteria. Many important and well-studied problems fall in this framework, such as the two-dimensional geometric variants of Bin Packing, Independent Set and many more. In this thesis we study two such problems, namely the Strip Packing problem and the two-dimensional Geometric Knapsack problem.

In the Strip Packing problem we are given a collection of rectangles, and an infinite vertical strip of width W in the two-dimensional plane. As mentioned before, we need to find an axis-parallel embedding of the rectangles *without rotations* inside the strip so that no two rectangles overlap (feasible *packing*). Our goal is to minimize the final height of this packing. More formally, we are given a parameter $W \in \mathbb{N}$ and a set $\mathcal{R} = \{R_1, \ldots, R_n\}$ of rectangles, each one characterized by a width $w(R_i) \in \mathbb{N}$, $w(R_i) \leq W$, and a height $h(R_i) \in \mathbb{N}$. A packing of $\mathcal{R}$ is a pair $(left(R_i), bottom(R_i)) \in \mathbb{N} \times \mathbb{N}$ for each R_i, with $0 \leq left(R_i) \leq W - w(R_i)$, meaning that the left-bottom corner of R_i is placed in position $(left(R_i), bottom(R_i))$ and its right-top corner in position $(right(R_i), top(R_i))$, where $right(R_i) = left(R_i) + w(R_i)$ and $top(R_i) = bottom(R_i) + h(R_i)$. This packing is feasible if the interiors of the rectangles are pairwise disjoint in this embedding (or equivalently rectangles are allowed to overlap on their boundary only). Our goal is to find a feasible packing of *minimum final height* $\max_i\{top(R_i)\}$ (see Figure 1.1 for an example). Strip packing is a natural generalization of *one-dimensional Bin Packing* (Coffman Jr. et al. [2013]), which is the case when all the rectangles have the same height, and *makespan minimization* (Coffman Jr. and Bruno [1976]) which is the case when all the rectangles have the same width. The problem has lots of applications in industrial engineering and computer science, specially in cutting stock, logistics and scheduling (Kenyon and Rémila [2000]; Harren et al. [2014]). More recently, there have been several applications of Strip Packing in electricity allocation and peak demand reduction in smart-grids (Tang et al. [2013]; Karbasioun et al. [2013]; Ranjan et al. [2015]).

The two-dimensional Geometric Knapsack problem is the geometric variant of the classical (one-dimensional) knapsack problem. We are given a set of n rectangles $\mathcal{R} = \{R_1, \ldots, R_n\}$, where each rectangle $R_i \in \mathcal{R}$ has integral height $h(R_i)$, integral width $w(R_i)$ and an associated integral profit $p(R_i)$. Furthermore, we are given an axis-aligned square knapsack $K = [0, N] \times [0, N]$, $N \in \mathbb{N}$, and our goal is to select a subset of rectangles $OPT \subseteq \mathcal{R}$ of maximum total profit $p(OPT) := \sum_{R_i \in OPT} p(R_i)$ and to find a feasible packing for the selected rectangles fully contained in the knapsack. Different variants of the problem have been considered in the literature, being the most important ones the cases with/without rotations (meaning that 90 degree rotations for the rectangles may be allowed) and whether the rectangles have different weights (weighted case) or all of them have weight 1 (cardinality case). See Figure 1.2 for an example. Besides being a natural mathematical problem, it is well-motivated by practical applications. For instance, one might want to place advertisements on a board or a website, or cut rectangular pieces from a sheet of some material of fixed size. Also, it models a scheduling setting where each rectangle corresponds to a job that needs some "consecutive amount" of a given resource (memory storage, frequencies, etc.). In all these cases, dealing with rectangular shapes only is a reasonable simpli-

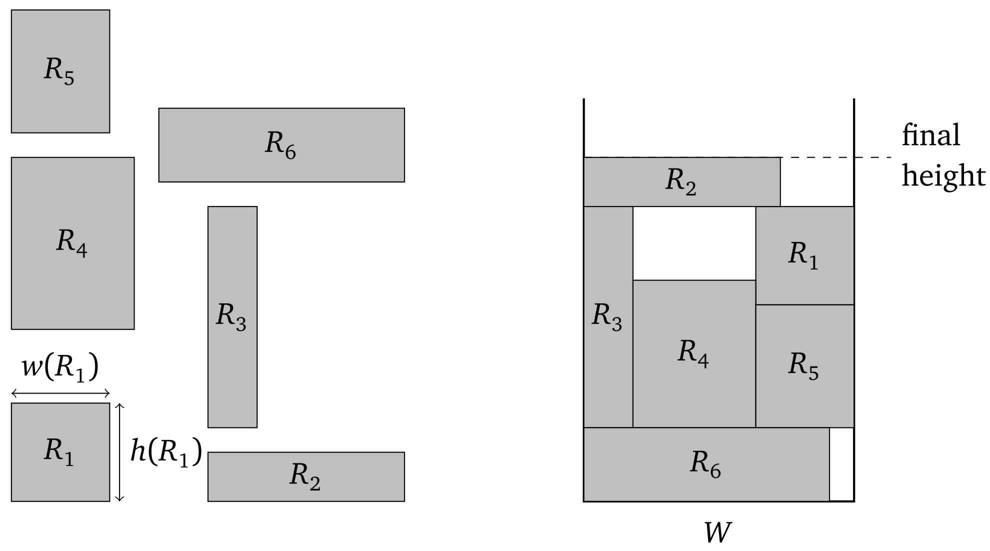


Figure 1.1. An instance of Strip Packing. Each rectangle R_i is characterized by its height $h(R_i)$ and width $w(R_i)$ (Left), and the goal is to pack them into a strip of width W so that the final height is minimized (Right).

fication and often the developed techniques can be extended to deal with more general instances.

Strip Packing and two-dimensional Geometric Knapsack are known to be strongly NP-hard even when the instance consists solely of unweighted squares (Leung et al. [1990]). Accordingly, they have received significant attention from the point of view of approximation algorithms. Along this work we make further progress in the understanding of the approximability of these problems by developing improved classical approximation algorithms and also improved algorithms for some of the described relaxed models.

1.3 Outline of the Thesis

In Chapter 2 we review some preliminary results and useful tools to handle rectangle packing problems. We also introduce the notion of Container-based solutions which will be central for our approach.

In Chapter 3 we describe our results on Strip Packing, namely improving the known upper bounds for the approximability of the problem in pseudo-polynomial time and also a tight approximation algorithm for a family of relevant restricted instances. In the end we discuss some open questions and further directions.

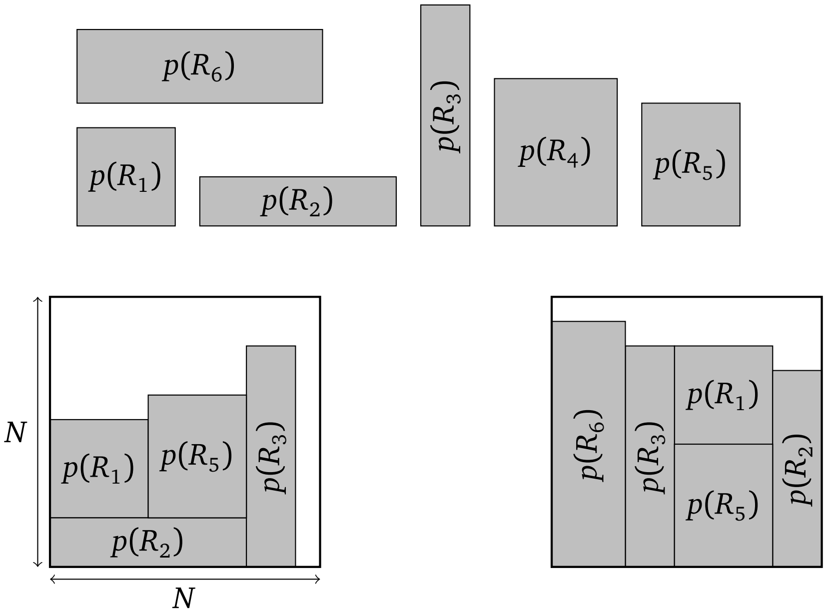


Figure 1.2. An instance of Two-Dimensional Geometric Knapsack. Each rectangle R_i is characterized by its height $h(R_i)$, width $w(R_i)$ and profit $p(R_i)$ (Top), and the goal is to pack a subset of them of maximum profit into a given squared Knapsack either without rotations (Left) or with 90 degree rotations (Right).

Then in Chapter 4 we describe our results on two-dimensional Geometric Knapsack, improving the approximability on all the variants mentioned in the previous section. Again, in the end of the chapter we discuss some open problems and future directions of research.

We defer to the Appendix some rather technical proofs and complementary results.

Chapter 2

Standard Tools

In this chapter we review some useful tools and preliminary results that will be used along this work. Most of these results are implicitly used in previous literature, so this is an attempt to standardize the whole approach and techniques in a comprehensive way. Given a set of rectangles $\mathscr{R}$ we will denote by $h_{\max}(\mathscr{R})$ (resp. $w_{\max}(\mathscr{R})$) the maximum height (resp. width) among the rectangles in $\mathscr{R}$, and by $a(\mathscr{R})$ the total area of the rectangles in $\mathscr{R}$, i.e., $\sum_{R_i \in \mathscr{R}} h(R_i)w(R_i)$.

We call a *box* a rectangular region in the plane where rectangles can be packed inside, and we say it has size $a \times b$ if it has width a and height b.

2.1 Useful Procedures under Size/Area Guarantees

2.1.1 Steinberg Algorithm

The following theorem due to Steinberg gives a useful tool to pack items into rectangular regions provided roughly only of area constraints and it comes handy when proving the existence of a feasible packing into a given region. For a given number x, let us denote $x_+ := \max(x, 0)$.

Theorem 5 (Steinberg [1997]). *Suppose we are given a set of rectangles $\mathscr{R}'$ and a rectangular region Q of width w and height h. If*

$$2a(\mathscr{R}') \leq wh - (2w_{\max} - w)_+(2h_{\max} - h)_+$$

then $\mathscr{R}'$ can be packed into Q. Furthermore, there is a polynomial time algorithm that computes such packing.

Sometimes we will make use of the following simpler corollary which is a direct application of Steinberg theorem.

Corollary 6. *Consider a set $\mathcal{R}'$ of rectangles and a rectangular region Q of width w and height h. If $w_{\max}(\mathcal{R}') \leq \frac{w}{2}$ (resp. $h_{\max}(\mathcal{R}') \leq \frac{h}{2}$) and $a(\mathcal{R}') \leq \frac{wh}{2}$, then there exists a feasible packing of $\mathcal{R}'$ inside Q.*

2.1.2 Next-Fit Decreasing Height

One of the most common algorithms to pack rectangles into a box of size $w \times h$ is Next-Fit Decreasing Height (NFDH). In this algorithm, the first step is to sort rectangles non-increasingly by height, say $h(R_1) \geq h(R_2) \geq \ldots \geq h(R_n)$. Then, the first rectangle is packed in the bottom-left corner, and a shelf is defined of height $h(R_1)$ and width w. The next rectangles are put in this shelf, next to each other and touching the bottom of the shelf, until one does not fit, say the i-th one. At this point we define a new shelf above the first one, with height $h(R_i)$. This process continues until all the rectangles are packed or the height of the next shelf does not fit inside the box (see Figure 2.1b for an example).

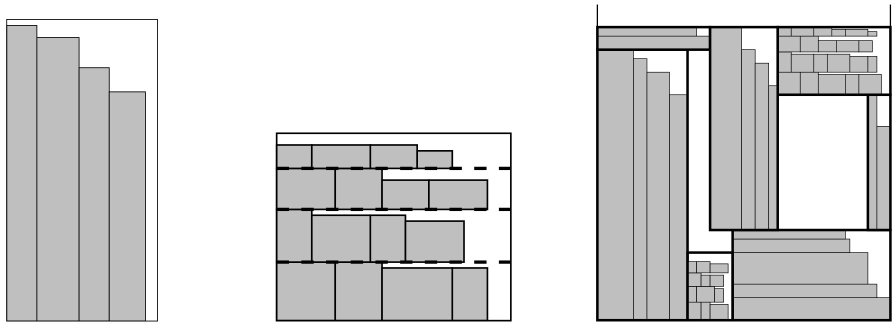
This algorithm was studied by Coffman Jr. et al. [1980] in the context of Strip Packing, in order to bound the obtained height when all the rectangles are packed into a strip. An important property of the algorithm is that, if a given set of rectangles needs to be packed into a given rectangular region and all of them are relatively small compared to the dimensions of the region, then NFDH is very efficient even in terms of area. This result can be summarized in the following lemma which will be useful for our purposes.

Lemma 7 (Coffman Jr. et al. [1980]). *Given a set of rectangles with width at most w and height at most h, if NFDH is used to pack these rectangles into a rectangular region of width a and height b, then all the rectangles are packed or the area of the rectangles that were packed is at least $(a-w)(b-h)$.*

We will also often make use of the following simpler corollary.

Corollary 8. *Consider a rectangular region of width w and height h, and a set of rectangles $\mathcal{R}'$ satisfying that $w_{\max}(\mathcal{R}') \leq \varepsilon w$, $h_{\max}(\mathcal{R}') \leq \varepsilon h$ and $a(\mathcal{R}') \leq (1-2\varepsilon)wh$, then NFDH packs all the rectangles from $\mathcal{R}'$ into the region.*

Proof. If NFDH does not pack all the rectangles into the region, then due to Lemma 7 the total area of the packed rectangles is at least $(1-\varepsilon)^2 wh \leq (1-2\varepsilon)wh$ which is a contradiction. □



(a) Example of vertical container. Every vertical line intersects at most one rectangle.

(b) Example of area container. Shelves are defined by the set of dashed lines.

(c) A container packing, consisting of a constant number of vertical, horizontal and area containers.

Figure 2.1. Illustration of Container-based solutions and related definitions.

2.2 Container Packings

A big part of previous work on Geometric Packing problems (e.g. Bansal et al. [2009]; Nadiradze and Wiese [2016]) implicitly or explicitly exploits a *container-based* approach. The idea is to restrict ourselves to solutions that are partitioned into a constant number of axis-aligned rectangular regions (*containers*) such that their sizes (and therefore positions) can be efficiently computed, and subsequently place the rectangles inside the containers in a simple way: either one next to the other from left to right or from bottom to top, or by means of NFDH. Container-like packings turn out to be particularly useful since they naturally induce (one-dimensional) knapsack instances as it is described in Section 2.2.2.

More in detail, a *container* is a box labeled as *horizontal*, *vertical*, or *area* (see Figure 2.1 for a reference). A *container packing* for a set of rectangles $\mathscr{R}'$ into a collection of non-overlapping containers has to satisfy the following properties:

- Items in a horizontal (resp., vertical) container are stacked one on top of the other (resp., one next to the other).

- Each $R \in \mathscr{R}'$ packed in an area container of size $a \times b$ must satisfy $w(R) \leq \varepsilon a$ and $h(R) \leq \varepsilon b$.

In what follows we will show how to reduce the possible sizes of the containers in any container packing to a set of polynomial size (that can then be computed efficiently), and then how to pack the rectangles into the containers once we have already chosen them. Depending on the setting, we may allow to discard some rectangles or to increase the running time in order to achieve this. With this framework in mind, the main remaining question would be: how good is the best container packing with respect to the optimal solution? Answering this question directly implies an approximation guarantee for the induced search algorithm and is the main focus of Chapters 3 and 4 for the cases of Strip Packing and two-dimensional Geometric Knapsack respectively.

2.2.1 Rounding the Container Dimensions

In this section we show that it is possible to round down the size of each horizontal, vertical or area container so that the resulting sizes can be chosen from a polynomially sized set, while removing a set of rectangles of small total profit.

Lemma 9. *Let $\mathcal{R}$ be a set of weighted rectangles that can be packed into $O_\varepsilon(1)$ containers. Then there exists a set $\mathcal{R}'$ of total profit $p(\mathcal{R}') \geq (1 - O(\varepsilon))p(\mathcal{R})$ such that there is a container packing for $\mathcal{R}'$ inside the original containers and the possible sizes of the new containers belong to a set of size poly(n).*

In order to prove Lemma 9, we will give first some definitions. For a set $\mathcal{R}$ of rectangles, we define $\mathit{WIDTHS}(\mathcal{R}) = \{w(R) \,|\, R \in \mathcal{R}\}$ and $\mathit{HEIGHTS}(\mathcal{R}) = \{h(R) \,|\, R \in \mathcal{R}\}$.

Given a finite set P of real numbers and a fixed natural number k, we define the set

$$P^{(k)} = \{(p_1 + p_2 + \cdots + p_l) + ip_{l+1} \,|\, p_j \in P \ \forall j, l \leq k, 0 \leq i \leq n, i \text{ integer}\}.$$

Note that if $|P| = O(n)$, then $|P^{(k)}| = O(n^{k+2})$. Moreover, if $P \subseteq Q$, then obviously $P^{(k)} \subseteq Q^{(k)}$, and if $k' \leq k''$, then $P^{(k')} \subseteq P^{(k'')}$. Finally, given two containers C_1 and C_2, we say that C_1 is smaller than C_2 if C_1 fits inside C_2.

Lemma 10. *Let $\varepsilon > 0$, and let $\mathcal{R}$ be a set of weighted rectangles packed in an horizontal (resp. vertical) container C. Then, for any $k \geq 1/\varepsilon$, there is a set $\mathcal{R}' \subseteq \mathcal{R}$ with total profit $p(\mathcal{R}') \geq (1 - \varepsilon)p(\mathcal{R})$ that can be packed in a container C' smaller than C such that $h(C') \in \mathit{HEIGHTS}(\mathcal{R})^{(k)}$ and $w(C') \in \mathit{WIDTHS}(\mathcal{R}^{(k)})$.*

Proof. Without loss of generality, we prove the claim for an horizontal container C; the proof for vertical containers is symmetric. Clearly, the width of C can be reduced to $w_{\max}(\mathcal{R})$, and $w_{\max}(\mathcal{R}) \in \mathit{WIDTHS}(\mathcal{R}) \subseteq \mathit{WIDTHS}(\mathcal{R})^{(k)}$.

If $|\mathcal{R}| \leq 1/\varepsilon$, then $\sum_{R_i \in \mathcal{R}} h(R_i) \in \mathit{HEIGHTS}(\mathcal{R})^{(k)}$ and we can fix the height of C to be that value. Otherwise, let $\mathcal{R}_{TALL}$ be the set of the $1/\varepsilon$ rectangles in $\mathcal{R}$ with largest height (breaking ties arbitrarily), let R_j be the one with smallest profit among them, and let $\mathcal{R}' = \mathcal{R} \setminus \{R_j\}$. Clearly, $p(\mathcal{R}') \geq (1-\varepsilon)p(\mathcal{R})$. Since each element of $\mathcal{R}' \setminus \mathcal{R}_{TALL}$ has height at most $h(R_j)$, it follows that $h(\mathcal{R} \setminus \mathcal{R}_{TALL}) \leq (n-1/\varepsilon)h(R_j)$. Thus, letting $i = \lceil h(\mathcal{R}' \setminus \mathcal{R}_{TALL})/h(R_j) \rceil \leq n$, all the rectangles in $\mathcal{R}'$ fit in a container C' of width $w_{\max}(\mathcal{R})$ and height $h(C') := h(\mathcal{R}_{TALL}) + ih(R_j) \in HEIGHTS(\mathcal{R})^{(k)}$. Since $h(\mathcal{R}_{TALL}) + ih(R_j) \leq h(\mathcal{R}_{TALL}) + h(\mathcal{R}' \setminus \mathcal{R}_{TALL}) + h(R_j) = h(\mathcal{R}) \leq h(C)$, this proves the result. □

Lemma 11. *Let $\varepsilon > 0$, and let $\mathcal{R}$ be a set of rectangles packed inside an area container C. Then there exists a subset $\mathcal{R}' \subseteq \mathcal{R}$ with total profit $p(\mathcal{R}') \geq (1-4\varepsilon)p(\mathcal{R})$ and a container C' smaller than C such that $w(C') \in WIDTHS(\mathcal{R})^{(0)}$, $h(C') \in HEIGHTS(\mathcal{R})^{(0)}$, and each $R_j \in \mathcal{R}'$ satisfies that $w(R_j) \leq \frac{\varepsilon}{1-\varepsilon}w(C')$ and $h(R_j) \leq \frac{\varepsilon}{1-\varepsilon}h(C')$.*

Proof. Without loss of generality, we can assume that $w(C) \leq nw_{\max}(\mathcal{R})$ and $h(C) \leq nh_{\max}(\mathcal{R})$: if not, we can first shrink C so that these conditions are satisfied, and all the rectangles still fit in C.

Define a container C' of width $w(C') = w_{\max}(\mathcal{R})\lfloor w(C)/w_{\max}(\mathcal{R})\rfloor$ and height $h(C') = h_{\max}(\mathcal{R})\lfloor h(C)/h_{\max}(\mathcal{R})\rfloor$, that is, C' is obtained by shrinking C to the closest integer multiples of $w_{\max}(\mathcal{R})$ and $h_{\max}(\mathcal{R})$. Observe that $w(C') \in WIDTHS(\mathcal{R})^{(0)}$ and $h(C') \in HEIGHTS(\mathcal{R})^{(0)}$. Clearly, $w(C') \geq w(C) - w_{\max}(\mathcal{R}) \geq w(C) - \varepsilon w(C) = (1-\varepsilon)w(C)$, and similarly $h(C') \geq (1-\varepsilon)h(C')$. Hence $a(C') \geq (1-\varepsilon)^2 a(C) \geq (1-2\varepsilon)a(C)$.

We now select a set $\mathcal{R}' \subseteq \mathcal{R}$ by greedily choosing elements from $\mathcal{R}$ sorted non-increasingly by profit/area ratio, adding as many elements as possible without exceeding a total area of $(1-2\varepsilon)a(C')$ (which then would be packable inside C' thanks to Corollary 8). Let R_{last} be the last rectangle added to $\mathcal{R}'$. Since each element of $\mathcal{R}$ has area at most $\varepsilon^2 a(C)$, then either all elements are selected (and then $p(\mathcal{R}') = p(\mathcal{R})$), or the total area of the selected elements is at least $(1-2\varepsilon)a(C') - \varepsilon^2 a(C) \geq (1-3\varepsilon)a(\mathcal{R})$. In this case, since $\frac{p(R_{last})}{a(R_{last})} \leq \frac{p(\mathcal{R})}{a(\mathcal{R}')}$ (as otherwise the selected rectangles would have profit $\sum_{R \in \mathcal{R}'} \frac{p(R)}{a(R)} a(R) > \frac{p(\mathcal{R})}{a(\mathcal{R}')} \sum a(R) = p(\mathcal{R})$), the total profit of the rectangles that were not packed is at most

$$\sum_{R \in \mathcal{R} \setminus \mathcal{R}'} \frac{p(R)}{a(R)} a(R) \leq \frac{p(R_{last})}{a(R_{last})} \sum_{R \in \mathcal{R} \setminus \mathcal{R}'} a(R) \leq \frac{p(\mathcal{R})}{a(\mathcal{R}')} 3\varepsilon a(\mathcal{R}) \leq 4\varepsilon p(\mathcal{R}).$$

Since the container was shrunk by a factor $(1-\varepsilon)$ at most, the claim is proved. □

By applying Lemmas 10 and 11 we conclude the proof of Lemma 9

2.2.2 Packing Rectangles into Containers

Now we proceed to show that, given a set of a constant number of containers and a set of weighted rectangles, it is possible to efficiently find an almost optimal packing of the rectangles into the containers. In order to achieve that we will first show that there is a PTAS for the closely related Maximum Generalized Assignment Problem (GAP) if the number of bins is constant. In GAP, we are given a set of k one-dimensional bins with capacity constraints and a set of n items that have a possibly different size and profit for each bin, and the goal is to pack a maximum-profit subset of items into the bins. Let us assume that if item i is packed in bin j, then it has size $s_{ij} \in \mathbb{Z}$ and profit $p_{ij} \in \mathbb{Z}$.

GAP is known to be APX-hard and the best known polynomial time approximation algorithm has ratio $(1-1/e+\varepsilon)$ (Fleischer et al. [2011]; Feige and Vondrak [2006]). In fact, for an arbitrarily small constant $\delta > 0$ (which can even be a function of n) GAP remains APX-hard even on the following instances: bin capacities are identical, and for each item i and bin j, $p_{ij} = 1$, and $s_{ij} = 1$ or $s_{ij} = 1+\delta$ (Chekuri and Khanna [2005]). The complementary case, where item sizes do not vary across bins but profits do, is also APX-hard (Chekuri and Khanna [2005]). However, when all profits and sizes are the same across all bins (i.e., $p_{ij} = p_{ik}$ and $s_{ij} = s_{ik}$ for all bins j, k), the problem is known as multiple knapsack problem (MKP) and it admits a PTAS (Chekuri and Khanna [2005]).

On the other hand, for our purposes we only need instances where $k = O(1)$. A PTAS for GAP for a constant number of bins follows from extending known techniques from the literature (Shmoys and Tardos [1993]). However, we did not find an explicit proof in the literature and thus, for the sake of completeness, in this section we present a full, self-contained description of such an algorithm.

Let C_j be the capacity of bin j. Let $p(OPT)$ be the profit of the optimal assignment.

Theorem 12. *There exists a pseudo-polynomial time algorithm for GAP with $k = O(1)$ bins.*

Proof. For each $i \in [n]$ and $c_j \in [C_j]$ for each $j \in [k]$, let $S_{i,c_1,c_2,\ldots,c_k}$ denote a subset of the set of items $\{1,2,\ldots,i\}$ packed into the bins such that the profit is maximized and capacity of bin j is at most c_j. Let $P[i,c_1,c_2,\ldots,c_k]$ denote the profit of $S_{i,c_1,c_2,\ldots,c_k}$. Clearly $P[1,c_1,c_2,\ldots,c_k]$ is known for all $c_j \in [C_j]$ for $j \in [k]$. Moreover, we define $P[i,c_1,c_2,\ldots,c_k] = 0$ if $c_j < 0$ for any $j \in [k]$. We can compute the value of $P[i,c_1,c_2,\ldots,c_k]$ by using a dynamic program (DP), that exploits the following recurrence:

$$P[i,c_1,c_2,\dots,c_k] = \max\{P[i-1,c_1,c_2,\dots,c_k], \max_j\{p_{ij} + P[i-1,c_1,\dots,c_j - s_{ij},\dots,c_k]\}\}$$

With a similar recurrence, we can easily compute a corresponding set $S_{i,c_1,c_2,\dots,c_k}$. The running time of the above program is $O\Big(n\prod_{j=1}^{k} C_j\Big)$ which is pseudo-polynomial.

□

Theorem 13. *There is an algorithm for maximum generalized assignment problem with k bins that runs in time* $O\Big(\big(\frac{1+\varepsilon}{\varepsilon}\big)^k n^{k/\varepsilon^2+k+1}\Big)$ *and returns a solution that has profit at least* $(1-3\varepsilon)p(OPT)$*, for any fixed* $\varepsilon > 0$.

In order to prove Theorem 13 we will first prove that it is possible to achieve such a result by slightly augmenting the sizes of the bins as the following lemma states.

Lemma 14. *There is a* $O\Big(\big(\frac{1+\varepsilon}{\varepsilon}\big)^k n^{k+1}\Big)$ *time algorithm for the maximum generalized assignment problem with k bins, which returns a solution with profit at least* $p(OPT)$ *if we are allowed to augment the bin capacities by a* $(1+\varepsilon)$*-factor for any fixed* $\varepsilon > 0$.

Proof. Consider the dynamic program described in Theorem 12 and its running time $O\Big(n\prod_{j=1}^{k} C_j\Big)$. If each C_j is polynomially bounded, then this running time is polynomial. Therefore, we will now create a modified instance where each bin size is polynomially bounded.

Let $\mu_j = \frac{\varepsilon C_j}{n}$. For item i and bin j, define the modified size $s'_{ij} = \Big\lceil \frac{s_{ij}}{\mu_j} \Big\rceil = \Big\lceil \frac{ns_{ij}}{\varepsilon C_j} \Big\rceil$ and $C'_j = \Big\lfloor \frac{(1+\varepsilon)C_j}{\mu_j} \Big\rfloor$. Note that $C'_j = \Big\lfloor \frac{(1+\varepsilon)n}{\varepsilon} \Big\rfloor \le \frac{(1+\varepsilon)n}{\varepsilon}$, so the aforementioned DP requires time at most $O\Big(n \cdot \big(\frac{(1+\varepsilon)n}{\varepsilon}\big)^k\Big)$

The DP finds the optimal solution $OPT_{modified}$ for the modified instance. Now consider the optimal solution for the original instance (i.e., with original item sizes and bin sizes) $OPT_{original}$. If we show the same assignment of items to the bins is a feasible solution (with modified bin sizes and item sizes) for the modified instance, we get $OPT_{modified} \ge OPT_{original}$ and that will conclude the proof.

Let S_j be the set of items packed in bin j in the $OPT_{original}$. So, $\sum_{i\in S_j} s_{ij} \le C_j$. Hence,

$$\sum_{i\in S_j} s'_{ij} \le \left\lfloor \sum_{i\in S_j}\left(\frac{s_{ij}}{\mu_j}+1\right)\right\rfloor \le \left\lfloor \frac{1}{\mu_j}\left(\sum_{i\in S_j} s_{ij} + |S_j|\mu_j\right)\right\rfloor \le \left\lfloor \frac{1}{\mu_j}(C_j + n\mu_j)\right\rfloor \le \left\lfloor \frac{(1+\varepsilon)C_j}{\mu_j}\right\rfloor = C'_j$$

Thus $OPT_{original}$ is a feasible solution for the modified instance and the DP will return a packing with profit at least $p(OPT)$ under resource augmentation. □

Now we can show how to employ this result to obtain a feasible solution with an almost optimal profit using the original bin capacities.

Proof of Theorem 13. First, we claim the following:

Claim 15. *If a set of items $\mathscr{R}_j$ is packed in a bin B_j with capacity C_j, then there exists a set of at most $O(1/\varepsilon^2)$ items X_j, and a set of items Y_j with $p(Y_j) \leq \varepsilon p(\mathscr{R}_j)$ such that all items in $\mathscr{R}_j \setminus (X_j \cup Y_j)$ have size at most $\varepsilon(C_j - \sum_{i \in X_j} s_{ij})$.*

Proof. Let Q_1 be the set of items i with $s_{ij} > \varepsilon C_j$. If $p(Q_1) \leq \varepsilon p(\mathscr{R}_j)$, we are done by taking $Y_j = Q_1$ and $X_j = \phi$. Otherwise, define $X_j := Q_1$ and we continue the next iteration with the remaining items. Let Q_2 be the items with size greater than $\varepsilon(C_j - \sum_{i \in X_j} s_{ij})$ in $\mathscr{R}_j \setminus X_j$. If $p(Q_2) \leq \varepsilon p(\mathscr{R}_j)$, we are done by taking $Y_j = Q_2$. Otherwise define $X_j := Q_1 \cup Q_2$ and we continue with further iterations till we get a set Q_t with $p(Q_t) \leq \varepsilon p(\mathscr{R}_j)$. Note that we need at most $\frac{1}{\varepsilon}$ iterations since the sets Q_i are disjoint. Otherwise, $p(\mathscr{R}_j) \geq \sum_{i=1}^{1/\varepsilon} p(Q_i) > \sum_{i=1}^{1/\varepsilon} \varepsilon p(\mathscr{R}_j) \geq p(\mathscr{R}_j)$, which is a contradiction. Thus, consider $Y_j = Q_t$ and $X_j = \bigcup_{l=1}^{t-1} Q_l$. One has $|X_j| \leq 1/\varepsilon^2$ and $p(Y_j) \leq \varepsilon p(\mathscr{R}_j)$. On the other hand, after removing Q_t, the remaining items have size $< \varepsilon(C_j - \sum_{i \in X_j} s_{ij})$. □

Now consider a bin with capacity $(C_j - \sum_{i \in X_j} s_{ij})$ where all packed items $\mathscr{R}'_j$ have sizes smaller than $\varepsilon(C_j - \sum_{i \in X_j} s_{ij})$, then we can divide the bin into $1/\varepsilon$ equal sized intervals $S_{j,1}, S_{j,2}, \ldots, S_{j,1/\varepsilon}$ of length $\varepsilon(C_j - \sum_{i \in X_j} s_{ij})$. Let $\mathscr{R}'_{j,l}$ be the set of items intersecting the interval $S_{j,l}$. As each packed item can belong to at most two such intervals, the cheapest set $\mathscr{R}''$ among $\{\mathscr{R}'_{j,1}, \ldots, \mathscr{R}'_{j,1/\varepsilon}\}$ has profit at most $2\varepsilon p(\mathscr{R}'_j)$. Thus we can remove this set $\mathscr{R}''$ and reduce the bin size by a factor of $(1-\varepsilon)$.

Now consider the packing of k bins B_j in the optimal packing OPT. Let $\mathscr{R}_j$ be the set of items packed in bin B_j. Now the algorithm first guesses all X_j's, a constant number of items, in all k bins. We assign them to corresponding bins in $O(n^{k/\varepsilon^2})$ time. Then for bin j we are left with capacity $r_j := C_j - \sum_{i \in X} s_{ij}$. From previous discussion, we know that there is packing of $\mathscr{R}''_j \subseteq \mathscr{R}_j \setminus X_j$ of profit $(1-2\varepsilon)p(\mathscr{R}_j \setminus X_j)$ in a bin with capacity $(1-\varepsilon)C_j$. Thus we can use the resource augmentation algorithm for GAP from Lemma 14 to pack the remaining items in k bins where for bin j we set its capacity to be $(1-\varepsilon)C_j$ for $j \in [k]$ before the resource augmentation. As Lemma 14 returns the optimal packing on this modified bin sizes we get total profit at least $(1-3\varepsilon)p(OPT)$. □

Provided of these tools we can now show how to pack rectangles into a given set of containers. The basic idea is to reduce the problem to an instance of GAP with one bin per container, and then solve the latter problem directly placing the assigned rectangles into horizontal and vertical containers plus using Next Fit Decreasing Height to pack rectangles in area containers.

Theorem 16. *There is a PTAS for the problem of computing a maximum profit packing of a subset of rectangles of a given set $\mathscr{R}'$ into a given set of containers of constant cardinality. On the other hand, it is possible to solve exactly this same problem in pseudo polynomial time.*

Proof. For each horizontal container C_j of size $w(C_j) \times h(C_j)$, we create a (one-dimensional) bin j of size $h(C_j)$. Furthermore, we define the size $b(i,j)$ of rectangle R_i w.r.t. knapsack j as $h(R_i)$ if $h(R_i) \leq h(C_j)$ and $w(R_i) \leq w(C_j)$. Otherwise $b(i,j) = +\infty$ (meaning that R_i does not fit in C_j). The construction for vertical containers is symmetric. For each area container C_k we create a bin k of size $(1-2\varepsilon)a(C_k)$ and define the size $b(i,k)$ of rectangle R_i w.r.t. knapsack k as $h(R_i)w(R_i)$ if $h(R_i) \leq \varepsilon h(C_k)$ and $w(R_i) \leq \varepsilon w(C_k)$, setting $b(i,k) = +\infty$ otherwise (meaning that the rectangle is not small with respect to the dimensions of the container). This way, thanks to Corollary 8, all the rectangles assigned to an area container can be packed using NFDH. The profit of rectangle R_i in every bin will be $p(R_i)$.

Thanks to Theorem 13 there is a PTAS for this problem and moreover it can be solved exactly in pseudo-polynomial time thanks to Theorem 12. □

2.3 Turning Fractional Packings into Feasible Solutions

A very common and useful tool is to develop solutions assuming that an instance can be *sliced*. We say that a set of rectangles $\mathscr{R}$ is **horizontally sliced** if for each rectangle $R_i \in \mathscr{R}$ is replaced by $h(R_i)$ rectangles of height 1 and width $w(R_i)$. As the following lemma states, if the rectangles have small enough height and it is possible to find a well structured packing for a sliced instance, then it is possible to turn this solution into a container packing without slicing while dropping a set of rectangles that can be efficiently repacked. An analogous statement can be achieved for vertical slicing just by rotating the instance 90 degrees first. This result is a slight generalization of the techniques presented by Kenyon and Rémila

[2000] in the context of Strip Packing and the proof is very similar in spirit to theirs.

Lemma 17. *Let $\varepsilon > 0$ and consider a rectangular region of width A and height B. Let also $\mathcal{R}$ be a set of rectangles having height at most δB for some given $\delta > 0$. If there exists a packing for $\mathcal{R}_{sliced}$ (horizontally sliced) inside the region which can be decomposed into $K \in O_\varepsilon(1)$ rectangular boxes and $\delta \leq \left(\frac{\varepsilon^8}{K}\right)^{2+K/\varepsilon^7}$, then it is possible to partition $\mathcal{R}$ into two sets $\mathcal{R}_{cont}$ and $\mathcal{R}_{disc}$ so that:*

- *There exists a container packing of $\mathcal{R}_{cont}$ such that the containers lie inside the original boxes, and*
- *$\mathcal{R}_{disc}$ can be packed into an horizontal container of height at most εB and width at most $\max_{R_i \in \mathcal{R}_{disc}} w(R_i)$ plus an area container of height at most $2\varepsilon B$ and width at most $2\varepsilon A$.*

Proof. We will first partition the rectangles into two sets, $\mathcal{R}_N$ the ones having width at most $\frac{\varepsilon^6}{K}A$ being *narrow* and $\mathcal{R}_W$ the ones having width larger than $\frac{\varepsilon^6}{K}A$ being *wide*. Let us assume by now that we remove narrow rectangles from the packing, we will argue later how to include them back. Consider the packing of wide sliced rectangles in the boxes and let us partition this area into stripes of height 1 in such a way that each rectangle is contained in exactly one stripe. Let us shift all the rectangles horizontally to the left as much as possible. Furthermore, let us rearrange the rectangles inside each stripe so that they are sorted non-increasingly by width from left to right.

We use now the standard technique of *linear grouping* in order to reduce the number of possible widths among wide rectangles to a constant. Consider the whole set of wide rectangles in $\mathcal{R}_{sliced}$ piled one on top of each other and consistently sorted non-increasingly by width from bottom to top, meaning that the slices in $\mathcal{R}_{sliced}$ corresponding to the same rectangle in $\mathcal{R}$ are together. We can form groups $H_1, H_2, \ldots, H_t$ of total height exactly εB (except maybe for the last group which might have smaller total height), and just to simplify the analysis let us complete the rectangles in H_1 by moving there all the slices corresponding to the original rectangles present in H_1 (at most δB slices). Since the total height of wide rectangles is at most $\frac{KB}{\varepsilon^6}$, the number of groups t is at most $\frac{K}{\varepsilon^7}$. Notice that for every $i = 2, \ldots, t$, the width of every rectangle in H_i is smaller than the smallest width present in H_{i-1}. We will then round up, for each $i = 2, \ldots, t$, the width of each rectangle in H_i to be the maximum width present in H_i, obtaining $\overline{H_i}$. For each $i \geq 2$ it is possible to pack $\overline{H_i}$ in the space used by H_{i-1}, so if we remove

H_1 from the packing we obtain a feasible packing where wide rectangles have at most $\frac{K}{\varepsilon^7}$ possible widths. Furthermore the total area of the rounded rectangles in the packing is not larger than $a(\mathscr{R}_W)$, and the total height of the discarded rectangles so far is at most $(\varepsilon + \delta)B \leq 2\varepsilon B$.

We will now define the concept of *configuration*. Let $Q = w_1, \ldots, w_t$ be the set of possible widths in the instance after rounding. A configuration C will be a vector of size $|Q|$ where each coordinate contains a non-negative integer number of value at most $\frac{K}{\varepsilon^6}$. Given a stripe with some wide rectangles inside, we will say that the stripe *obeys* a configuration C if for each $i = 1, \ldots, |Q|$, coordinate i of C expresses exactly the number of rectangles of width w_i inside the stripe. It is not difficult to see that there are at most $(K/\varepsilon^7)^{|Q|}$ many configurations.

Let us rearrange the stripes inside each box so that stripes obeying the same configuration appear together. Now if we take all the stripes obeying the same configuration inside a box we can define at most $\frac{K}{\varepsilon^6}$ containers for them, where each container will have the width of the rectangles inside (same for everyone) and the total height of the rectangles inside. If we do this for every box and every configuration, we get a container packing for the wide sliced rectangles into at most $K(K/\varepsilon^7)^{|Q|}\frac{K}{\varepsilon^6} \leq (K/\varepsilon^7)^{|Q|+2}$ containers. Notice that, for each possible width, the total area of the containers of that width and the total area of the packed slices of that width is the same.

We will describe now how to turn this packing into a feasible one without slicing while discarding some rectangles which can be repacked using small extra height. Consider all the containers of width w_1 and we will greedily assign the rectangles of (rounded) width w_1 until we cover them, meaning that for each container we assign rectangles to them until their total height becomes larger than the height of the container for the first time. Notice that since the area of the containers and the area of the rectangles we are packing is the same, this procedure packs all the considered rectangles. If we discard now from each container the last rectangle that was added to it by this procedure then we obtain a feasible packing of rectangles of width w_1 into the containers of width w_1. By repeating this procedure for each possible width we obtain a feasible packing while discarding at most one rectangle per container. The total height of discarded rectangles is at most $(K/\varepsilon^7)^{|Q|+2}\delta B \leq \varepsilon B$.

Now we will include back narrow rectangles to the packing. First of all, we will use only boxes (among the original ones) having width at least $\frac{\varepsilon^2}{K}A$ and height at least $\frac{\delta}{\varepsilon^2}B$ to pack rectangles, so we will discard the wide rectangles inside the boxes not satisfying these properties. The total area of boxes that we will not use (and hence of the discarded rectangles) is at most $K(\frac{\varepsilon^2}{K} + \frac{\delta}{\varepsilon^2})AB \leq \varepsilon^2 AB$, and notice that each one of them has width at most $\frac{\varepsilon^2}{K}B$.

As argued before, the total area of the containers is at most $a(\mathscr{R}_W)$ and the total area of the boxes we are using so far is at least $a(\mathscr{R}_N)+a(\mathscr{R}_W)-\varepsilon^2 AB$, so the total area in the considered boxes outside the containers is at least $a(\mathscr{R}_N)-\varepsilon^2 AB$. We will now pack a subset of the narrow rectangles in this area, and then prove that the rectangles that could not be packed in this space (if any) can be packed in a small extra area container. Notice first that the aforementioned free area can be decomposed into $K(K/\varepsilon^7)^{|Q|}$ rectangular regions as follows: for each maximal group of stripes obeying the same configuration inside a box, the union of the empty space in those stripes is a rectangular region (this includes the case of totally empty stripes). We will now focus only on such regions having height at least $\frac{\delta}{\varepsilon^2}B$, the total area of the boxes we are not using is at most $\varepsilon^2 AB$. Furthermore, we will not use regions having width at most $\frac{\varepsilon^4}{K}A$. Since the width of the original boxes is at least $\frac{\varepsilon^2}{K}$, the total area of such regions is at most $\varepsilon^2 a(Boxes) \leq \varepsilon^2 AB$. So the total area of the remaining regions is at least $a(\mathscr{R}_N) - \varepsilon^2 AB$ and the rectangles we want to pack inside them are smaller in each dimension by a factor of ε^2 so we can pack narrow rectangles inside them using NFDH due to Corollary 8, packing almost everything except for a set of narrow rectangles of total area at most $\varepsilon^2 AB$.

So in total we have discarded a set of rectangles of total height at most εB for which we define an extra horizontal container, plus a set of rectangles having width at most $\frac{\varepsilon^2}{K}A \leq \varepsilon^2 A$ and height at most $\delta B \leq \varepsilon^2 B$ whose total area is at most $\varepsilon^2 AB$. This we can pack into an extra area container of height $2\varepsilon B$ and width $2\varepsilon A$ thanks to Corollary 8.

□

Unfortunately there is no control on the profits of the discarded rectangles in this specific procedure so it is mostly used in the context of problems where every rectangle has to be packed (such as Strip Packing). However, as shown in Appendix A it is possible to ensure that even the profit of P_{disc} is small but for the purposes of the presented results this suffices.

2.4 Including Small Rectangles into a Container Packing

Another useful technique is a way to pack "small" items provided that we have a packing for the rest of the instance which can be decomposed into boxes which are efficient in terms of area. More in detail, let $\mathscr{R}$ be a set of rectangles that can be packed into a given rectangular region, and let ε_{small} be a given parameter. Consider now the set S of rectangles whose dimensions are at most an ε_{small}-fraction of the region where we are packing the whole instance, and suppose we have a packing of non-small items which can be decomposed into $O_\varepsilon(1)$ rectangular regions such that these items are completely contained in the union of these regions (notice that some items might intersect more than one such region), and the area outside of these regions is close to $a(S)$. It turns out that this residual area is sufficient to pack almost all the items of S into a constant number of area containers (not overlapping with the previous rectangular regions) for ε_{small} small enough as the following lemma shows.

Lemma 18 (Small Items Packing Lemma). *Let $\varepsilon > 0$ and Q be a rectangular region of width w and height h, and let $\mathscr{R}$ be a set of rectangles. Suppose we are given a packing of non-small items into Q which can be covered by $O_\varepsilon(1)$ disjoint rectangular regions of total area at most $\min\{(1-2\varepsilon)wh, a(\mathscr{R}\setminus S)+\varepsilon_{area}wh\}$ for some given $\varepsilon_{area} > 0$. Then for ε_{small} small enough it is possible to define $O_{\varepsilon_{small}}(1)$ area containers of width $\frac{\varepsilon_{small}}{\varepsilon}w$ and height $\frac{\varepsilon_{small}}{\varepsilon}h$ not overlapping with the rectangular regions such that it is possible to pack $S' \subseteq S$ of profit $p(S') \geq (1-O(\varepsilon))p(S)$ inside these new area containers.*

Proof. Let A be the total area of the rectangular regions covering the packing and k the number of such regions. We will build a grid of width $\varepsilon' w = \frac{\varepsilon_{small}}{\varepsilon}\cdot w$ and height $\varepsilon' h = \frac{\varepsilon_{small}}{\varepsilon}\cdot h$ inside Q. We delete any cell of the grid that overlaps with some of the rectangular regions in the covering, and call the remaining cells *free*. The new area containers are the free cells.

The total area of the deleted grid cells is, for ε_{small} small enough, at most

$$A + 4k\frac{1}{\varepsilon'}\cdot\varepsilon'^2 wh \leq A + 2\varepsilon^2 N^2 \leq \min\{(1-\varepsilon)wh, a(\mathscr{R}\setminus S)+3\varepsilon^2 wh\}$$

For the sake of simplicity, suppose that any empty space in the packing of $\mathscr{R}$ is filled in with dummy small items of profit 0, so that $a(\mathscr{R}) = wh$. We observe that the area of the free cells is at least $(1-O(\varepsilon))a(S)$: Either, $a(S) \geq \varepsilon wh$ and then the area of the free cells is at least $a(S) - 3\varepsilon^2 wh \geq (1-3\varepsilon)a(S)$; otherwise, we have that the area of the free cells is at least $\varepsilon wh > a(S)$. Therefore we can select a subset of small items $S' \subseteq S$, with $p(S') \geq (1-O(\varepsilon))p(S)$ and area $a(S') \leq (1-O(\varepsilon))a(S)$ that can be fully packed into free cells using NFDH according to Corollary 8. □

2.5 Packing Rectangles with Resource Augmentation

One last important known result that will be extensively used in this thesis is the following PTAS for two-dimensional Geometric Knapsack when we allow to slightly extend one dimension of the given knapsack (while comparing the profit with the optimal profit for the original knapsack), usually known as *one-sided resource augmentation*. The following lemma was essentially proved by Jansen and Solis-Oba [2009][1]. However we strengthen it by adding area guarantees for the containers which will be useful for our purposes. We defer its proof to Appendix A.

Lemma 19 (Resource Augmentation Packing Lemma). *Let $\mathscr{R}'$ be a collection of weighted rectangles that can be packed into a box of width a and height b, and $\varepsilon > 0$ be a given constant. Then there exists a container packing of $\mathscr{R}'' \subseteq \mathscr{R}'$ inside a box of width a and height $(1+\varepsilon)b$ (resp., width $(1+\varepsilon)a$ and height b) such that:*

1. $p(\mathscr{R}'') \geq (1-O(\varepsilon))p(\mathscr{R}')$;
2. The number of containers is $O_\varepsilon(1)$ and their sizes belong to a set of cardinality $n^{O_\varepsilon(1)}$ that can be computed in polynomial time.
3. The total area of the containers is upper-bounded by $a(\mathscr{R}') + \varepsilon \cdot ab$.

[1] In Appendix A we reprove this lemma in a *container-based* form, rather than using LP-based arguments, since this is more convenient for our algorithms. Our version of the lemma might also be a handy tool for future work.

Chapter 3

Improved Approximation Algorithms for Strip Packing

In this chapter we present our results on Strip Packing. In the first part we discuss an asymptotically tight approximation algorithm in the special case when the rectangles of the instance have always small width or small height with respect to the dimensions of the optimal solution inside the strip. Then in the second part we discuss a $\left(\frac{4}{3}+\varepsilon\right)$-approximation in pseudo-polynomial time for the problem. The first result has not yet been published (Gálvez et al. [2019]) while the second one appeared in FSTTCS 2016 (Gálvez et al. [2016]).

We recall that in Strip Packing we are given a parameter $W \in \mathbb{N}$ and a set $\mathcal{R} = \{R_1, \ldots, R_n\}$ of rectangles, each one characterized by a width $w(R_i) \in \mathbb{N}$, $w(R_i) \leq W$, and a height $h(R_i) \in \mathbb{N}$. A packing is a pair $(\mathit{left}(R_i), \mathit{bottom}(R_i)) \in \mathbb{N} \times \mathbb{N}$ for each R_i, with $0 \leq \mathit{left}(R_i) \leq W - w(R_i)$, meaning that the left-bottom corner of R_i is placed in position $(\mathit{left}(R_i), \mathit{bottom}(R_i))$ and its right-top corner in position $(\mathit{right}(R_i), \mathit{top}(R_i))$ where $\mathit{right}(R_i) = \mathit{left}(R_i) + w(R_i)$ and $\mathit{top}(R_i) = \mathit{bottom}(R_i) + h(R_i)$, and it is feasible if the interiors of the rectangles are pairwise disjoint in this embedding. Our goal is to find a feasible packing of minimum *height* $\max_i\{\mathit{top}(R_i)\}$.

A simple reduction from the *Partition* problem shows that the problem cannot be approximated within a factor $\frac{3}{2} - \varepsilon$ for any $\varepsilon > 0$ in polynomial-time unless P=NP. This reduction relies on exponentially large (in n) rectangle widths.

3.0.1 Prior work

Let $OPT = OPT(\mathcal{R})$ denote the optimal height for the considered Strip Packing instance $(\mathcal{R}, W)$, and recall that $h_{\max} = h_{\max}(\mathcal{R})$ (resp. $w_{\max} = w_{\max}(\mathcal{R})$)

is the largest height (resp. width) of any rectangle in $\mathcal{R}$. Observe that trivially $OPT \geq h_{\max}$, and w.l.o.g. we can assume that $W \leq nw_{\max}$. The first non-trivial approximation algorithm for Strip Packing, with approximation ratio 3, was given by Baker et al. [1980]. The First-Fit-Decreasing-Height algorithm (FFDH) by Coffman Jr. et al. [1980] gives a 2.7-approximation. Sleator [1980] gave an algorithm that generates a packing of height $2OPT+\frac{h_{\max}}{2}$, hence achieving a 2.5-approximation. Afterwards, Steinberg [1997] and Schiermeyer [1994] independently improved the approximation ratio to 2. Harren and van Stee [2009] first broke the barrier of 2 with their 1.9396-approximation. The present best $(\frac{5}{3}+\varepsilon)$-approximation is due to Harren et al. [2014].

More recently, algorithms running in pseudo-polynomial time (PPT) for this problem have been developed. More specifically, the running time of a PPT algorithm for Strip Packing is $O((Nn)^{O(1)})$, where $N = \max\{w_{max}, h_{\max}\}$[1]. As Strip Packing is strongly NP-hard (Garey and Johnson [1978]), it does not admit an exact PPT algorithm unless P=NP.

First, Jansen and Thöle [2010] showed a PPT $(3/2+\varepsilon)$-approximation algorithm, and later Nadiradze and Wiese [2016] overcame the $\frac{3}{2}$-inapproximability barrier by presenting a PPT $(\frac{7}{5}+\varepsilon)$-approximation algorithm. After the publication of our extended abstract Gálvez et al. [2016], Adamaszek et al. [2017] proved that there is no PPT $(\frac{12}{11}-\varepsilon)$-approximation algorithm for Strip Packing unless NP $\subseteq$ DTIME($2^{\text{polylog}(n)}$), and this lower bound was later improved to $\left(\frac{5}{4}-\varepsilon\right)$ by Henning et al. [2018]. On the other hand, Jansen and Rau [2017] independently showed a PPT $(4/3+\varepsilon)$-approximation algorithm with running time $(nW)^{1/\varepsilon^{O(2^{1/\varepsilon})}}$ for the case without rotations which was later improved to a $(5/4+\varepsilon)$-approximation by the same authors (Jansen and Rau [2019]).

On other line of research, authors have also considered asymptotic approximation algorithms. Coffman Jr. et al. [1980] described two *level-oriented* algorithms, Next-Fit-Decreasing-Height (NFDH) and First-Fit-Decreasing-Height (FFDH), that achieve asymptotic approximation ratios of 2 and 1.7, respectively. After a sequence of improvements (Golan [1981]; Baker et al. [1981]), the seminal work of Kenyon and Rémila [2000] provided an asymptotic polynomial-time approximation scheme (APTAS) with an additive term $O\left(\frac{h_{\max}}{\varepsilon^2}\right)$. The latter additive term was subsequently improved to $h_{\max}$ by Jansen and Solis-Oba [2009].

In the variant of Strip Packing *with rotations*, we are allowed to rotate the input rectangles by 90° (in other terms, we are free to swap the width and height of any input rectangle). The case with rotations is much less studied in the lit-

[1] For the case without rotations, the polynomial dependence on $h_{\max}$ can indeed be removed with standard techniques.

erature. It seems that most of the techniques that work for the case without rotations can be extended to the case with rotations, however this is not always a trivial task. In particular, it is not hard to achieve a $(2+\varepsilon)$-approximation and the 3/2 hardness of approximation extends to this case as well (Jansen and Solis-Oba [2009]). In terms of asymptotic approximation, Miyazawa and Wakabayashi [2004] gave an algorithm with asymptotic performance ratio of 1.613. Later, Epstein and van Stee [2006] gave a $\frac{3}{2}$ asymptotic approximation. Finally, Jansen and van Stee [2005] achieved an APTAS for the case with rotations.

Strip packing has also been studied in higher dimensional settings. The present best asymptotic approximation for 3-D Strip Packing is due to Jansen and Prädel [2014] who presented a $\frac{3}{2}$-approximation extending techniques from 2-D Bin Packing.

3.1 Strip Packing without Large Rectangles

In this section we study the special but non-trivial case of Strip Packing where all the rectangles are *skewed*, which informally means that every rectangle in the instance is small with respect to the strip in at least one dimension, or equivalently that there are no large rectangles. More in detail, given $\delta > 0$, we say that an instance of Strip Packing is δ*-skewed* if it does not contain rectangles having width larger than δW and height larger than δOPT. The main result we will prove is the following.

Theorem 20. *Let $\varepsilon > 0$. There exists $\delta > 0$ such that there is an algorithm for Strip Packing computing $\left(\frac{3}{2}+\varepsilon\right)$-approximate solutions if the instances are δ-skewed.*

We remark that the algorithm does not need to recognize first if the instance is δ-skewed (which would be NP-hard as it would require to compute the optimal height); in turn it always returns a feasible solution but its approximation ratio is guaranteed only if the instance satisfies the requirements. Furthermore, we complement the result with an almost matching lower bound for this case.

3.1.1 Preliminaries

Given a set of rectangles $\mathscr{R}' \subseteq \mathscr{R}$, we recall that $w(\mathscr{R}') = \sum_{R_i \in \mathscr{R}'} w(R_i)$, $h(\mathscr{R}') = \sum_{R_i \in \mathscr{R}'} h(R_i)$ and $a(\mathscr{R}') = \sum_{R_i \in \mathscr{R}} h(R_i) w(R_i)$.

The operation of changing the bottom-left corner of a rectangle R_i in a given packing from $(left(R_i), bottom(R_i))$ to $(left(R_i), bottom(R_i)+a)$ will be denoted by *shifting* R_i *vertically by* a. Analogously, changing the corner from $(left(R_i), bottom(R_i))$ to $(left(R_i)+a, bottom(R_i))$ will be denoted by *shifting* R_i *horizontally by* a. These operations are only allowed if the resulting packing is feasible and remains inside the strip.

A monotone polygonal curve is a curve in the space consisting of the union of axis parallel lines such that the obtained curve is connected and every vertical line intersects the curve either on a single point or on a single vertical line. We say that a rectangle R_i in the packing *lies above (resp. below)* a polygonal curve P if for any $x_1 \in [left(R_i), left(R_i) + w(R_i)]$ we have that $bottom(R_i)$ (resp. $top(R_i)$) is larger (resp. smaller) than the y-coordinate of P at x-coordinate x_1.

From now on we will assume that instance $\mathcal{R}$ is δ-skewed for some $\delta > 0$ to be fixed, and let OPT be the height of the optimal solution. We will assume (by possibly scaling up the heights) that OPT is an even number.

Let $\varepsilon > 0$ and assume for simplicity that $\frac{1}{\varepsilon} \in \mathbb{N}$. We will classify the rectangles according to their heights as follows:

- A rectangle R_i is *Tall* if $h(R_i) > \frac{1}{2}OPT$,
- A rectangle R_i is *Vertical* if $h(R_i) \in (\delta OPT, \frac{1}{2}OPT]$, and
- A rectangle R_i is *Short* if $h(R_i) \leq \delta OPT$.

We use T, V and S to denote tall, vertical and short rectangles respectively. Notice that the set of short rectangles S includes the rectangles which are small in both dimensions.

3.1.2 Hardness of Approximation

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

Lemma 21. *Given $\delta > 0$ and $\varepsilon > 0$, there is no $\left(\frac{3}{2} - \varepsilon\right)$-approximation Strip Packing even when restricted to δ-skewed instances only, unless P=NP.*

Proof. We will prove this result via a reduction from *Partition* problem. Consider an instance of Partition consisting of a set of integer numbers $P = \{x_1, \ldots, x_n\}$ and let $p = \sum_{i=1}^{n} x_i$. We define our Strip Packing instance as follows: The width of the strip will be $W = (1+\delta/4)M$ where $M = \frac{2p}{\delta}$. Also, we will have $n+\frac{4}{\delta}$ rectangles in the instance, from which $\frac{4}{\delta}$ will have height 1 and width $\frac{\delta}{2}M$ (*dummy rectangles*)

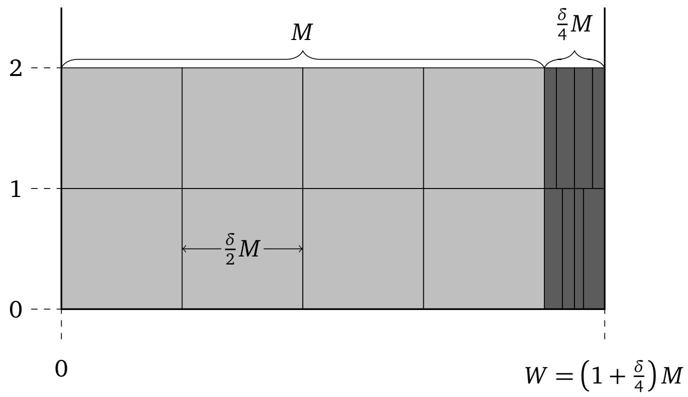


Figure 3.1. Construction from Lemma 21. Light gray rectangles represent dummy rectangles and dark gray rectangles represent partition rectangles.

and the remaining n rectangles will have, for each $i = 1, \ldots, n$, height 1 and width x_i (*partition rectangles*). Notice that the instance is indeed δ-skewed as the width of the rectangles is either $\frac{\delta}{2}M \leq \frac{\delta}{2}W$ or at most $p = \frac{\delta}{2}M \leq \frac{\delta}{2}W$.

We will prove now that the Partition instance is YES if and only if the optimal height of the Strip Packing instance is 2. Since all the heights in the instance are 1, this would conclude the proof of the claim. Notice that if the Partition instance is YES then we can pack one next to each other $\frac{2}{\delta}$ dummy rectangles plus one side of the partition since their total width would be $M + \frac{p}{2} = \left(1 + \frac{\delta}{4}\right)M$. We then analogously pack the rest of the rectangles on top, obtaining a packing of height 2 which is optimal as the total area of the rectangles is $2W$ (see Figure 3.1). On the other hand, if the optimal height of the Strip Packing instance is 2, the region $[0, W] \times [0, 2]$ in the strip must be fully occupied by rectangles. This actually implies that the horizontal line $y = 1$ does not intersect the interior of any rectangle in the packing: if it does, then the space below that rectangle cannot be occupied by another one as their heights are 1. This divides the solution into two rows of height 1 and width W which are completely filled with rectangles. The only way to place dummy rectangles using only two rows is to have exactly $\frac{2}{\delta}$ in each row (as the largest total width below W that they can sum up to is M and their total width is $2M$), hence the remaining partition rectangles in each row have total width exactly $\frac{p}{2}$ forming then a YES instance for Partition. □

3.1.3 Existence of a Structured Solution

In this section we will prove our main structural result.

Theorem 22. *Given a δ-skewed instance of Strip Packing $(\mathcal{R}, W)$, there exists a feasible container packing of final height $\left(\frac{3}{2} + O(\varepsilon)\right) OPT$. Furthermore, the possible sizes of the containers belong to a set of polynomial size, and it is possible to include an empty rectangular region inside $[0, W] \times [0, \left(\frac{3}{2} + O(\varepsilon)\right) OPT]$ of width $\varepsilon^2 W$ and height $\left(\frac{1}{2} + \varepsilon\right) OPT$.*

In order to achieve this, we will first show a way to pack $T \cup S$ in such a way that the remaining space in the packing can be decomposed into a constant number of rectangular regions where rectangles from V can be included afterwards.

Packing of Tall and Short Rectangles

Let us assume first that short rectangles can be *horizontally sliced*, meaning that each short rectangle R_i is replaced by $h(R_i)$ rectangles of height 1 and width $w(R_i)$. Later we can turn such a packing into a feasible one without slicing while increasing the final height of the solution by a negligible amount thanks to Lemma 17. Let S_{sliced} be the set of horizontal slices.

Lemma 23. *It is possible to pack $T \cup S_{sliced}$ into the strip in such a way that:*

- *The final height of the packing is $\frac{3}{2}OPT$;*
- *Rectangles in T are packed one next to each other in the bottom-left part of the strip, sorted non-increasingly by height from left to right;*
- *There exist two non-crossing monotone polygonal curves $\mathcal{C}_1$ and $\mathcal{C}_2$ with integral vertices starting at $x = 0$ and ending at $x = W$ such that rectangles in T lie below $\mathcal{C}_1$, rectangles in S_{sliced} lie above $\mathcal{C}_2$, and the total area in the strip above $\mathcal{C}_1$ and below $\mathcal{C}_2$ is at least $\frac{1}{2}OPT \cdot W + a(V)$.*

Proof. Consider the optimal solution and let us remove V from it. Let us draw the horizontal line $y = \frac{1}{2}OPT$ and partition S_{sliced} into two sets S_{sliced}^{top} and S_{sliced}^{bottom} corresponding to the rectangles in S_{sliced} which lie above and below the line $y = \frac{1}{2}OPT$ respectively (notice that this line does not intersect any rectangle in S_{sliced} as OPT is even). If we shift up the rectangles in S_{sliced}^{bottom} by OPT we obtain a feasible packing (since the region $[0, W] \times [OPT, \frac{3}{2}OPT]$ was empty) with final height at most $\frac{3}{2}OPT$. Notice that every rectangle in T intersects the line $y = \frac{1}{2}$. Now let us shift down each rectangle R_i in T by $bottom(R_i)$ so that its bottom-left corner becomes $(left(R_i), 0)$.

Suppose that the tall rectangles are labeled in such a way that $h(R_1) \geq h(R_2) \geq \cdots \geq h(R_{|T|})$. We will now describe a procedure to consistently shift them to the

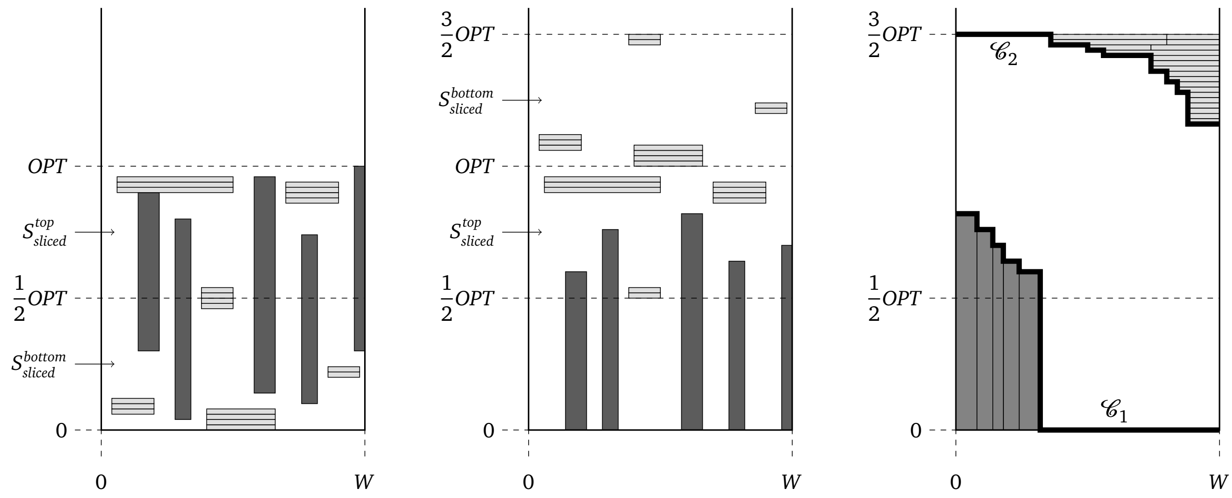


Figure 3.2. Depiction of the proof of Lemma 23. Left: Packing of $T \cup S_{sliced}$ in the optimal solution. Light gray rectangles correspond to S_{sliced}, dark gray rectangles correspond to T. Center: By shifting S^{bottom}_{sliced} to the top we can shift down the rectangles in T. Right: We can shift now horizontally rectangles in T and sort stripes of S_{sliced} to obtain the claimed structure.

left one by one so that the second claim of the lemma is satisfied. Notice that the horizontal line $y = h(R_1)$ does not intersect any rectangle, so we can shift rectangle R_1 to the left by $left(R_1)$ and shift to the right all the rectangles contained in the region $[0, left(R_1)] \times [0, bottom(R_1)]$ by $w(R_1)$ obtaining a feasible solution. If we now restrict ourselves to the region $[w(R_1), W] \times [0, h(R_1)]$ and the rectangles contained inside we can recourse this argument in order to shift R_2 to the left so that it is contiguous to R_1 and then continue, obtaining in the end a feasible solution that satisfies the second claim of the lemma. Furthermore, we can define the polygonal curve $\mathscr{C}_1$ just from the boundary of the rectangles in T: starting at the point $(0, h(R_1))$ we draw an horizontal line to the right up to $(w(R_1), h(R_1))$, then a vertical line down to $(w(R_1), h(R_2))$, then a horizontal line to the right up to $(w(R_1)+w(R_2), h(R_2))$ and continue like that until reaching the point $(w(T), h(R_{|T|})$. Then we finish the curve by adding a vertical line from that point down to $(w(T), 0)$ and then an horizontal line to the right up to $(W, 0)$. Every rectangle in T clearly lies below $\mathscr{C}_1$ and furthermore the total area below $\mathscr{C}_1$ in the strip is exactly $a(T)$.

Consider now the current packing of S_{sliced} which is completely contained above $\mathscr{C}_1$ and below $y = \frac{3}{2}OPT$. If we divide this space into horizontal stripes of height 1 just by drawing horizontal lines at each possible integer Y-coordinate, every rectangle in S_{sliced} belongs to exactly one such stripe. Let us shift all the rectangles in S_{sliced} to the right as much as possible. Analogously to the case of T, we can now rearrange the stripes (hence the corresponding rectangles inside) vertically so that they are packed one on top of each other, sorted non-increasingly from top to bottom according to the total width of the rectangles inside them and in such a way that the final height of the current packing is exactly $\frac{3}{2}OPT$. Analogously to the case of tall rectangles, we can define the polygonal curve $\mathscr{C}_2$ from the boundary of the rectangles, obtaining that the area of the region above $\mathscr{C}_2$ and below $y = \frac{3}{2}OPT$ in the strip is exactly $a(S)$. See Figure 3.2 for a depiction of the procedure.

Due to the way we defined the curves $\mathscr{C}_1$ and $\mathscr{C}_2$ and the fact that $a(T)+a(S)+a(V) \leq W \cdot OPT$, we conclude that the region of the strip contained between $\mathscr{C}_1$ and $\mathscr{C}_2$ has area at least $\frac{3}{2}OPT \cdot W - a(T) - a(S) \geq \frac{1}{2}OPT \cdot W + a(V)$, concluding the proof of the lemma. □

Now our goal is to refine a bit the packing obtained from Lemma 23 in order to create $O_\varepsilon(1)$ boxes for S_{sliced} and then apply Lemma 17 to obtain a feasible packing without slicing of $T \cup S$. Notice that we can include the extra containers from Lemma 17 on top of the current solution which would increase its final height by at most $O(\varepsilon)OPT$. The following lemma summarizes the properties of the solution we can obtain plus some extra guarantees that will be useful in the latter steps.

Lemma 24. *It is possible to pack $T \cup S$ inside the strip in such a way that:*

- *The final height of the packing is $\left(\frac{3}{2} + O(\varepsilon)\right)OPT$;*
- *Rectangles in T are packed one next to each other in the bottom-left part of the strip, sorted non-increasingly by height from left to right, and fully contained in at most $\frac{1}{\varepsilon}$ rectangular boxes;*
- *Rectangles in S are packed into $O_\varepsilon(1)$ containers which are fully contained in at most $\frac{2}{\varepsilon}$ rectangular boxes; and*
- *The total area inside the region $[0, W] \times [0, \left(\frac{3}{2} + O(\varepsilon)\right)OPT]$ and outside the aforementioned boxes is at least $\left(\frac{1}{2} + O(\varepsilon)\right)OPT \cdot W + a(V)$ and can be partitioned into at most $\frac{3}{\varepsilon}$ rectangular boxes.*

Proof. Consider the packing obtained from Lemma 23 and let us shift vertically all the rectangles in S_{sliced} by $O(\varepsilon OPT)$ and the curve $\mathscr{C}_2$ accordingly. The rectangles still lie above the curve $\mathscr{C}_2$ and the final height of the current solution is

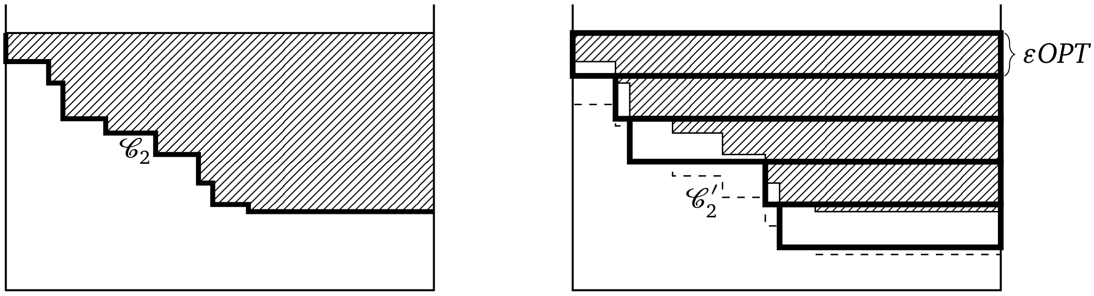


Figure 3.3. Description of the construction of boxes for S_{sliced}. Left: Packing of S_{sliced} lying above the curve $\mathscr{C}_2$. Right: After shifting up the rectangles by εOPT we can create boxes lying above $\mathscr{C}_2'$ (dashed curve).

$\left(\frac{3}{2}+O(\varepsilon)\right)OPT$. We will now define a set of at most $\frac{2}{\varepsilon}$ rectangular boxes of total area at most $a(S)+\varepsilon OPT\cdot W$ and such that every rectangle in S_{sliced} is contained in exactly one box. Let $\mathscr{C}_2'$ be the polygonal curve resulting from shifting down $\mathscr{C}_2$ by εOPT. We define the first box by a vertical line starting at the left-topmost point of $\mathscr{C}_2$ and going down by εOPT, which is a point from $\mathscr{C}_2'$, and then we draw an horizontal line to the right until $x=W$. Let p_1 be the intersection between the previous horizontal line and $\mathscr{C}_2$. Then we define our second box starting from p_1 and moving vertically down εOPT distance (which is a point from $\mathscr{C}_2'$) and then drawing an horizontal line to the right until $x=W$. By iterating this procedure until it is not possible to continue (meaning that the last horizontal line does not intersect $\mathscr{C}_2$), we obtain our set of boxes (see Figure 3.3).

Notice that every box has height exactly εOPT, so the total number of boxes is at most $\frac{1}{\varepsilon}+1\leq\frac{2}{\varepsilon}$ (as there are no short rectangles below $y=\frac{1}{2}OPT$). Furthermore these boxes are completely contained above $\mathscr{C}_2'$ and below $y=\left(\frac{3}{2}+O(\varepsilon)\right)OPT$ so their total area is at most $a(S)+\varepsilon W\cdot OPT$ and each rectangle in S_{sliced} is contained in exactly one box as $\mathscr{C}_2$ is completely contained in the union of the boxes.

Now we can apply Lemma 17 to this packing of S_{sliced} and obtain a feasible packing of $T\cup S$ plus the two extra containers which we can pack on top of the current solution by increasing the final height by at most $O(\varepsilon)OPT$. Notice that by shifting every rectangle in S by εOPT we can round up the height of the rectangles in T to multiples of εOPT, obtaining at most $\frac{1}{\varepsilon}$ boxes (which will actually be containers) for T by grouping rectangles of equal rounded height.

It just remains to partition the free area outside the boxes. If we extend the vertical boundaries of the boxes (one line per box will matter due to their structure in the solution) we naturally decompose that region into a set of at most $\frac{3}{\varepsilon}$ rectangular boxes of total area at least $a(V)+\left(\frac{1}{2}+O(\varepsilon)\right)OPT$ (see Figure 3.4

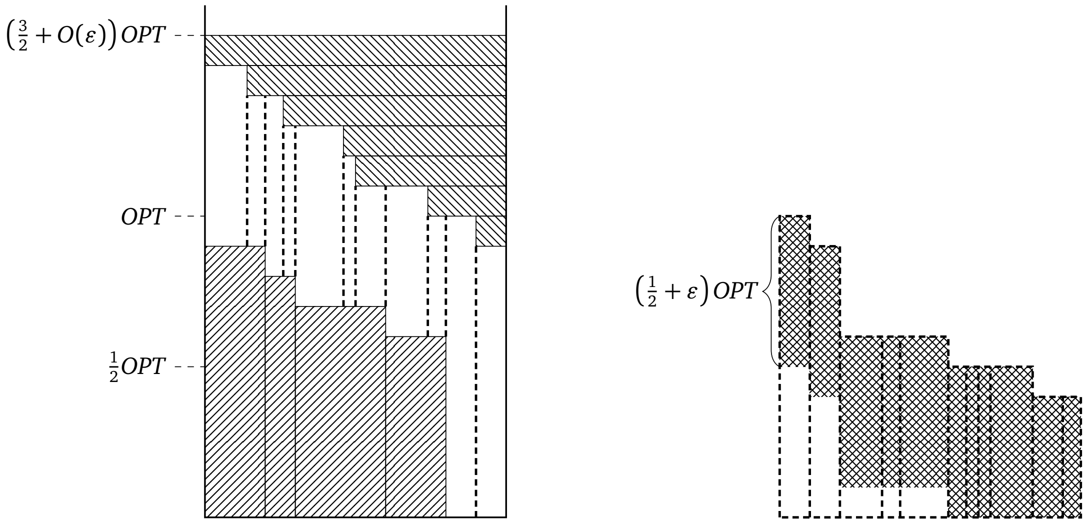


Figure 3.4. Description of the Packing of V_{sliced}. Left: Packing of $T \cup S$ as described in Lemma 24. The dashed vertical lines naturally induce a partition of the free area into $O_\varepsilon(1)$ boxes. Right: Boxes in the free area sorted by height. By ignoring $\left(\frac{1}{2}+\varepsilon\right)OPT$ height from each box we can cover the free white area with V_{sliced} and even reserve space for future discarded vertical rectangles.

(Left)). □

Including Vertical rectangles

In this section we will build on the packing obtained from Lemma 24 in order to include V_{sliced} in the given free boxes. Then we can apply Lemma 17 to obtain a feasible solution without slicing. However, in this case we cannot simply pack the extra containers on top as their height can be as large as $\frac{1}{2}OPT$ but we will prove that these extra containers can be included together with V in the free area between T and S.

Lemma 25. *Consider the set of $\frac{3}{\varepsilon}$ free boxes from Lemma 24. It is possible to pack V_{sliced} inside these boxes and furthermore define an empty rectangular region of height $\left(\frac{1}{2}+\varepsilon\right)OPT$ and width $3\varepsilon W$ contained in the union of these boxes.*

Proof. Consider the set of given boxes one next to each other and sorted non-decreasingly by height. We will say that for each box its top region of height $\left(\frac{1}{2}+\varepsilon\right)OPT$ is *forbidden* and the rest is *usable* (if a box has height less than $\left(\frac{1}{2}+\varepsilon\right)OPT$ then all its area is forbidden). See Figure 3.4 (Right) for a depiction of this partition. Notice that the total usable area is at least $a(V)+O(\varepsilon)OPT$.

We will now pack V_{sliced} trying to cover the usable area as follows: We first divide the boxes into vertical stripes of width 1 and consider them in the order of non-increasing height. Let $\mathscr{S}_{usable}$ be the set of stripes having non-empty usable area. We will pack the rectangles from V_{sliced} in a Next-Fit fashion into $\mathscr{S}_{usable}$, meaning that we start opening the leftmost stripe and iteratively pack rectangles (in any arbitrary order) in the current open stripe; if a rectangle does not fit we close the stripe and open the leftmost empty stripe. Notice that by doing so every stripe containing some rectangle (except probably the last one) has its usable area completely covered, which implies that we used only stripes from $\mathscr{S}_{usable}$. Furthermore, since $a(\mathscr{S}_{usable}) \geq a(V) + O(\varepsilon)OPTW + \frac{OPT}{2}w(\mathscr{S}_{usable})$ and the total area of the stripes we used is at most $a(V) + \frac{OPT}{2}w(\mathscr{S}_{usable})$, we have a set of contiguous free stripes with non-empty usable area having total width at least $O(\varepsilon)W$ (as their height is at most $\left(\frac{3}{2} + O(\varepsilon)\right)OPT$). We can place a box of width $3\varepsilon W$ and height $\left(\frac{1}{2} + \varepsilon\right)OPT$ in these stripes, concluding the proof. □

Now we can obtain a feasible solution decomposed into containers as follows: Starting from the solution given by Lemma 24 we can pack V_{sliced} into the boxes thanks to Lemma 25 and then use Lemma 17 to obtain a container packing for V inside the boxes. We still have some extra containers, one of size $2\varepsilon W \times 2\varepsilon OPT$ that can be packed on top of the current solution and one of height at most $\frac{1}{2}OPT$ and width at most εW which can be packed in the free box left from Lemma 25. In order to restrict the possible sizes of the containers we apply Lemma 9 with the area of the rectangles as profit, obtaining an extra vertical container of width εW and height at most $\frac{1}{2}OPT$ which can again be packed in the aforementioned free box. Having all this we can conclude the proof of Theorem 22 as there is still an empty box of width εW and height $\left(\frac{1}{2} + \varepsilon\right)OPT$ inside the region $[0, W] \times \left[0, \left(\frac{3}{2} + O(\varepsilon)\right)OPT\right]$.

3.1.4 Algorithm

In this section we present our final algorithm. It is a slight modification of the PTAS for container packings described in Section 2.2. We will assume that the instance is δ-skewed for $\delta \leq (\varepsilon)^{(1/\varepsilon)^{O(1)}}$ so that all the previous lemmas hold. We start by computing a value OPT' such that $OPT \leq OPT' \leq (1+\varepsilon)OPT$. This can be done using any 2-approximation APX for Strip Packing and then guessing the best value of the form $(1 + k\varepsilon)\frac{APX}{2}$, $k \in \{0, \dots, 1/\varepsilon\}$. One of them must satisfy $OPT \leq (1 + k\varepsilon)\frac{APX}{2} \leq OPT + \varepsilon\frac{APX}{2}$.

It would be desirable to guess the set of containers from Theorem 22 and try to assign rectangles to them using our PTAS for container packings (Theorem 16).

However, some rectangles will be left out using the previous PTAS and must be repacked. To this end, the discarded rectangles must have height at most $\frac{1}{2}OPT$ so that they can be repacked in the free space left by Theorem 22.

This is achievable due to the simple structure of tall rectangles in the constructed solution (see Lemma 24). So we first guess the tall rectangles, which can be done by sorting the rectangles non-increasingly by height and guessing the smallest tall rectangle. Then we just pack them one next to each other in this order in the left-bottom corner of the strip. Now we enumerate all the possible subsets of non-overlapping containers for the remaining rectangles to be packed, where the number and sizes of the containers are properly bounded. In particular, there are at most $O_\varepsilon(1)$ containers and there is a set of size $n^{O_\varepsilon(1)}$ that we can compute in polynomial time such that the height and the width of each container is contained in this set as stated in Theorem 22.

We compute an approximate solution for each resulting set of containers using the PTAS from Theorem 16 with accuracy ε^4 and using the area of each rectangle as its profit. There exists a set of containers for which the discarded rectangles has total area $\varepsilon^4 W \cdot OPT$ and that can be packed into height at most $\left(\frac{3}{2} + O(\varepsilon)\right) OPT$ while leaving an empty rectangular region of height $\left(\frac{1}{2} + \varepsilon\right) OPT$ and width $\varepsilon^2 W$ due to Theorem 22.

We will assume now that we deal with the aforementioned container packing (though this procedure is done for every possible combination). We now will repack the remaining rectangles into three extra containers: one horizontal container of width W and height $\varepsilon OPT'$, one vertical container of width $\varepsilon^2 W$ and height $\frac{1}{2}OPT'$ and an area container of width $2\varepsilon W$ and height $2\varepsilon OPT'$. Let us partition the set of remaining rectangles into three sets: A_1 are the rectangles of width at least $\varepsilon^2 W$, which have total height at most $\varepsilon^2 OPT$ hence being packable in the horizontal container; A_2 are the rectangles of width smaller than $\varepsilon^2 W$ but height at least $\varepsilon^2 OPT'$ whose total width is at most $\varepsilon^2 W$ and hence are packable into the vertical container; and finally the remaining rectangles of height at most $\varepsilon^2 OPT'$ and width at most $\varepsilon^2 W$ which can be packed into the area container due to Corollary 8. The horizontal and area containers can be packed on top of the current packing while the vertical container can be placed inside the free space left by Theorem 22. This implies that there exists a set of containers computed by the algorithm that can be packed into the strip using height $\left(\frac{3}{2} + O(\varepsilon)\right) OPT$ so we output the computed solution of smaller height that packs all the rectangles into the described containers.

This concludes the proof of Theorem 20.

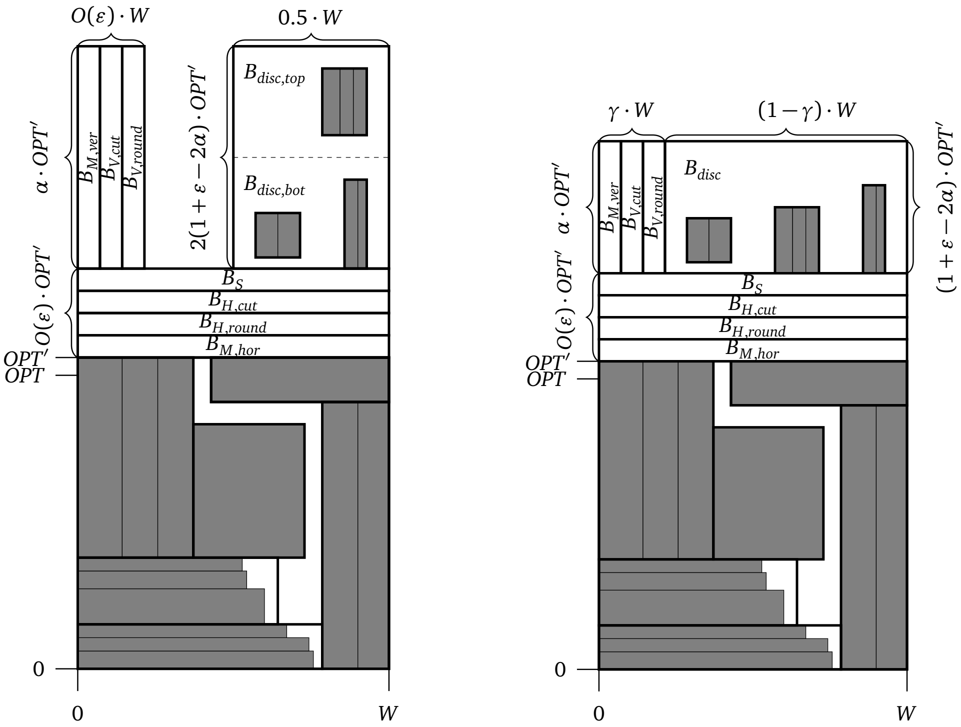
3.2 A PPT $(4/3 + \varepsilon)$-approximation for Strip Packing

In this section we show our progress on the PPT approximability of Strip Packing, by presenting an improved $(\frac{4}{3} + \varepsilon)$-approximation. Our approach refines the result presented by Nadiradze and Wiese [2016], which modulo several technical details works as follows: let $\alpha \in [1/3, 1/2)$ be a proper constant parameter, and define a rectangle R_i to be *tall* if $h(R_i) > \alpha \cdot OPT$. They prove that the optimal packing can be structured into a constant number of rectangular boxes, that occupy a total height of $OPT' \leq (1 + \varepsilon)OPT$ inside the vertical strip. Some rectangles are not fully contained in one box (they are *cut* by some box). Among them, tall rectangles remain in their original position. All the other cut rectangles are repacked on top of the boxes: part of them in a horizontal box of size $W \times O(\varepsilon)OPT$, and the remaining ones in a vertical box of size $O(\varepsilon W) \times \alpha\, OPT$ (that we next imagine as placed on the top-left of the packing under construction).

Some of these boxes contain only relatively high rectangles (including tall ones) of relatively small width. The next step is a rearrangement of the rectangles inside one such *vertical* box $\overline{B}$ (see Figure 3.7a), say of size $\overline{w} \times \overline{h}$: they first slice non-tall rectangles into unit width rectangles (this slicing can be finally avoided with standard techniques). Then tall rectangles are shifted to the top/bottom of $\overline{B}$, shifting sliced rectangles consequently (see Figure 3.7b). Now they discard all the (sliced) rectangles completely contained in a central horizontal region of size $\overline{w} \times (1 + \varepsilon - 2\alpha)\overline{h}$, and then *nicely rearrange* the remaining rectangles into a constant number of *sub-boxes* (excluding possibly a few more non-tall rectangles, that can be placed in the additional vertical box).

These discarded rectangles can be packed into 2 extra boxes of size $\frac{\overline{w}}{2} \times (1+\varepsilon - 2\alpha)\overline{h}$ (see Figure 3.7d). In turn, the latter boxes can be packed into two *discarded* boxes of size $\frac{W}{2} \times (1 + \varepsilon - 2\alpha)OPT'$, that we can imagine as placed, one on top of the other, on the top-right of the packing. See Figure 3.5a for an illustration of the final packing. This leads to a total height of $(1 + \max\{\alpha, 2(1-2\alpha)\} + O(\varepsilon)) \cdot OPT$, which is minimized by choosing $\alpha = \frac{2}{5}$.

Our main technical contribution is a repacking lemma that allows one to repack a small fraction of the discarded rectangles of a given box inside the free space left by the corresponding sub-boxes (while still having $O_\varepsilon(1)$ many sub-boxes in total). This is illustrated in Figure 3.7e. This way we can pack all the discarded rectangles into a *single* discarded box of size $(1-\gamma)W \times (1+\varepsilon-2\alpha)OPT'$, where γ is a small constant depending on ε, that we can place on the top-right of the packing. The vertical box where the remaining rectangles are packed still fits to the top-left of the packing, next to the discarded box. See Figure 3.5b for an illustration. Choosing $\alpha = 1/3$ gives the claimed approximation factor.



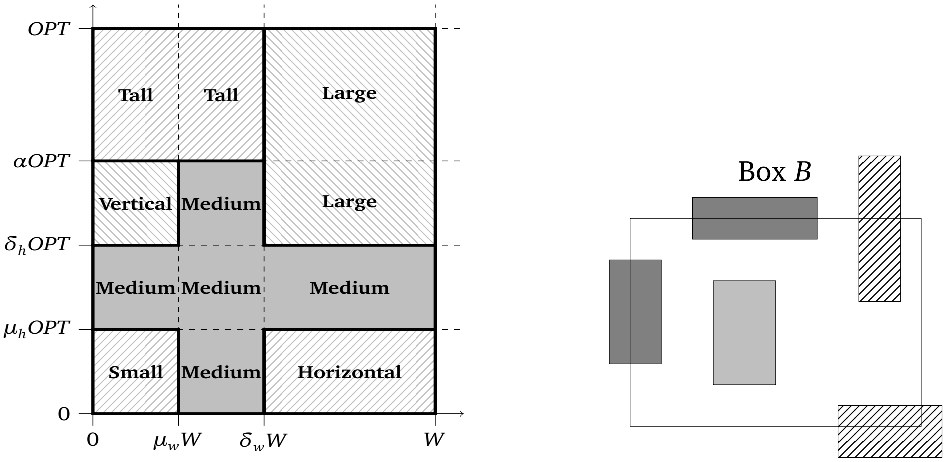
(a) Final packing obtained by Nadiradze and Wiese [2016].

(b) Final packing obtained in this work. Here γ is a small constant depending on ε.

Figure 3.5. Comparison of final solutions.

We remark that the basic approach by Nadiradze and Wiese strictly requires that at most 2 tall rectangles can be packed one on top of the other in the optimal packing, hence imposing $\alpha \geq 1/3$. Thus in some sense this work pushes their approach to its limit.

The algorithm from Nadiradze and Wiese [2016] is not directly applicable to the case when 90° rotations are allowed. In particular, they use a linear program to pack some rectangles. When rotations are allowed, it is unclear how to decide which rectangles are packed by the linear program. We use our *container*-based approach to circumvent this limitation, which allows us to pack all the rectangles using dynamic programming. This way we achieve a PPT $(4/3+\varepsilon)$-approximation for Strip Packing with rotations, breaking the polynomial-time approximation barrier of $3/2$ for that variant as well.



(a) Each rectangle is represented as a point on the plane with x (resp., y) coordinate indicating its width (resp., height).

(b) Gray rectangles are nicely cut by B, dashed ones are cut but not nicely cut by B, and light gray one is not cut by B.

Figure 3.6. Illustration of some of the preliminary definitions.

3.2.1 Preliminaries

Let $0 < \varepsilon < \alpha$, $\alpha \geq \frac{1}{3}$, and assume for simplicity that $\frac{1}{\varepsilon} \in \mathbb{N}$. We first classify the input rectangles into six groups according to parameters $\delta_h, \delta_w, \mu_h, \mu_w$ satisfying $\varepsilon \geq \delta_h > \mu_h > 0$ and $\varepsilon \geq \delta_w > \mu_w > 0$, whose values will be chosen later (see also Figure 3.6a). A rectangle R_i is

- *Large* if $h(R_i) \geq \delta_h OPT$ and $w(R_i) \geq \delta_w W$.
- *Tall* if $h(R_i) > \alpha OPT$ and $w(R_i) < \delta_w W$.
- *Vertical* if $h(R_i) \in [\delta_h OPT, \alpha OPT]$ and $w(R_i) \leq \mu_w W$,
- *Horizontal* if $h(R_i) \leq \mu_h OPT$ and $w(R_i) \geq \delta_w W$,
- *Small* if $h(R_i) \leq \mu_h OPT$ and $w(R_i) \leq \mu_w W$;
- *Medium* in all the remaining cases, i.e., if $h(R_i) \in (\mu_h OPT, \delta_h OPT)$, or $w(R_i) \in (\mu_w W, \delta_w W)$ and $h(R_i) \leq \alpha OPT$.

We use L, T, V, H, S, and M to denote large, tall, vertical, horizontal, small, and medium rectangles, respectively. We remark that, differently from Nadiradze and Wiese [2016], we need to allow $\delta_h \neq \delta_w$ and $\mu_h \neq \mu_w$ due to some additional constraints in our construction (see Section 3.2.5).

Notice that according to this classification, every vertical line across the optimal packing intersects at most two tall rectangles. The following lemma allows us to choose $\delta_h, \delta_w, \mu_h$ and μ_w in such a way that δ_h and μ_h (δ_w and μ_w, respectively) differ by a large factor, and medium rectangles have small total area.

Lemma 26. *Given a polynomial-time computable function $f : (0,1) \to (0,1)$, with $f(x) < x$, any constant $\varepsilon \in (0,1)$, and any positive integer k, we can compute in polynomial time a set Δ of $T = 2(\frac{1}{\varepsilon})^k$ many positive real numbers upper bounded by ε, such that there is at least one number $\delta_h \in \Delta$ so that $a(M) \leq \varepsilon^k \cdot OPT \cdot W$ by setting $\mu_h = f(\delta_h)$, $\mu_w = \frac{\varepsilon\mu_h}{12}$, and $\delta_w = \frac{\varepsilon\delta_h}{12}$.*

Proof. Let $T = 2(\frac{1}{\varepsilon})^k$. Let $y_1 = \varepsilon$, and, for each $j \in \{1,\dots,T\}$, define $y_{j+1} = f(y_j)$. Let $x_j = \frac{\varepsilon y_j}{12}$. For each $j \leq T$, let $W_j = \{R_i \in \mathscr{R} : w(R_i) \in [x_{i+1}, x_i)\}$ and similarly $H_j = \{R_i \in \mathscr{R} : h(R_i) \in [y_{i+1}, y_i)\}$. Observe that sets W_j (respectively H_j) are pairwise disjoint and the total area of rectangles in $\bigcup W_i$ ($\bigcup H_i$ respectively) is at most $W \cdot OPT$. Thus, there exists a value $\bar{j}$ such that the total area of the elements in $W_{\bar{j}} \cup H_{\bar{j}}$ is at most $\dfrac{2OPT \cdot W}{T} = \varepsilon^k \cdot OPT \cdot W$. By choosing $\delta_h = y_{\bar{j}}$, $\mu_h = y_{\bar{j}+1}$, $\delta_w = x_{\bar{j}}$, $\mu_w = x_{\bar{j}+1}$ all the conditions of the lemma are fulfilled. □

Function f and constant k will be chosen later. From now on, assume that $\delta_h, \delta_w, \mu_h$ and μ_w are chosen according to Lemma 26.

3.2.2 Overview of the Algorithm

We next overview some of the basic results from Nadiradze and Wiese [2016] that are required for our result. We define the constant $\gamma := \frac{\varepsilon\delta_h}{2}$, and w.l.o.g. assume $\gamma \cdot OPT \in \mathbb{N}$.

Let us temporarily remove small rectangles S from the instance. We will pack all the remaining rectangles $L \cup H \cup T \cup V \cup M$ into a sufficiently small number of boxes embedded into the strip. By using Lemma 18 it is possible to include them back afterwards.

The following lemma from Nadiradze and Wiese [2016] allows one to round the heights and positions of rectangles of large enough height, without increasing much the height of the packing.

Lemma 27 (Nadiradze and Wiese [2016]). *There exists a feasible packing of height $OPT' \leq (1+\varepsilon)OPT$ where: (1) the height of each rectangle in $L \cup T \cup V$ is rounded up to the closest integer multiple of $\gamma \cdot OPT$ and (2) their x-coordinates are as in the optimal solution and their y-coordinates are integer multiples of $\gamma \cdot OPT$.*

We next focus on rounded rectangle heights (i.e., implicitly replace $L \cup T \cup V$ by their rounded version) and on this slightly suboptimal solution of height OPT'.

The following lemma helps us to pack rectangles in M in a structured way.

Lemma 28. *If k in Lemma 26 is chosen sufficiently large, all the rectangles in M can be packed in polynomial time into a box $B_{M,hor}$ of size $W \times O(\varepsilon)OPT$ and a vertical container $B_{M,ver}$ of size $(\frac{\gamma}{3}W) \times (\alpha OPT)$. Furthermore, $B_{M,hor}$ can be decomposed into one horizontal and one area container.*

Proof. Consider first the rectangles in $\mathscr{A}_1 := \{R_i \in M : h(R_i) \leq \varepsilon^2 OPT, w(R_i) \leq \varepsilon^2 W\}$. Due to Lemmas 7 and 26 we know that they can be packed using NFDH into an area container of height εOPT and width W as $\varepsilon^k \leq (1-2\varepsilon^2)\varepsilon$. Next we proceed to pack rectangles in $\mathscr{A}_2 := \{R_i \in M : w(R_i) > \varepsilon^2 W\}$. Due to Lemma 26, the total area of this set is at most $\varepsilon^k OPT \cdot W$, and hence we can pack them into a single horizontal container of width W and height at most $\varepsilon^{k-2} OPT \leq \varepsilon OPT$. It is easy to see that we can put these two containers together one on top of each other inside box $B_{M,hor}$.

We next pack $\mathscr{A}_3 := M \setminus (\mathscr{A}_1 \cup \mathscr{A}_2) = \{R_i \in M : h(R_i) > \varepsilon^2 OPT\}$ into a vertical container $B_{M,ver}$ of size $(\frac{\gamma}{3}W) \times (\alpha OPT)$. Recall that $\gamma := \frac{\varepsilon\delta_h}{2}$. Note that, for each $R_i \in \mathscr{A}'$, we have $w(R_i) \in (\mu_w W, \delta_w W)$ and $h(R_i) \leq \alpha OPT$. If we pack these rectangles one next to each other the total width becomes at most $\varepsilon^{k-2} W \leq \frac{\gamma}{3} W$ which is true for any $k \geq \log_{\frac{1}{\varepsilon}}\left(\frac{3}{\varepsilon^2\gamma}\right)$. □

We say that a rectangle R_i is *cut* by a box B if $R_i \cap B$, $R_i \setminus B$ and $B \setminus R_i$ are non-empty (considering both R_i and B as open regions with an implicit embedding on the plane). We say that a rectangle $R_i \in H$ (resp. $R_i \in T \cup V$) is *nicely cut* by a box B if R_i is cut by B and their intersection is a rectangular region of width $w(R_i)$ (resp. height $h(R_i)$). Intuitively, this means that an edge of B cuts R_i along its longest side (see Figure 3.6b).

Now it remains to pack $L \cup H \cup T \cup V$. The following lemma describes an almost optimal packing of those rectangles.

Lemma 29. *There is an integer $K_B = (\frac{1}{\varepsilon})(\frac{1}{\delta_w})^{O(1)}$ such that, assuming $\mu_h \leq \frac{\varepsilon^2\delta_w}{4K_B}$, there is a partition of the region $B_{OPT'} := [0, W] \times [0, OPT']$ into a set $\mathscr{B}$ of at most K_B boxes and a packing of the rectangles in $L \cup T \cup V \cup H$ such that:*

- *each box has size equal to the size of some $R_i \in L$ (*large box*), or has height at most $\delta_h OPT'$ (*horizontal box*), or has width at most $\delta_w W$ (*vertical box*);*
- *each $R_i \in L$ is contained in a large box of the same size;*

- *each $R_i \in H$ is contained in an horizontal box or is cut by some box. Furthermore, the total area of horizontal cut rectangles is at most $W \cdot \varepsilon^2 OPT'$;*

- *each $R_i \in T \cup V$ is contained in a vertical box or is nicely cut by some vertical box.*

In order to prove this lemma we will adapt the following result from Nadiradze and Wiese [2016].

Lemma 30 (Nadiradze and Wiese [2016])**.** *Suppose that we perform the classification described in Section 3.2.1 using symmetric parameters $\delta_h = \delta_w = \delta$ and $\mu_h = \mu_w = \mu$. Then, there exists a universal integer $K = (1/\varepsilon)(1/\delta)^{O(1)}$ such that for any input instance there is a partition of $[0, W] \times [0, OPT']$ by line segments with integer coordinates yielding a partition into at most K boxes and a packing of the rectangles $L \cup T \cup V \cup H$ such that*

- *each face of the partition is a rectangular box of height at most $\delta OPT'$ or width at most δW, or its size equals the size of rectangle in L,*

- *for each rectangle R in L there is a box containing R and no other rectangle and whose size equals the size of R in both dimensions,*

- *Every rectangle in $T \cup V$ is either contained in a vertical box or nicely cut by a vertical box, and*

- *all rectangles in H that are not nicely cut by some horizontal box have a total area of at most $\varepsilon \cdot OPT' \cdot W$.*

Proof of Lemma 29. We apply Lemma 30, where we use parameters $\frac{\varepsilon^2}{2}$ and set δ to be δ_w. Recall that $\delta_w < \delta_h$; by requiring that $\mu_h < \delta_w$, and since rectangles with height in $[\delta_w, \delta_h)$ are in M, we have that $\{R_i \in \mathscr{R} \setminus M : w(R_i) \geq \delta_h W \text{ and } h(R_i) \geq \delta_h OPT\} = \{R_i \in \mathscr{R} \setminus M : w(R_i) \geq \delta_w W \text{ and } h(R_i) \geq \delta_h OPT\}$.

Let $H_{cut} \subseteq H$ be the set of horizontal rectangles that are nicely cut by a box. Since rectangles in H_{cut} satisfy $w(R) \geq \delta_w W$, at most $\frac{2}{\delta_w}$ of them are nicely cut by a box, and there are at most K_B boxes. Hence, their total area is at most $\frac{\mu_h OPT \cdot W \cdot 2K_B}{\delta_w}$, which is at most $\frac{\varepsilon^2}{2} \cdot OPT \cdot W$, provided that $\mu_h \leq \frac{\varepsilon^2}{4} \cdot \frac{\delta_w}{K_B}$. Since Lemma 30 implies that the area of the cut horizontal rectangles that are not nicely cut is at most $\frac{\varepsilon^2}{2} OPT' \cdot W$, the total area of horizontal cut rectangles is at most $\varepsilon^2 OPT' \cdot W$. □

We denote the sets of vertical, horizontal, and large boxes by $\mathscr{B}_V$, $\mathscr{B}_H$ and $\mathscr{B}_L$, respectively. We next use $T_{cut} \subseteq T$ and $V_{cut} \subseteq V$ to denote tall and vertical cut rectangles in the above lemma, respectively. Let us also define $T_{box} = T \setminus T_{cut}$ and $V_{box} = V \setminus V_{cut}$.

Using standard techniques we can pack all the rectangles excluding the ones contained in vertical boxes in a convenient manner. This is summarized in the following lemma.

Lemma 31. *Given $\mathcal{B}$ as in Lemma 29 and assuming $\mu_w \leq \frac{\gamma\delta_h}{6K_B(1+\varepsilon)}$, there exists a packing of $L \cup H \cup T \cup V$ such that:*

1. *all the rectangles in L are packed in $\mathcal{B}_L$;*
2. *all the rectangles in H are packed in $\mathcal{B}_H$ plus an additional box $B_{H,cut}$ of size $W \times O(\varepsilon)OPT$. Furthermore, this additional box can be decomposed into one horizontal container and one area container;*
3. *all the rectangles in $T_{cut} \cup T_{box} \cup V_{box}$ are packed as in Lemma 29;*
4. *all the rectangles in V_{cut} are packed in an additional vertical container $B_{V,cut}$ of size $(\frac{\gamma}{3}W) \times (\alpha OPT)$.*

Proof. Note that there are at most $1/(\delta_w\delta_h)$ rectangles in L and at most $4K_B$ rectangles in T_{cut}, since at most 2 tall rectangles can be nicely cut by the left (resp. right) side of each box; this is enough to prove points (1) and (3).

Thanks to Lemma 29, the total area of horizontal cut rectangles is at most $\varepsilon^2 OPT' \cdot W$. Let us partition this set of rectangles according to their width, obtaining sets A_1 and A_2 corresponding to the rectangles that have width $w(R) \leq \varepsilon W$ and the rectangles that have width $w(R) > \varepsilon W$ respectively. By Corollary 8, we can remove A_1 from the packing and pack it using NFDH into an additional area container of width W and height εOPT as $\mu_h \leq \varepsilon^2$ and $\varepsilon^2 \leq (1-2\varepsilon)\varepsilon$. Furthermore, as the total height of the rectangles in A_2 is at most εOPT, we can remove them from the packing and place them in an horizontal container of width W and height εOPT. We can put these two containers one on top of the other in box $B_{H,cut}$, proving point (2).

At most $\frac{2(1+\varepsilon)}{\delta_h}$ rectangles in V can be nicely cut by a box; thus, in total there are at most $\frac{2K_B(1+\varepsilon)}{\delta_h}$ nicely cut vertical rectangles. Since the width of each vertical rectangle is at most $\mu_w W$, they can be removed from the packing and placed in $B_{V,cut}$, piled side by side, as long as $\frac{2K_B(1+\varepsilon)}{\delta_h} \cdot \mu_w W \leq \frac{\gamma}{3} W$, which is equivalent to $\mu_w \leq \frac{\gamma\delta_h}{6K_B(1+\varepsilon)}$. This proves point (4). □

Up to this point the structure of this constructed solution and the one constructed by Nadiradze and Wiese [2016] is (essentially) the same. The main difference is the way to pack $T_{box} \cup V_{box}$ where we exploit a refined approach. This is the technical heart of this paper, and it is discussed in the next section.

3.2.3 A Repacking Lemma

We next describe how to pack rectangles in $T_{box} \cup V_{box}$. In order to highlight our contribution, we first describe how the approach developed by Nadiradze and Wiese [2016] works.

It is convenient to assume that all the rectangles in V_{box} are sliced vertically[2]. Let V_{sliced} be such *sliced* rectangles. We will show how to pack all the rectangles in $T_{box} \cup V_{sliced}$ into a constant number of sub-boxes. Using Lemma 17 it is then possible to pack V_{box} into the space occupied by V_{sliced} plus an additional box $B_{V,round}$ of size $(\frac{\gamma}{3}W) \times \alpha OPT$.

We next focus on a specific vertical box $\overline{B}$, say of size $\overline{w} \times \overline{h}$ (see Figure 3.7a). Let $\overline{T}_{cut}$ be the tall rectangles cut by $\overline{B}$. Observe that there are at most 4 such rectangles (2 on the left/right side of $\overline{B}$). The rectangles in $\overline{T}_{cut}$ are packed as in Lemma 31. Let also $\overline{T}$ and $\overline{V}$ be the tall rectangles and sliced vertical rectangles, respectively, originally packed completely inside $\overline{B}$.

They show that it is possible to pack $\overline{T} \cup \overline{V}$ into a constant size set $\overline{\mathscr{S}}$ of sub-boxes contained inside $\overline{B} - \overline{T}_{cut}$, plus an additional box $\overline{D}$ of size $\overline{w} \times (1+\varepsilon-2\alpha)\overline{h}$. Here $\overline{B} - \overline{T}_{cut}$ denotes the region inside $\overline{B}$ not contained in $\overline{T}_{cut}$. In more detail, they start by considering each rectangle $R_i \in \overline{T}$. Since $\alpha \geq \frac{1}{3}$ by assumption, one of the regions above or below R_i cannot contain another tall rectangle in $\overline{T}$, say the first case applies (the other one being symmetric). Then R_i is moved up so that its top side touches the top boundary of $\overline{B}$. The sliced rectangles in $\overline{V}$ that are covered this way are shifted right below R_i (note that there is enough free space by construction). At the end of the process all the rectangles in $\overline{T}$ touch at least one of the top and bottom side of $\overline{B}$ (see Figure 3.7b). Note that no rectangle is discarded up to this point.

Next, the space inside $\overline{B} - (\overline{T} \cup \overline{T}_{cut})$ is partitioned into maximal height unit-width vertical stripes. We call each such stripe a *free rectangle* if both its top and bottom side overlap with the top or bottom side of some rectangle in $\overline{T} \cup \overline{T}_{cut}$, and otherwise a *pseudo rectangle* (see Figure 3.7c). We define the i-th free rectangle to be the free rectangle contained in stripe $[i-1, i] \times [0, \overline{h}]$.

Note that all the free rectangles are contained in a rectangular region of width

[2]For technical reasons, slices have width 1/2 in Nadiradze and Wiese [2016]. For our algorithm, slices of width 1 suffice.

$\overline{w}$ and height at most

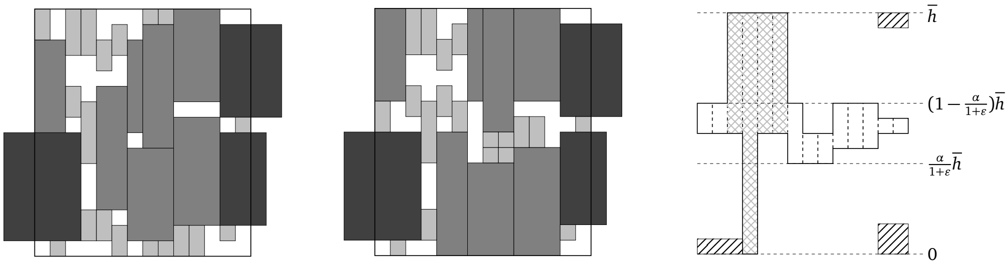
$$\overline{h}-2\alpha OPT \leq \overline{h}-2\alpha\frac{OPT'}{1+\varepsilon} \leq \overline{h}(1-\frac{2\alpha}{1+\varepsilon}) \leq \overline{h}(1+\varepsilon-2\alpha)$$

contained in the central part of $\overline{B}$. Let $\overline{V}_{disc}$ be the set of (sliced vertical) rectangles contained in the free rectangles. Rectangles in $\overline{V}_{disc}$ can be obviously packed inside $\overline{D}$. For each corner Q of the box $\overline{B}$, we consider the maximal rectangular region that has Q as a corner and only contains pseudo rectangles whose top/bottom side overlaps with the bottom/top side of a rectangle in $\overline{T}_{cut}$; there are at most 4 such non-empty regions, and for each of them we define a *corner sub-box*, and we call the set of such sub-boxes $\overline{B}_{corn}$ (see Figure 3.7c). The final step of the algorithm is to rearrange horizontally the pseudo/tall rectangles so that pseudo/tall rectangles of the same height are grouped together *as much as possible* (modulo some technical details). The rectangles in $\overline{B}_{corn}$ are not moved. The *sub-boxes* are induced by maximal consecutive subsets of pseudo/tall rectangles of the same height touching the top (resp., bottom) side of $\overline{B}$ (see Figure 3.7d). We crucially remark that, by construction, the height of each sub-box (and of $\overline{B}$) is a multiple of γOPT.

By splitting each discarded box $\overline{D}$ into two halves $\overline{B}_{disc,top}$ and $\overline{B}_{disc,bot}$, and replicating the packing of boxes inside $B_{OPT'}$, it is possible to pack all the discarded boxes into two boxes $B_{disc,top}$ and $B_{disc,bot}$, both of size $\frac{W}{2} \times (1+\varepsilon-2\alpha)OPT'$.

A feasible packing of boxes (and hence of the associated rectangles) of height $(1+\max\{\alpha, 2(1-2\alpha)\}+O(\varepsilon))OPT$ is then obtained as follows. We first pack $B_{OPT'}$ at the base of the strip, and then on top of it we pack $B_{M,hor}$, two additional boxes $B_{H,round}$ and $B_{H,cut}$ (which will be used to repack the horizontal items; see Lemma 31 for details), and a box B_S (which will be used to pack some of the small items). The latter 4 boxes all have width W and height $O(\varepsilon OPT')$. On the top right of this packing we place $B_{disc,top}$ and $B_{disc,bot}$, one on top of the other. Finally, we pack $B_{M,ver}$, $B_{V,cut}$ and $B_{V,round}$ on the top left, one next to the other. See Figure 3.5a for an illustration. The height is minimized for $\alpha = \frac{2}{5}$, leading to a $(7/5+O(\varepsilon))$-approximation.

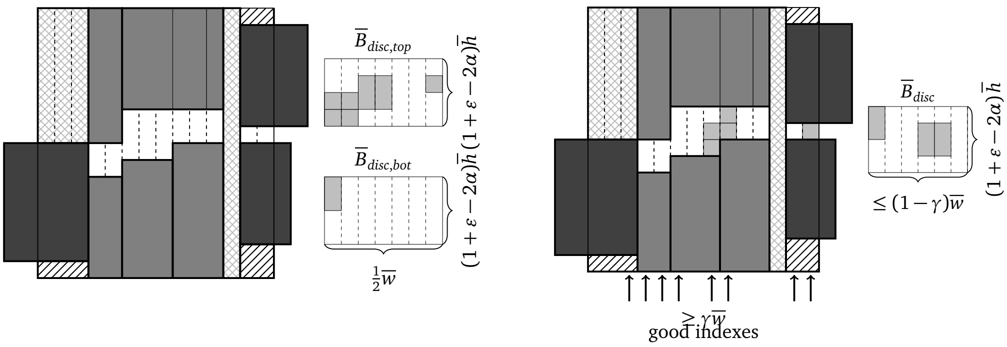
The main technical contribution of this paper is to show how it is possible to repack a subset of $\overline{V}_{disc}$ into the *free* space inside $\overline{B}_{cut} := \overline{B} - \overline{T}_{cut}$ not occupied by sub-boxes, so that the residual sliced rectangles can be packed into a single discarded box $\overline{B}_{disc}$ of size $(1-\gamma)\overline{w} \times (1+\varepsilon-2\alpha)\overline{h}$ (*repacking lemma*). See Figure 3.7e. This apparently minor saving is indeed crucial: with the same approach as above all the discarded sub-boxes $\overline{B}_{disc}$ can be packed into a single *discarded box* B_{disc} of size $(1-\gamma)W \times (1+\varepsilon-2\alpha)OPT'$. Therefore, we can pack all the previous boxes as before, and B_{disc} on the top right. Indeed, the total width of $B_{M,ver}$, $B_{V,cut}$



(a) Original packing in a vertical box $\overline{B}$ after removing V_{cut}. Gray rectangles correspond to $\overline{T}$, dark gray ones to $\overline{T}_{cut}$ and light gray ones to $\overline{V}$.

(b) Rectangles in $\overline{T}$ are shifted vertically so that they touch either the top or the bottom of box $\overline{B}$, shifting also slices in $\overline{V}$ accordingly.

(c) Classification in $\overline{B}-(\overline{T}\cup\overline{T}_{cut})$. Crosshatched stripes correspond to pseudo rectangles, empty stripes to free rectangles, and dashed regions to corner sub-boxes.



(d) Rearrangement of pseudo and tall rectangles to get $O_\varepsilon(1)$ sub-boxes, and additional packing of $\overline{V}_{disc}$ as in Nadiradze and Wiese [2016].

(e) Our refined repacking of $\overline{V}_{disc}$ according to Lemma 32: some vertical slices are repacked in the free space and the extra box has reduced size.

Figure 3.7. Creation of pseudo rectangles, how to get constant number of sub-boxes and repacking of vertical slices in a vertical box $\overline{B}$.

and $B_{V,round}$ is at most γW for a proper choice of the parameters.

Altogether the resulting packing has height $(1+\max\{\alpha, 1-2\alpha\}+O(\varepsilon))OPT$. This is minimized for $\alpha=\frac{1}{3}$, leading to the claimed $(4/3+O(\varepsilon))$-approximation. See Figure 3.5b for an illustration.

It remains to prove our repacking lemma.

Lemma 32 (Repacking Lemma). *Consider a partition of $\overline{D}$ into $\overline{w}$ unit-width vertical stripes. There is a subset of at least $\gamma\overline{w}$ such stripes so that the corresponding sliced vertical rectangles $\overline{V}_{repack}$ can be repacked inside $\overline{B}_{cut}=\overline{B}-\overline{T}_{cut}$ in the space not occupied by sub-boxes.*

Proof. Let $g(i)$ denote the height of the i-th free rectangle, where for notational convenience we introduce a degenerate free rectangle of height $g(i) = 0$ whenever the stripe $[i-1, i] \times [0, \overline{h}]$ inside $\overline{B}$ does not contain any free rectangle. This way we have precisely $\overline{w}$ free rectangles. We remark that free rectangles are defined before the horizontal rearrangement of tall/pseudo rectangles, and the consequent definition of sub-boxes.

Recall that sub-boxes contain tall and pseudo rectangles. Now consider the area in $\overline{B}_{cut}$ not occupied by sub-boxes, which is in fact contained in the central region of height $\overline{h}(1-\frac{2\alpha}{1+\varepsilon})$. Partition this area into maximal-height unit-width vertical stripes as before (*newly free rectangles*). Let $g'(i)$ be the height of the i-th newly free rectangle, where again we let $g'(i) = 0$ if the stripe $[i-1, i] \times [0, \overline{h}]$ does not contain any (positive area) free region. Note that, since tall and pseudo rectangles are only shifted horizontally in the rearrangement, it must be the case that:

$$\sum_{i=1}^{\overline{w}} g(i) = \sum_{i=1}^{\overline{w}} g'(i).$$

Let G be the (*good*) indexes where $g'(i) \geq g(i)$, and $\overline{G} = \{1, \ldots, \overline{w}\} \setminus G$ be the *bad* indexes with $g'(i) < g(i)$. Observe that for each $i \in G$, it is possible to pack the i-th free rectangle inside the i-th newly free rectangle, therefore freeing a unit-width vertical strip inside $\overline{D}$. Thus it is sufficient to show that $|G| \geq \gamma \overline{w}$.

Observe that, for $i \in \overline{G}$, $g(i) - g'(i) \geq \gamma OPT \geq \gamma \frac{\overline{h}}{1+\varepsilon}$: indeed, both $g(i)$ and $g'(i)$ must be multiples of γOPT since they correspond to the height of $\overline{B}$ minus the height of one or two tall/pseudo rectangles. On the other hand, for any index i, $g'(i) - g(i) \leq g'(i) \leq (1-\frac{2\alpha}{1+\varepsilon})\overline{h}$, by the definition of g'. Altogether

$$(1-\frac{2\alpha}{1+\varepsilon})\overline{h}\cdot|G| \geq \sum_{i \in G}(g'(i)-g(i)) = \sum_{i \in \overline{G}}(g(i)-g'(i)) \geq \frac{\gamma\overline{h}}{1+\varepsilon}\cdot|\overline{G}| = \frac{\gamma\overline{h}}{1+\varepsilon}\cdot(\overline{w}-|G|)$$

We conclude that $|G| \geq \frac{\gamma}{1+\varepsilon-2\alpha+\gamma}\overline{w}$, and then the claim follows since by assumption $\alpha > \varepsilon \geq \gamma$. □

3.2.4 A refined Structural Lemma

The original algorithm from Nadiradze and Wiese [2016] uses standard LP-based techniques, as in Kenyon and Rémila [2000], to pack the horizontal rectangles. We can avoid that via a refined structural lemma: here boxes and sub-boxes are further partitioned into containers, and then the resulting container packing can be found optimally in PPT via dynamic programming (see Theorem 16). This approach has two main advantages:

- It leads to a simpler algorithm.
- It can be easily adapted to the case with rotations, as discussed in Section 3.2.6.

The goal of this section is to prove the following lemma that summarizes the aforementioned properties. Recall that $\mathcal{B}_{OPT'} = [0,W] \times [0,OPT']$.

Lemma 33. *By choosing $\alpha = 1/3$, there is an integer $K_F \leq \left(\frac{1}{\varepsilon\delta_w}\right)^{O(1/(\delta_w\varepsilon))}$ such that, assuming $\mu_h \leq \frac{\varepsilon}{K_F}$ and $\mu_w \leq \frac{\gamma}{3K_F}$, there is a packing of $\mathcal{R}$ in the region $[0,W] \times [0,(4/3+O(\varepsilon))OPT']$ with the following properties:*

- *All the rectangles in $\mathcal{R} \setminus S$ are contained in $K_{TOTAL} = O_\varepsilon(1)$ containers, such that each of these containers is either contained in or disjoint from $\mathcal{B}_{OPT'}$;*
- *At most K_F containers are contained in $\mathcal{B}_{OPT'}$, and their total area is at most $a(\mathcal{R} \setminus S)$.*
- *All the rectangles in S are packed into $O(K_F^2)$ area containers inside $\mathcal{B}_{OPT'}$ plus an extra area container B_S disjoint from $\mathcal{B}_{OPT'}$.*

Horizontal rectangles

Thanks to Lemma 31 we know that horizontal rectangles are packed into a set of boxes of cardinality at most $K_B = \left(\frac{1}{\varepsilon}\right)\left(\frac{1}{\delta_w}\right)^{O(1)}$. By requiring that $\mu_h \leq \frac{\varepsilon}{2K_B}\frac{1}{K_B(K_B/\varepsilon^2)^{4K_B/\varepsilon^2}}$ we can refine these boxes into containers by means of Lemma 17. The two extra containers can be packed into a box $B_{H,round}$ of size $W \times O(\varepsilon)OPT'$.

Vertical and tall rectangles

The main goal of this section is to prove the following lemma:

Lemma 34. *There is a constant $K_V \in O_\varepsilon(1)$ such that, assuming $\mu_w \leq \left(\frac{\gamma^5}{6^{2/\gamma}}\right)^{\frac{6^{2/\gamma}}{\gamma^4}}$, it is possible to pack all the rectangles into $T \cup V$ in at most K_V vertical containers, so that each container is packed completely either:*

- *in one of the boxes in $\mathcal{B}_V$;*
- *in the original position of a nicely cut rectangle from Lemma 29 and containing only the corresponding nicely cut rectangle;*
- *in a box B_{disc} of size $(1-\gamma)W \times (1+\varepsilon-2\alpha)OPT'$;*

- *in one of two vertical containers $B_{V,cut}$ and $B_{V,round}$, each of size $\frac{\gamma}{3}W \times \alpha OPT$.*

Moreover, the area of the vertical containers packed inside $B_{OPT'}$ is at most $a(T \cup V)$.

Consider a specific vertical box $\overline{B}$ of size $\overline{w} \times \overline{h}$; as described in Section 3.2.3, the rectangles are repacked so that each rectangle in $\overline{T}$ touches either the top or the bottom edge of $\overline{B}$, and then the set $\overline{P}$ of pseudo rectangles plus the (up to four) corner sub-boxes $\overline{B}_{corn}$ are defined, each one of them containing only slices of rectangles in V. Let $\overline{B}_{rem} := \overline{B} - \overline{T}_{cut}$. We now get a rearrangement of this packing applying the following lemma from Nadiradze and Wiese [2016]:

Lemma 35 (follows from the proof of Lemma 3.6 and Section 4 in Nadiradze and Wiese [2016]). *There is packing of $\overline{T} \cup \overline{P} \cup \overline{B}_{corn}$ into at most $K_R := 2\frac{1+\varepsilon}{\gamma} \cdot 6^{(1+\varepsilon)/\gamma} + 4$ sub-boxes inside $\overline{B}_{rem}$, such that:*

- *each sub-box contains only tall rectangles or only pseudo rectangles, that are all of the same height as the sub-box;*
- *each sub-box is completely occupied by the contained pseudo/tall rectangles, and the y-coordinate of such rectangles is the same as before the rearrangement;*
- *the corner sub-boxes in $\overline{B}_{corn}$ and the rectangle slices inside them are packed in the same position as before the rearrangement.*

Proof. We give a brief outline of the proof, the details can be found in Section 4 in Nadiradze and Wiese [2016].
First, it is possible to combine rectangles in $\overline{B}_{corn}$ with $\overline{T}_{cut}$ to form new unmovable items, which we denote by $\overline{T}'_{cut}$ (Lemma 4.6 in Nadiradze and Wiese [2016]). This way we can assume that the boundary of each item in $\overline{T}'_{cut}$ intersects a corner of $\overline{B}$. Recall that items in $\overline{T} \cup \overline{P}$ touch either the top or the bottom boundary of $\overline{B}$. Now the following result can be proved by induction:
Given a packing into a box $\overline{B}$ such that:

- each item touches the top or the bottom boundary of $\overline{B}$;
- the height of each item equals one out of at most Γ many values;
- the heights of the items touching the bottom boundary have at most k distinct values;
- the items touching the four corners are called unmovable items, all other items are movable items;

then there exists another packing that does not change the positions of the unmovable items and allows a *nice* partition into $6^k \cdot \Gamma$ sub-boxes for the movable items.
In this *nice* partition, the *sub-boxes* are induced by maximal consecutive subsets of movable items of the same height touching the top (resp., bottom) side of $\overline{B}$. In our case, by Lemma 3.2 in Nadiradze and Wiese [2016], we get $k = \frac{1+\varepsilon}{\gamma}$ and $\Gamma = \frac{1+\varepsilon}{\gamma}$. Now each sub-box can be divided into two sub-boxes by rearranging tall/pseudo rectangles inside: one sub-box contains only tall rectangles while the other one contains only pseudo-rectangles. By considering also the corner sub-boxes $\overline{B}_{corn}$ we get the desired value of K_R. Furthermore, each sub-box contains only tall rectangles or only pseudo rectangles that are all of the same height as the sub-box (notice that this holds for corner sub-boxes in $\overline{B}_{corn}$ as well since each one of them contains only pseudo-rectangles of the same height). On the other hand, in this procedure every rectangle is moved only horizontally, implying that the y-coordinate of each pseudo/tall rectangle in $\overline{B}$ remains unchanged after the rearrangement. □

Consider the packing obtained by the above lemma; partition all the free space in $\overline{B}_{rem}$ which is not occupied by the above defined boxes into at most $2K_R + 1$ empty sub-boxes by considering the maximal rectangular regions that are not intersected by the vertical lines passing through the edges of the sub-boxes. By Lemma 32, a fraction of the rectangles contained in slices of $\overline{D}$ of total width at least $\gamma\overline{w}$ can be repacked inside the empty sub-boxes.

Among the at most $3K_R + 1$ sub-boxes that we defined, some only contain tall rectangles, while the others contain pseudo rectangles. The ones that only contain tall rectangles are already containers and box $B_{V,cut}$ defined in the proof of Lemma 31 is already a vertical container as well. For each sub-box B' that contains pseudo rectangles, we now consider the sliced vertical rectangles that are packed in it. By Lemma 17, there is a packing of almost all the rectangles in B' into at most $O_\varepsilon(1)$ containers, and their total area is equal to the total area of the rectangles they contain. There are also at most $4K_B$ containers to pack the tall rectangles that are nicely cut; each of them is packed in its original position in a vertical container of exactly the same size. In total we defined at most $\kappa \in O_\varepsilon(1)$ containers. As the remaining vertical rectangles can be packed in a vertical container $B_{V,round}$ of size $\frac{\gamma W}{3} \times \alpha OPT$, this concludes the proof of Lemma 34.

Small rectangles

It remains to pack the small rectangles S. We can pack them in the free space left by horizontal and vertical containers inside $\mathcal{B}_{OPT'}$ plus an additional box B_S of small height by means of Lemma 18 using the area of the rectangles as profit. By placing box B_S on top of the remaining packed rectangles, the final height of the solution increases only by $\varepsilon \cdot OPT'$.

Concluding the proof

There are at most $K_L := \frac{1}{\delta_h \delta_w}$ many large rectangles. Each such large rectangle is assigned to one horizontal container of the same size.

Rectangles in M are packed as described in the proof of Lemma 28, using at most 3 containers, which are placed in the boxes $B_{M,hor}$ and $B_{M,ver}$.

Horizontal, vertical and small rectangles are packed as explained in the previous subsection. The total number of containers is clearly $O_\varepsilon(1)$, and each of these containers is either contained in or disjoint from $\mathcal{B}_{OPT'}$.

By packing the boxes and containers we defined as in Figure 3.5b, we obtain a packing in a strip of width W and height $OPT' \cdot (\max\{1+\alpha, 1+(1-2\alpha)\}+O(\varepsilon))$, which is at most $(4/3 + O(\varepsilon))OPT'$ for $\alpha = 1/3$. This concludes the proof of Lemma 33.

3.2.5 The Final Algorithm

Consider the packing of Lemma 33: all the rectangles are packed into $K_{TOTAL} = O_\varepsilon(1)$ containers. Since their positions (x, y) and their sizes (w, h) are w.l.o.g. contained in $\{0, \ldots, W\} \times \{0, \ldots, nh_{\max}\}$, we can enumerate in PPT over all the possible feasible such packings of $k \leq K_{TOTAL}$ containers, and one of those will coincide with the packing defined by Lemma 33. The problem of assigning the rectangles to the containers can be solved exactly in PPT as stated in Theorem 16, hence packing all the rectangles.

Note that unlike Nadiradze and Wiese [2016], we do not use linear programming to pack horizontal rectangles, which will be crucial when we extend our approach to the case with rotations.

It is not difficult to see that function f and constant k from Lemma 26 can be chosen in such a way that all the constraints are satisfied. Finally we achieve the claimed result.

Theorem 36. *There is a PPT $(\frac{4}{3} + \varepsilon)$-approximation algorithm for Strip Packing.*

3.2.6 Extension to the Case with Rotations

In this section, we briefly explain the changes needed in the above algorithm to handle the case with rotations.

We first observe that, by considering the rotation of the rectangles as in the optimum solution, Lemma 33 still applies. Therefore we can define a multiple knapsack instance, where knapsack sizes are defined as before. Some extra care is needed to define the size $b(i,j)$ of rectangle R_i into a container C_j of size $w(C_j) \times h(C_j)$. Assume C_j is horizontal, the vertical case being symmetric. If rectangle R_i fits in C_j both rotated and non-rotated, then we set $b(i,j) = \min\{w(R_i), h(R_i)\}$ (this dominates the size occupied in the knapsack by the optimal rotation of R_i). If R_i fits in C_j only non-rotated (resp., rotated), we set $b(i,j) = h(R_i)$ (resp., $b(i,j) = w(R_i)$). Otherwise we set $b(i,j) = +\infty$.

In the case of an area container C_k, we set the size $b(i,k)$ of rectangle R_i to be $a(R_i)$ if $h(R_i) \leq \varepsilon h(C_k)$ and $w(R_i) \leq \varepsilon w(C_k)$, or $w(R_i) \leq \varepsilon h(C_k)$ and $h(R_i) \leq w(C_k)$ (in other words, if there exists an orientation such that the rectangle is small compared to the container in both dimensions). Otherwise we set $b(i,k) = +\infty$. This way, we know that there exists a rotation for the rectangles assigned to an area container such that NFDH can pack all of them, and such rotation can be determined just from the size of the rectangle and the dimensions of the container.

By construction, the above multiple knapsack instance admits a feasible solution that packs all the rectangles. This immediately implies a packing of all the rectangles. Altogether we achieve:

Theorem 37. *There is a PPT $(\frac{4}{3} + \varepsilon)$-approximation algorithm for Strip Packing with rotations.*

3.3 Conclusions

The approximability of Strip Packing in pseudo-polynomial running time has been basically settled recently (Henning et al. [2018]; Jansen and Rau [2019]), but for the case of polynomial running time there is still a gap between the lower and upper bound. This question seems to be hard (either knowing if the lower or the upper bound can be improved) but making progress towards the final answer is an important task in the field. Furthermore, addressing non-trivial special families of instances is also an interesting direction that may help to answer the most general question. As our first result suggests that the presence of large items makes the instances difficult, considering restricted instances that include large items is a reasonable and interesting direction (for example, rectangles with lower-bounded height or lower-bounded width).

A related problem in the field is the *dynamic storage allocation* problem (DSA). Here we are given a strip of integral width W and a set of n rectangles $\{R_1, \ldots, R_n\}$, each one characterized by an integral height $h(R_i)$, an integral starting x-coordinate $s(R_i)$ and an integral ending x-coordinate $t(R_i)$, satisfying $0 \leq s(R_i) < t(R_i) \leq W$, and the goal is to pack all the rectangles inside the strip such that the starting and ending coordinates of each task are respected and the rectangles do not overlap while minimizing the final height of the packing (in other words, it is similar to Strip Packing but the horizontal position of each rectangle is fixed, being the width of each rectangle equal to the distance between its starting and ending coordinates). This problem is known to be NP-hard and the best known approximation factor for it is $(2 + \varepsilon)$ (Buchsbaum et al. [2004]), leaving space to look for improved approximation algorithms.

Chapter 4

On the Two-Dimensional Geometric Knapsack problem

In this chapter we present our results for the two-dimensional Geometric Knapsack problem. Namely, we present improved approximation algorithms for the four most studied variants (with/without rotations, uniform/general profits). These results were published at FOCS 2017 (Gálvez et al. [2017]).

We recall that in the two-dimensional Geometric Knapsack problem (2DK) we are given a set of n elements $\mathscr{R} = \{1, \ldots, n\}$, where each $i \in \mathscr{R}$ is an axis-aligned open rectangle $(0, w(i)) \times (0, h(i))$ in the two-dimensional plane, and has an associated profit $p(i)$[1]. Furthermore, we are given an axis-aligned square knapsack $K = [0, N] \times [0, N]$. All the values $w(i)$, $h(i)$, $p(i)$ and N are positive integers. Our goal is to select a subset of items $OPT \subseteq \mathscr{R}$ of maximum total profit $opt = p(OPT) := \sum_{i \in OPT} p(i)$ and to place them so that the selected items are pairwise disjoint and fully contained in the knapsack. As in Strip Packing, for each $i \in OPT$ we have to define a pair of coordinates $(\mathit{left}(i), \mathit{bottom}(i))$ that specify the position of the bottom-left corner of i in the packing. In other words, i is mapped into a rectangle $R(i) := (\mathit{left}(i), \mathit{right}(i)) \times (\mathit{bottom}(i), \mathit{top}(i))$, with $\mathit{right}(i) = \mathit{left}(i) + w(i)$ and $\mathit{top}(i) = \mathit{bottom}(i) + h(i)$. For any two $i, j \in OPT$, we must have $R(i) \subseteq K$ and $R(i) \cap R(j) = \emptyset$.

2DK is strongly NP-hard (Leung et al. [1990]), and it has been intensively studied from the point of view of approximation algorithms. The best known polynomial time approximation algorithm for it is due to Jansen and Zhang and yields a $(2 + \varepsilon)$-approximation (Jansen and Zhang [2004b]). This is the best known result even in the *cardinality* case (i.e. with all profits being 1).

[1] In order to follow the standard notation for Knapsack problems we refer to rectangles with associated profits as *items*.

However, there are reasons to believe that much better polynomial time approximation ratios are possible: there is a QPTAS under the assumption that $N = n^{\text{poly}(\log n)}$ (Adamaszek and Wiese [2015]), and there are PTASs if the profit of each item equals its area (Bansal et al. [2009]), if the size of the knapsack can be slightly increased (Fishkin, Gerber, Jansen and Solis-Oba [2005]; Jansen and Solis-Oba [2009]), if all items are relatively small (Fishkin, Gerber and Jansen [2005]) and if all input items are squares Jansen and Solis-Oba [2008]; Heydrich and Wiese [2017]. Note that, with no restriction on N, the current best approximation for 2DK is $2+\varepsilon$ *even in quasi-polynomial time*.

All prior polynomial-time approximation algorithms for 2DK implicitly or explicitly exploit a *container-based* packing approach as described in Section 2.2. Indeed, also the QPTAS from Adamaszek and Wiese [2015] can be cast in this framework, with the relevant difference that the number of containers in this case is poly-logarithmic (leading to quasi-polynomial running time).

One of the major bottlenecks to achieve approximation factors better than 2 (in polynomial-time) is that items that are high and narrow (*vertical* items) and items that are wide and thin (*horizontal* items) can interact in a very complicated way. Indeed, consider the following seemingly simple *L-packing* problem: we are given a set of items i with either $w(i) > \frac{N}{2}$ (horizontal items) or $h(i) > \frac{N}{2}$ (vertical items). Our goal is to pack a maximum profit subset of them inside an L-shaped region $L = ([0, N] \times [0, h_L]) \cup ([0, w_L] \times [0, N])$, so that horizontal (resp., vertical) items are packed in the bottom-right (resp., top-left) of L. To the best of our knowledge, the best-known approximation ratio for this problem is $2+\varepsilon$: Remove either all vertical or all horizontal items, and then pack the remaining items by a simple reduction to one-dimensional knapsack (for which an FPTAS is known). It is unclear whether a container-based packing can achieve a better approximation factor, and we conjecture that this is not the case. As we will see, a better understanding of this L-packing problem will play a major role in the design of improved approximation algorithms for 2DK.

4.0.1 Description of the Results

We break the 2-approximation barrier for 2DK. In order to do that, we substantially deviate for the first time from *pure* container-based packings, which are, either implicitly or explicitly, at the hearth of prior work. Namely, we consider *L&C-packings* that combine $O_\varepsilon(1)$ containers *plus* one L-packing of the above type (see Fig.4.1.(a)), and show that one such packing has large enough profit.

While it is easy to pack almost optimally items into containers, the mentioned $(2+\varepsilon)$-approximation for L-packings is not sufficient to achieve altogether a better than 2 approximation factor: indeed, the items of the L-packing might carry all the profit! The main algorithmic contribution of this paper is a PTAS for the L-packing problem. It is easy to solve this problem optimally in pseudo-polynomial time $(Nn)^{O(1)}$ by means of dynamic programming. We show that a $(1+\varepsilon)$-approximation can be obtained by restricting the top (resp., right) coordinates of horizontal (resp., vertical) items to a proper set that can be computed in polynomial time $n^{O_\varepsilon(1)}$. Given that, one can adapt the above dynamic program to run in polynomial time.

Theorem 38. *There is a PTAS for the L-packing problem.*

In order to illustrate the power of our approach, we next sketch a simple $\left(\frac{16}{9}+O(\varepsilon)\right)$-approximation for the cardinality case of 2DK (details in Section 4.4.1). By standard arguments[2] it is possible to discard *large* items with both sides longer than $\varepsilon \cdot N$. The remaining items have height or width smaller than $\varepsilon \cdot N$ (*horizontal* and *vertical* items, resp.). Let us delete all items intersecting a random vertical or horizontal strip of width $\varepsilon \cdot N$ inside the knapsack. We can pack the remaining items into $O_\varepsilon(1)$ containers by exploiting the PTAS under one-dimensional resource augmentation for 2DK (Lemma 19). A vertical strip deletes vertical items with $O(\varepsilon)$ probability, and horizontal ones with probability roughly proportional to their width, and symmetrically for a horizontal strip. In particular, let us call *long* the items with longer side larger than $N/2$, and *short* the remaining items. Then the above argument gives in expectation roughly one half of the profit opt_{long} of long items, and three quarters of the profit opt_{short} of short ones. This is already good enough unless opt_{long} is large compared to opt_{short}.

At this point L-packings and our PTAS come into play. We shift long items such that they form 4 stacks at the sides of the knapsack in a *ring-shaped* region, see Fig.4.1.(b)-(c): this is possible since any vertical long item cannot have an horizontal long item *both* at its left and at its right, and vice versa. Next we delete the least profitable of these stacks and rearrange the remaining long items into an L-packing, see Fig.4.1.(d). Thus using our PTAS for L-packings, we can compute a solution of profit roughly three quarters of opt_{long}. It is not difficult to check that the combination of these two algorithms gives the claimed approximation factor.

Above we used either $O_\varepsilon(1)$ containers or one L-packing: by combining the two approaches together and with a more sophisticated case analysis we achieve the following result (see Appendix C).

[2]There can be at most $O_\varepsilon(1)$ such items in any feasible solution, and if the optimum solution contains only $O_\varepsilon(1)$ items we can solve the problem optimally by brute force.

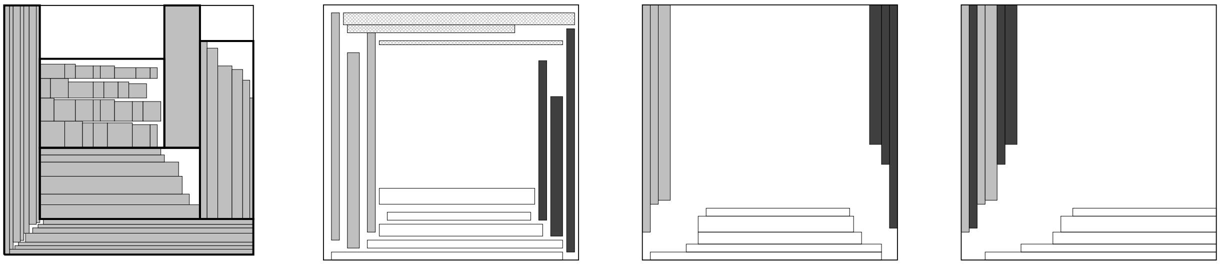


Figure 4.1. (a): An L&C-packing with 4 containers, where the top-left container is packed by means of Next-Fit-Decreasing-Height. (b): A subset of long items. (c): Such items are shifted into 4 stacks at the sides of the knapsack, and the top stack is deleted. (d): The final packing into an L-shaped region.

Theorem 39. *There is a polynomial-time* $\frac{558}{325} + \varepsilon < 1.72$ *approximation algorithm for cardinality 2DK.*

For weighted 2DK we face severe technical complications for proving that there is a profitable L&C-packing. One key reason is that in the weighted case we cannot discard large items since even one such item might contribute a large fraction to the optimal profit. In order to circumvent these difficulties, we exploit the *corridor-partition* at the hearth of the QPTAS for 2DK from Adamaszek and Wiese [2015] (in turn inspired by prior work from Adamaszek and Wiese [2013]). Roughly speaking, there exists a partition of the knapsack into $O_\varepsilon(1)$ *corridors*, consisting of the *concatenation* of $O_\varepsilon(1)$ (partially overlapping) rectangular regions (*subcorridors*). In the QPTAS from Adamaszek and Wiese [2015] the authors partition the corridors into a *poly-logarithmic* number of containers. Their main algorithm then guesses these containers in time $n^{\text{poly}(\log n)}$. However, we can only handle a *constant* number of containers in polynomial time. Therefore, we present a different way to partition the corridors into containers: here we lose the profit of a set of *thin* items, which in some sense play the role of long items in the previous discussion. These thin items fit in a *very narrow* ring at the boundary of the knapsack and we map them to an *L*-packing in the same way as in the cardinality case above. Some of the remaining non-thin items are then packed into $O_\varepsilon(1)$ containers that are placed in the (large) part of the knapsack not occupied by the *L*-packing. Our partition of the corridors is based on a somewhat intricate case analysis that exploits the fact that *long* consecutive subcorridors are arranged in the shape of *rings* or *spirals*: this is used to show the existence of a profitable L&C-packing.

Theorem 40. *There is a polynomial-time* $\frac{17}{9} + \varepsilon < 1.89$ *approximation algorithm for (weighted) 2DK.*

Rotation setting. In the variant of 2DK *with rotations* (2DKR), we are allowed to rotate any item i by 90 degrees. This means that i can also be placed in the knapsack as a rectangle of the form $(left(i), left(i)+h(i)) \times (bottom(i), bottom(i)+w(i))$. The best known polynomial time approximation factor for 2DKR (even for the cardinality case) is again $2+\varepsilon$ due to Jansen and Zhang [2004b] and the mentioned QPTAS from Adamaszek and Wiese [2015] works also for this case.

By using the techniques described above and exploiting a few more ideas, we are also able to improve the approximation factor for 2DKR (see Sections 4.5.1 and 4.5.2 for the cardinality and general case, resp.). The basic idea is that any thin item can now be packed inside a narrow vertical strip (say at the right edge of the knapsack) by possibly rotating it. This way we do not lose one quarter of the profit due to the mapping to an L-packing and instead place all items from the ring into the mentioned strip (while we ensure that their total width is small). The remaining short items are packed by means of a novel *resource contraction* lemma: unless there is one *huge item* that occupies almost the whole knapsack (a case that we consider separately), we can pack almost one half of the profit of non-thin items in a *reduced* knapsack where one of the two sides is shortened by a factor $1-\varepsilon$ (hence leaving enough space for the vertical strip). We remark that here we heavily exploit the possibility to rotate items. Thus, roughly speaking, we obtain either all profit of non-thin items, or all profit of thin items plus one half of the profit of non-thin items: this gives a $\left(\frac{3}{2}+\varepsilon\right)$-approximation.

Theorem 41. *For any constant* $\varepsilon > 0$*, there exists a polynomial-time* $\left(\frac{3}{2}+\varepsilon\right)$ *approximation algorithm for 2DKR.*

A further refinement of this approach yields a $\left(\frac{4}{3}+\varepsilon\right)$-approximation in the cardinality case. We remark that, while resource augmentation is a well-established notion in approximation algorithms, resource contraction seems to be a rather novel direction to explore.

Theorem 42. *For any constant* $\varepsilon > 0$*, there exists a polynomial-time* $\left(\frac{4}{3}+\varepsilon\right)$*-approximation algorithm for cardinality 2DKR.*

4.0.2 Other related work

The mentioned $(2+\varepsilon)$-approximation for two-dimensional Geometric Knapsack (Jansen and Zhang [2004b]) works in the weighted case of the problem. However, in the unweighted case a simpler $(2+\varepsilon)$-approximation is known (Jansen and Zhang [2004a]). If one can increase the size of the knapsack by a factor $1+\varepsilon$ in both dimensions then one can compute a solution of optimal weight, rather than an approximation, in time $f(1/\varepsilon)\cdot n^{O(1)}$ where the exponent of n does not depend on ε for some suitable function f (Heydrich and Wiese [2017]). Similarly, for the case of squares there is a $(1+\varepsilon)$-approximation algorithm known with such a running time, i.e., an EPTAS (Heydrich and Wiese [2017]). This improves previous results such as a $(5/4+\varepsilon)$-approximation (Harren [2006]) and the previously mentioned PTAS (Jansen and Solis-Oba [2008]).

4.1 Preliminaries

In this section we will review some building blocks used later in the analysis of our algorithm.

4.1.1 Item classification

We start with a classification of the input items according to their heights and widths. For two given constants $1 \geq \varepsilon_{large} > \varepsilon_{small} > 0$, we classify an item i as:

- *small* if $h(i), w(i) \leq \varepsilon_{small}N$;
- *large* if $h(i), w(i) > \varepsilon_{large}N$;
- *horizontal* if $w(i) > \varepsilon_{large}N$ and $h(i) \leq \varepsilon_{small}N$;
- *vertical* if $h(i) > \varepsilon_{large}N$ and $w(i) \leq \varepsilon_{small}N$;
- *intermediate* otherwise (i.e., at least one side has length in $(\varepsilon_{small}N, \varepsilon_{large}N]$).

We also call *skewed* items that are horizontal or vertical. We let $\mathcal{R}_{small}$, $\mathcal{R}_{large}$, $\mathcal{R}_{hor}$, $\mathcal{R}_{ver}$, $\mathcal{R}_{skew}$, and $\mathcal{R}_{int}$ be the items which are small, large, horizontal, vertical, skewed, and intermediate, respectively. The corresponding intersection with OPT defines the sets OPT_{small}, OPT_{large}, OPT_{hor}, OPT_{ver}, OPT_{skew} and OPT_{int} respectively.

Notice that $|OPT_{large}| \leq 1/\varepsilon_{large}^2$. Analogously to Lemma 26 in Chapter 3, the next lemma shows that we can neglect OPT_{int}.

Lemma 43. *For any constant $\varepsilon > 0$ and positive increasing function $f(\cdot)$, $f(x) > x$, there exist constant values $\varepsilon_{large}, \varepsilon_{small}$, with $\varepsilon \geq \varepsilon_{large} \geq f(\varepsilon_{small}) \geq \Omega_\varepsilon(1)$ and*

$\varepsilon_{small} \in \Omega_\varepsilon(1)$ *such that the total profit of intermediate items is bounded by* $\varepsilon p(OPT)$*. The pair* $(\varepsilon_{large}, \varepsilon_{small})$ *is one pair from a set of* $O_\varepsilon(1)$ *pairs and this set can be computed in polynomial time.*

Proof. Define $k+1 = 2/\varepsilon + 1$ constants $\varepsilon_1, \ldots, \varepsilon_{k+1}$, such that $\varepsilon = f(\varepsilon_1)$ and $\varepsilon_i = f(\varepsilon_{i+1})$ for each i. Consider the k ranges of widths and heights of type $(\varepsilon_{i+1}N, \varepsilon_i N]$. By an averaging argument there exists one index j such that the total profit of items in *OPT* with at least one side length in the range $(\varepsilon_{j+1}N, \varepsilon_j N]$ is at most $2\frac{\varepsilon}{2}p(OPT)$. It is then sufficient to set $\varepsilon_{large} = \varepsilon_j$ and $\varepsilon_{small} = \varepsilon_{j+1}$. □

4.1.2 Corridors, Spirals and Rings

With the goal of decomposing the optimal solution into simpler substructures, we build on a partition of the knapsack into corridors as in the work of Adamaszek and Wiese [2015]. We define an *open corridor* to be a face on the 2D-plane bounded by a simple rectilinear polygon with $2k$ edges $e_0, \ldots, e_{2k-1}$ for some integer $k \geq 2$, such that for each pair of horizontal (resp., vertical) edges e_i, e_{2k-i}, $i \in \{1, ..., k-1\}$ there exists a vertical (resp., horizontal) line segment ℓ_i such that both e_i and e_{2k-i} intersect ℓ_i and ℓ_i does not intersect any other edge. Note that e_0 and e_k are not required to satisfy this property: we call them the *boundary edges* of the corridor. Similarly a *closed corridor* (or *cycle*) is a face on the 2D-plane bounded by two simple rectilinear polygons defined by edges $e_0, \ldots, e_{k-1}$ and $e'_0, \ldots, e'_{k-1}$ such that the second polygon is contained inside the first one, and for each pair of horizontal (resp., vertical) edges e_i, e'_i, $i \in \{0, ..., k-1\}$, there exists a vertical (resp., horizontal) line segment ℓ_i such that both e_i and e'_i intersect ℓ_i and ℓ_i does not intersect any other edge (see Figure 4.2 for examples). Let us focus on minimum length such ℓ_i's: then the *width* α of the corridor is the maximum length of any such ℓ_i. We say that an open (resp., closed) corridor of the above kind has $k-2$ (resp., k) *bends*. A corridor decomposition is a partition of the knapsack into corridors.

The following lemma summarizes the properties of the corridor decomposition for OPT_{skew} presented by Adamaszek and Wiese [2015]. Later we will explain how to deal with OPT_{small} and OPT_{large} in each specific case.

Lemma 44 (Corridor Packing Lemma from Adamaszek and Wiese [2015]). *There exists a corridor partition and a set* $OPT_{corr} \subseteq OPT_{skew}$ *such that:*

1. there is a subset $OPT_{corr}^{cross} \subseteq OPT_{corr}$ with $|OPT_{corr}^{cross}| \leq O_\varepsilon(1)$ such that each item $i \in OPT_{corr} \setminus OPT_{corr}^{cross}$ is fully contained in some corridor,
2. $p(OPT_{corr}) \geq (1 - O(\varepsilon))p(OPT_{skew})$,

3. the number of corridors is $O_{\varepsilon,\varepsilon_{large}}(1)$ and each corridor has width at most $\varepsilon_{large}N$ and has at most $1/\varepsilon$ bends.

We next identify some structural properties of the corridors that are later exploited in our analysis. Observe that an open (resp., closed) corridor of the above type is the union of $k-1$ (resp., k) boxes, that we next call *subcorridors* (see also Figure 4.2). Each such subcorridor is a maximally large rectangular region contained in the corridor. The subcorridor S_i of an open (resp., closed) corridor of the above kind is the one containing edges e_i, e_{2k-i} (resp., $e_i, e_{i'}$) on its boundary. The length of S_i is the *length* of the shortest such edge. We say that a subcorridor is *long* if its length is more than $N/2$, and *short* otherwise. The partition of subcorridors into short and long will be crucial in our analysis.

We call a subcorridor *horizontal* (resp., *vertical*) if the corresponding edges are so. Note that each item in OPT_{corr} is univocally associated with the only subcorridor that fully contains it: indeed, the longer side of a skewed item is longer than the width of any corridor. Consider the sequence of consecutive subcorridors $S_1, \dots, S_{k'}$ of an open or closed corridor. Consider two consecutive corridors S_i and $S_{i'}$, with $i' = i+1$ in the case of an open corridor and $i' = (i+1) \pmod{k'}$ otherwise. First assume that $S_{i'}$ is horizontal. We say that $S_{i'}$ is to the right (resp., left) of S_i if the right-most (left-most) boundary of $S_{i'}$ is to the right (left) of the right-most (left-most) boundary of S_i. If instead $S_{i'}$ is vertical, then S_i must be horizontal and we say that $S_{i'}$ is to the right (left) of S_i if S_i is to the left (right) of $S_{i'}$. Similarly, if $S_{i'}$ is vertical, we say that $S_{i'}$ is above (below) S_i if the top (bottom) boundary of $S_{i'}$ is above (below) the top (bottom) boundary of S_i. If $S_{i'}$ is horizontal, we say that it is above (below) S_i if S_i (which is vertical) is below (above) $S_{i'}$. We say that the pair $(S_i, S_{i'})$ forms a clockwise bend if S_i is horizontal and $S_{i'}$ is to its bottom-right or top-left, and the complementary cases if S_i is vertical. In all the other cases the pairs form a counter-clockwise bend. Consider a triple $(S_i, S_{i'}, S_{i''})$ of consecutive subcorridors in the above sense. It forms a U-bend if $(S_i, S_{i'})$ and $(S_{i'}, S_{i''})$ are both clockwise or counterclockwise bends. Otherwise it forms a Z-bend. In both cases $S_{i'}$ is the *center* of the bend, and $S_i, S_{i''}$ its *sides*. An open corridor whose bends are all clockwise (resp., counter-clockwise) is a *spiral*. A closed corridor with $k = 4$ is a *ring*. Note that in a ring all bends are clockwise or counter-clockwise, hence in some sense it is the closed analogue of a spiral. We remark that a corridor whose subcorridors are all long is a spiral or a ring as Z-bends cannot appear. As we will see, spirals and rings play a crucial role in our analysis. In particular, we will exploit the following simple facts.

Lemma 45. *The following properties hold:*

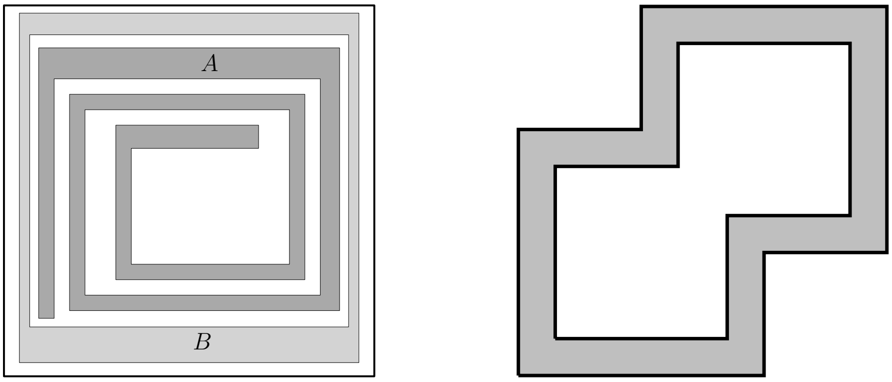


Figure 4.2. Left: Illustration of two specific types of corridors: spirals (A) and rings (B). Right: Example of a closed corridor which is not a ring.

1. *The two sides of a Z-bend cannot be long. In particular, an open corridor whose subcorridors are all long is a spiral.*

2. *A closed corridor contains at least* 4 *distinct (possibly overlapping) U-bends.*

Proof. (1) By definition of long subcorridors and Z-bend, the 3 subcorridors of the Z-bend would otherwise have total width or height larger than N. (2) Consider the left-most and right-most vertical subcorridords, and the top-most and bottom-most horizontal subcorridors. These 4 subcorridors exist, are distinct, and are centers of a U-bend. □

Finally, the following definition characterizes more in detail the interaction between neighboring subcorridors. Given two consecutive subcorridors S_i and $S_{i'}$, we define the *boundary curve* among them as follows (see also Figure 4.2). Suppose that $S_{i'}$ is to the top-right of S_i, the other cases being symmetric. Let $S_{i,i'} = S_i \cap S_{i'}$ be the rectangular region shared by the two subcorridors. Then the boundary curve among them is any simple rectilinear polygon inside $S_{i,i'}$ that decreases monotonically from its top-left corner to its bottom-right one and that does not cut any item in these subcorridors. For a boundary horizontal (resp., vertical) subcorridor of an open corridor (i.e., a subcorridor containing e_0 or e_{2k-1}) we define a dummy boundary curve given by the vertical (resp., horizontal) side of the subcorridor that coincides with a (boundary) edge of the corridor.

Remark 46. *Each subcorridor has two boundary curves (including possibly dummy ones). Furthermore, all its items are fully contained in the region delimited by such curves plus the two edges of the corridor associated with the subcorridor (private region).*

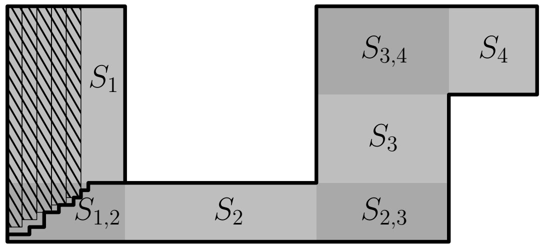


Figure 4.3. The subcorridors S_1 and S_3 are vertical, S_2 and S_4 are horizontal. The subcorridor S_3 is on the top-right of S_2. The curve on the bottom left shows the boundary curve between S_1 and S_2. The pair (S_3, S_4) forms a clockwise bend and the pair (S_2, S_3) forms a counter-clockwise bend. The triple (S_1, S_2, S_3) forms a U-bend and the triple (S_2, S_3, S_4) forms a Z-bend.

Later in Section 4.4.2 we will use this concept to further simplify the given corridor partition into regions that can be searched efficiently.

4.2 Our approach: Search for L&C Packings

We proceed now to describe in detail the kind of structured solutions we will search for and how to compute them efficiently. Instead of searching for purely container based solutions as in prior work, we consider solutions that combine $O_\varepsilon(1)$ containers plus one L-packing which we proceed to define. We will refer to this kind of solutions as *L&C-packings*.

We will require the following technical definitions. A *boundary ring* of width N' is a ring having as external boundary the edges of the knapsack and as internal boundary the boundary of a square box of size $(N-N') \times (N-N')$ in the middle of the knapsack. A *boundary L* of width N' is the region covered by two boxes of size $N' \times N$ and $N \times N'$ that are placed on the left and bottom boundaries of the knapsack.

L-packing problem. In this problem we are given a set of *horizontal* items $\mathscr{R}_{hor}$ having width larger than $N/2$, and a set of *vertical* items $\mathscr{R}_{ver}$ with height larger than $N/2$. Furthermore, we are given an L-shaped region $L = ([0,N] \times [0,h_L]) \cup ([0,w_L] \times [0,N])$. Our goal is to pack a subset $OPT \subseteq \mathscr{R} := \mathscr{R}_{hor} \cup \mathscr{R}_{ver}$ of maximum total profit such that $OPT_{hor} := OPT \cap \mathscr{R}_{hor}$ is packed inside the *horizontal box* $[0,N] \times [0,h_L]$ and $OPT_{ver} := OPT \cap \mathscr{R}_{ver}$ is packed inside the *vertical box* $[0,w_L] \times [0,N]$. A solution to this problem we call an L-packing.

We remark that packing horizontal and vertical items independently is not suitable due to the possible overlaps in the intersection of the two boxes: this is what makes this problem non-trivial, in particular harder than standard (one-dimensional) knapsack.

L&C packings. An *L&C* packing is then defined as follows. We are given two integer parameters $N' \in [0, N/2]$ and $\ell \in (N/2, N]$. We define a boundary L-shaped region of width N', and a collection of non-overlapping containers contained in the space not occupied by the boundary L. We let $\mathcal{R}_{long} \subseteq \mathcal{R}$ be the items whose longer side has length longer than ℓ (hence longer than $N/2$), and $\mathcal{R}_{short} = \mathcal{R} \setminus \mathcal{R}_{long}$ be the remaining items. We can pack only items from $\mathcal{R}_{long}$ in the boundary L, and only items from $\mathcal{R}_{short}$ in the containers (satisfying the usual container packing constraints). See also Figure 4.1.

Remark 47. *L&C packings contain container packings as a special case if we set $N' = 0$ and $\ell = N$. This we call a degenerate L case.*

4.2.1 Main algorithm

Now we can present the main algorithm for all the considered variants of 2DK. It is in fact an approximation scheme for L&C packings. In order to bound the approximation factor that this algorithm achieves in each variant we need to understand how good with respect to the optimal value the most profitable L&C packing is, which will be discussed in detail in Sections 4.4 and 4.5.

Our algorithm combines two basic packing procedures. The first one is the PTAS to pack items into a constant number of containers (Theorem 16) described in Chapter 2. The second packing procedure has to deal with our L-packing problem. As we will prove in Section 4.3, there is a PTAS for this problem (see Theorem 38).

To use these packing procedures, we first guess whether the optimal L&C-packing uses a non-degenerate boundary L. If yes, we guess a parameter ℓ which denotes the minimum height of the vertical items in the boundary L and the minimum width of the horizontal items in the boundary L. For ℓ we allow all heights and widths of the input items that are larger than $N/2$, i.e., at most $2n$ values. Let $\mathcal{R}_{long}$ be the items whose longer side has length at least ℓ (hence longer than $N/2$). We guess the width of the boundary L from the candidate values (which we will prove can be done efficiently) and solve the resulting instance $(L, \mathcal{R}_{long})$ almost optimally using the PTAS for L-packings due to Theorem 38.

Then we enumerate all the possible subsets of non-overlapping containers in the space not occupied by the boundary L (or in the full knapsack, in the case of a degenerate L), where we will prove that only $O_\varepsilon(1)$ containers and a set of size $n^{O_\varepsilon(1)}$ that we can compute in polynomial time such that the height and the width of each container is contained in this set suffice. We compute an approximate solution for the resulting container packing instance with items $\mathcal{R}_{short} = \mathcal{R} \setminus \mathcal{R}_{long}$ using the PTAS from Theorem 16. Finally, we output the most profitable solution that we computed.

4.3 A PTAS for L-packings

In this section we present our main algorithmic tool, which is a PTAS for the problem of finding an optimal L-packing. As mentioned before, the interaction between vertical and horizontal items in the L-shaped region makes this problem interesting and non-trivial. We will borrow notation described in Section 4.2

Observe that in an optimal packing we can assume w.l.o.g. that items in OPT_{hor} are pushed as far to the right/bottom as possible. Furthermore, the items in OPT_{hor} are packed from bottom to top in non-increasing order of width. Indeed, it is possible to *permute* any two items violating this property while keeping the packing feasible. A symmetric claim holds for OPT_{ver}. See Figure 4.1.(d) for an illustration.

Given the above structure, it is relatively easy to define a dynamic program (DP) that computes an optimal L-packing in pseudo-polynomial time $(Nn)^{O(1)}$. The basic idea is to scan items of $\mathcal{R}_{hor}$ (resp. $\mathcal{R}_{ver}$) in decreasing order of width (resp., height), and each time *guess* if they are part of the optimal solution OPT. At each step either both the considered horizontal item i and vertical item j are not part of the optimal solution, or there exists a *guillotine cut*[3] separating i or j from the rest of OPT. Depending on the cases, one can define a smaller L-packing sub-instance (among N^2 choices) for which the DP table already contains a solution.

In order to achieve a $(1+\varepsilon)$-approximation in polynomial time $n^{O_\varepsilon(1)}$, we show that it is possible (with a small loss in the profit) to restrict the possible top coordinates of OPT_{hor} and right coordinates of OPT_{ver} to proper polynomial-size subsets $\mathcal{C}_{top}$ and $\mathcal{C}_{bottom}$, resp. We call such an L-packing $(\mathcal{C}_{top}, \mathcal{C}_{bottom})$-*restricted*. By adapting the above DP one obtains:

Lemma 48. *An optimal $(\mathcal{C}_{top}, \mathcal{C}_{bottom})$-restricted L-packing can be computed in time polynomial in $m := n + |\mathcal{C}_{top}| + |\mathcal{C}_{bottom}|$ using dynamic programming.*

[3] A guillotine cut is an infinite, axis-parallel line ℓ that partitions the items in a given packing in two subsets without intersecting any item.

Proof. For notational convenience we assume $0 \in \mathscr{C}_{top}$ and $0 \in \mathscr{C}_{bottom}$. Let $H_1, \ldots, H_{|\mathscr{R}_{hor}|}$ be the items in $\mathscr{R}_{hor}$ in decreasing order of width and $V_1, \ldots, V_{|\mathscr{R}_{ver}|}$ be the items in $\mathscr{R}_{ver}$ in decreasing order of height (breaking ties arbitrarily). For $w \in [0, w_L]$ and $h \in [0, h_L]$, let $L(w,h) = ([0,w] \times [0,N]) \cup ([0,N] \times [0,h]) \subseteq L$. Let also $\Delta L(w,h) = ([w,w_L] \times [h,N]) \cup ([w,N] \times [h,h_L]) \subseteq L$. Note that $L = L(w,h) \cup \Delta L(w,h)$.

We define a dynamic program table DP indexed by $i \in [1, |\mathscr{R}_{hor}|]$ and $j \in [1, |\mathscr{R}_{ver}|]$, by a top coordinate $t \in \mathscr{C}_{top}$, and a right coordinate $r \in \mathscr{C}_{bottom}$. The value of $\mathrm{DP}(i,t,j,r)$ is the maximum profit of a $(\mathscr{C}_{top}, \mathscr{C}_{bottom})$-restricted packing of a subset of $\{H_i, \ldots, H_{|\mathscr{R}_{hor}|}\} \cup \{V_j, \ldots, V_{|\mathscr{R}_{ver}|}\}$ inside $\Delta L(r,t)$. The value of $\mathrm{DP}(1,0,1,0)$ is the value of the optimum solution we are searching for. Note that the number of table entries is upper bounded by m^4.

We fill in DP according to the partial order induced by vectors (i,t,j,r), processing larger vectors first. The base cases are given by $(i,j) = (n(h)+1, n(v)+1)$ and $(r,t) = (w_L, h_L)$, in which case the table entry has value 0.

In order to compute any other table entry $\mathrm{DP}(i,t,j,r)$, with optimal solution OPT', we take the maximum of the following few values:

- If $i \leq |\mathscr{R}_{hor}|$, the value $\mathrm{DP}(i+1,t,j,r)$. This covers the case that $H_i \notin OPT'$;
- If $j \leq |\mathscr{R}_{ver}|$, the value $\mathrm{DP}(i,t,j+1,r)$. This covers the case that $V_j \notin OPT'$;
- Assume that there exists $t' \in \mathscr{C}_{top}$ such that $t' - h(H_i) \geq t$ and that $w(H_i) \leq N - r$. Then for the minimum such t' we consider the value $p(H_i) + \mathrm{DP}(i+1, t', j, r)$. This covers the case that $H_i \in OPT'$, and there exists a (horizontal) guillotine cut separating H_i from $OPT' \setminus \{H_i\}$.
- Assume that there exists $r' \in \mathscr{C}_{bottom}$ such that $r' - w(V_j) \geq r$ and that $h(V_j) \leq N - t$. Then for the minimum such r' we consider the value $p(V_j) + \mathrm{DP}(i,t,j+1,r')$. This covers the case that $V_j \in OPT'$, and there exists a (vertical) guillotine cut separating V_j from $OPT' \setminus \{v_j\}$.

We observe that the above cases (which can be explored in polynomial time) cover all the possible configurations in OPT'. Indeed, if the first two cases do not apply, we have that $H_i, V_j \in OPT'$. Then either the line containing the right side of V_j does not intersect H_i (hence any other item in OPT') or the line containing the top side of H_i does not intersect V_j (hence any other item in OPT'). Indeed, the only remaining case is that V_j and H_i overlap, which is impossible since they both belong to OPT'. □

We will show that there exists a $(\mathscr{C}_{top}, \mathscr{C}_{bottom})$-restricted L-packing with the desired properties.

Lemma 49. *There exists a* $(\mathscr{C}_{top}, \mathscr{C}_{bottom})$*-restricted L-packing solution of profit at least* $(1-2\varepsilon)opt$*, where the sets* $\mathscr{C}_{top}$ *and* $\mathscr{C}_{bottom}$ *have cardinality at most* $n^{O(1/\varepsilon^{1/\varepsilon})}$ *and can be computed in polynomial time based on the input (without knowing OPT).*

Lemmas 48 and 49 together immediately imply a PTAS for L-packings (showing Theorem 38). The rest of this section is devoted to the proof of Lemma 49.

We will describe a way to delete a subset of items $D_{hor} \subseteq OPT_{hor}$ with $p(D_{hor}) \leq 2\varepsilon p(OPT_{hor})$, and *shift down* the remaining items $OPT_{hor} \setminus D_{hor}$ so that their top coordinate belongs to a set $\mathscr{C}_{top}$ with the desired properties. Symmetrically, we will delete a subset of items $D_{ver} \subseteq OPT_{ver}$ with $p(D_{ver}) \leq 2\varepsilon p(OPT_{ver})$, and *shift to the left* the remaining items $OPT_{ver} \setminus D_{ver}$ so that their right coordinate belongs to a set $\mathscr{C}_{bottom}$ with the desired properties. We remark that shifting down (resp. to the left) items of OPT_{hor} (resp., OPT_{ver}) cannot create any overlap with items of OPT_{ver} (resp., OPT_{hor}). This allows us to reason on each such set separately.

We next focus on OPT_{hor} only: the construction for OPT_{ver} is symmetric. For notational convenience we let $1, \ldots, n_{hor}$ be the items of OPT_{hor} in non-increasing order of width *and* from bottom to top in the starting optimal packing. We remark that this sequence is not necessarily sorted (increasingly or decreasingly) in terms of item heights: this makes our construction much more complicated.

Let us first introduce some useful notation. Consider any subsequence $B = \{b_{start}, \ldots, b_{end}\}$ of consecutive items (*interval*). For any $i \in B$, we define $top_B(i) := \sum_{k \in B, k \leq i} h(k)$ and $bottom_B(i) = top_B(i) - h(i)$. The *growing subsequence* $G = G(B) = \{g_1, \ldots, g_h\}$ of B (with possibly non-contiguous items) is defined as follows. We initially set $g_1 = b_{start}$. Given the item g_i, g_{i+1} is the smallest-index (i.e., lowest) item in $\{g_i + 1, \ldots, b_{end}\}$ such that $h(g_{i+1}) \geq h(g_i)$. We halt the construction of G when we cannot find a proper g_{i+1}. For notational convenience, define $g_{h+1} = b_{end} + 1$. We let $B_i^G := \{g_i + 1, \ldots, g_{i+1} - 1\}$ for $i = 1, \ldots, h$. Observe that the sets B_i^G partition $B \setminus G$. We will crucially exploit the following simple property.

Lemma 50. *For any* $g_i \in G$ *and any* $j \in \{b_{start}, \ldots, g_{i+1} - 1\}$*,* $h(j) \leq h(g_i)$*.*

Proof. The items $j \in B_i^G = \{g_i + 1, \ldots, g_{i+1} - 1\}$ have $h(j) < h(g_i)$. Indeed, any such j with $h(j) \geq h(g_i)$ would have been added to G, a contradiction.

Consider next any $j \in \{b_{start}, \ldots g_i - 1\}$. If $j \in G$ the claim is trivially true by construction of G. Otherwise, one has $j \in B_k^G$ for some $g_k \in G$, $g_k < g_i$. Hence, by the previous argument and by construction of G, $h(j) < h(g_k) \leq h(g_i)$. □

The intuition behind our construction is as follows. Consider the growing sequence $G = G(OPT_{hor})$, and suppose that $p(G) \leq \varepsilon \cdot p(OPT_{hor})$. Then we might simply delete G, and shift the remaining items $OPT_{hor} \setminus G = \cup_j B_j^G$ as follows. Let $\lceil x \rceil_y$ denote the smallest multiple of y not smaller than x. We consider each set B_j^G separately. For each such set, we define a baseline vertical coordinate $base_j = \lceil bottom(g_j) \rceil_{h(g_j)/2}$, where $bottom(g_j)$ is the bottom coordinate of g_j in the original packing. We next round up the height of $i \in B_j^G$ to $\hat{h}(i) := \lceil h(i) \rceil_{h(g_j)/(2n)}$, and pack the rounded items of B_j^G as low as possible above the baseline. It is not difficult to check that the possible top coordinates for rounded items fall in a polynomial size set (using Lemma 50). It is also not hard to check that items are *not* shifted up.

We use recursion in order to handle the case $p(G) > \varepsilon \cdot p(OPT_{hor})$. Rather than deleting G, we consider each B_j^G and build a new growing subsequence for each such set. We repeat the process recursively for r_{hor} many rounds. Let $\mathcal{G}^r$ be the union of all the growing subsequences in the recursive calls of level r. Since the sets $\mathcal{G}^r$ are disjoint by construction, there must exist a value $r_{hor} \leq \frac{1}{\varepsilon}$ such that $p(\mathcal{G}^{r_{hor}}) \leq \varepsilon \cdot p(OPT_{hor})$. Therefore we can apply the same shifting argument to all growing subsequences of level r_{hor} (in particular we delete all of them). In the remaining growing subsequences we can afford to delete 1 out of $1/\varepsilon$ consecutive items (with a small loss of the profit), and then apply a similar shifting argument.

We next describe our approach in more detail. We exploit a recursive procedure `delete&shift`. This procedure takes as input two parameters: an interval $B = \{b_{start}, \ldots, b_{end}\}$, and an integer *round parameter* $r \geq 1$. Procedure `delete&shift` returns a set $D(B) \subseteq B$ of deleted items, and a shift function $shift : B \setminus D(B) \to \mathbb{N}$. Intuitively, $shift(i)$ is the value of the top coordinate of i in the shifted packing w.r.t. a proper baseline value which is implicitly defined. We initially call `delete&shift`(OPT_{hor}, r_{hor}), for a proper $r_{hor} \in \{1, \ldots, \frac{1}{\varepsilon}\}$ to be fixed later. Let $(D, shift)$ be the output of this call. The desired set of deleted items is $D_{hor} = D$, and in the final packing $top(i) = shift(i)$ for any $i \in OPT_{hor} \setminus D_{hor}$ (the right coordinate of any such i is N).

The procedure behaves differently in the cases $r = 1$ and $r > 1$. If $r = 1$, we compute the growing sequence $G = G(B) = \{g_1 = b_{start}, \ldots, g_h\}$, and set $D(B) = G(B)$. Consider any set $B_j^G = \{g_j + 1, \ldots, g_{j+1} - 1\}$, $j = 1, \ldots, h$. Let $base_j := \lceil bottom_B(g_j) \rceil_{h(g_j)/2}$. We define for any $i \in B_j^G$,

$$shift(i) = base_j + \sum_{k \in B_j^G, k \leq i} \lceil h(k) \rceil_{h(g_j)/(2n)}.$$

Observe that $shift$ is fully defined since $\cup_{j=1}^h B_j^G = B \setminus D(B)$.

If instead $r > 1$, we compute the growing sequence $G = G(B) = \{g_1 =$

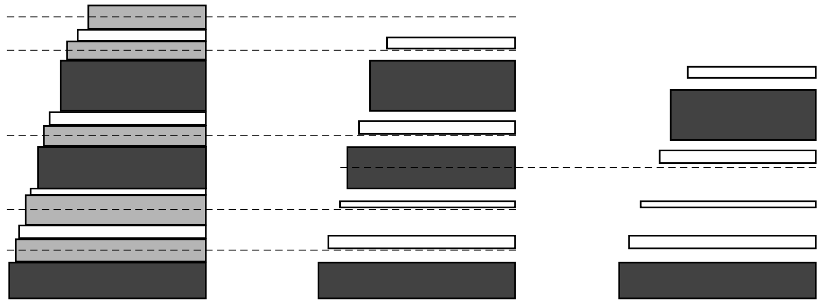


Figure 4.4. Illustration of the `delete&shift` procedure with $r_{hor} = 2$. The dashed lines indicate the value of the new baselines in the different stages of the algorithm. (Left) The starting packing. Dark and light gray items denote the growing sequences for the calls with $r = 2$ and $r = 1$, resp. (Middle) The shift of items at the end of the recursive calls with $r = 1$. Note that light gray items are all deleted, and dark gray items are not shifted. (Right) The shift of items at the end of the process. Here we assume that the middle dark gray item is deleted.

$b_{start}, \ldots, g_h\}$. We next delete a subset of items $D' \subseteq G$. If $h < \frac{1}{\varepsilon}$, we let $D' = D'(B) = \emptyset$. Otherwise, let $G_k = \{g_j \in G : j = k \pmod{1/\varepsilon}\} \subseteq G$, for $k \in \{0, \ldots, 1/\varepsilon - 1\}$. We set $D' = D'(B) = \{d_1, \ldots, d_p\} = G_x$ where $x = \arg\min_{k \in \{0,\ldots,1/\varepsilon-1\}} p(G_k)$. See Figure 4.4 for a sketch of the procedure.

Proposition 51. *It holds that $p(D') \leq \varepsilon \cdot p(G)$. Furthermore, any subsequence $\{g_x, g_{x+1}, \ldots, g_y\}$ of G with at least $1/\varepsilon$ items contains at least one item from D'.*

Let us consider each set $B_j^G = \{g_j + 1, \ldots, g_{j+1} - 1\}$, $j = 1, \ldots, h$: We run `delete&shift`$(B_j^G, r-1)$. Let $(D_j, shift_j)$ be the output of the latter procedure, and $shift_j^{\max}$ be the maximum value of $shift_j$. We set the output set of deleted items to $D(B) = D' \cup (\cup_{j=1}^{h} D_j)$.

It remains to define the function *shift*. Consider any set B_j^G, and let d_q be the deleted item in D' with largest index (hence in topmost position) in $\{b_{start}, \ldots, g_j\}$: define $base_q = \lceil bottom_B(d_q) \rceil_{h(d_q)/2}$. If there is no such d_q, we let $d_q = 0$ and $base_q = 0$. For any $i \in B_j^G$ we set:

$$\begin{aligned} shift(i) &= base_q + \textstyle\sum_{g_k \in G, d_q < g_k \leq g_j} h(g_k) \\ &+ \textstyle\sum_{g_k \in G, d_q \leq g_k < g_j} shift_k^{\max} + shift_j(i). \end{aligned}$$

Analogously, if $g_j \neq d_q$, we set

$$\begin{aligned} shift(g_j) &= base_q + \textstyle\sum_{g_k \in G, d_q < g_k \leq g_j} h(g_k) \\ &+ \textstyle\sum_{g_k \in G, d_q \leq g_k < g_j} shift_k^{\max}. \end{aligned}$$

This concludes the description of `delete&shift`. We next show that the final packing has the desired properties. Next lemma shows that the total profit of deleted items is small for a proper choice of the starting round parameter r_{hor}.

Lemma 52. *There is a choice of $r_{hor} \in \{1, \ldots, \frac{1}{\varepsilon}\}$ such that the final set D_{hor} of deleted items satisfies $p(D_{hor}) \leq 2\varepsilon \cdot p(OPT_{hor})$.*

Proof. Let $\mathscr{G}^r$ denote the union of the sets $G(B)$ computed by all the recursive calls with input round parameter r. Observe that by construction these sets are disjoint. Let also $\mathscr{D}^r$ be the union of the sets $D'(B)$ on those calls (the union of sets $D(B)$ for $r = r_{hor}$). By Proposition 51 and the disjointness of sets $\mathscr{G}^r$ one has

$$\begin{aligned} p(D_{hor}) &= \textstyle\sum_{1 \leq r \leq r_{hor}} p(\mathscr{D}^r) \\ &\leq \varepsilon \cdot \textstyle\sum_{r < r_{hor}} p(\mathscr{G}^r) + p(\mathscr{D}^{r_{hor}}) \\ &\leq \varepsilon \cdot p(OPT_{hor}) + p(\mathscr{D}^{r_{hor}}). \end{aligned}$$

Again by the disjointness of sets $\mathscr{G}^r$ (hence $\mathscr{D}^r$), there must exists a value of $r_{hor} \in \{1, \ldots, \frac{1}{\varepsilon}\}$ such that $p(\mathscr{D}^{r_{hor}}) \leq \varepsilon \cdot p(OPT_{hor})$. The claim follows. $\square$

Next lemma shows that, intuitively, items are only shifted down with respect to the initial packing.

Lemma 53. *Let $(D, shift)$ be the output of some execution of* `delete&shift`(B, r). *Then, for any $i \in B \setminus D$, $shift(i) \leq top_B(i)$.*

Proof. We prove the claim by induction on r. Consider first the case $r = 1$. In this case, for any $i \in B_j^G$:

$$\begin{aligned} & shift(i) \\ =& \lceil bottom_B(g_j) \rceil_{h(g_j)/2} + \sum_{k \in B_j^G, k \leq i} \lceil h(k) \rceil_{h(g_j)/(2n)} \\ \leq& top_B(g_j) - \frac{1}{2} h(g_j) + \sum_{k \in B_j^G, k \leq i} h(k) + n \cdot \frac{h(g_j)}{2n} \\ =& top_B(i). \end{aligned}$$

Assume next that the claim holds up to round parameter $r - 1 \geq 1$, and consider round r. For any $i \in B_j^G$ with $base_q = \lceil bottom_B(d_q)\rceil_{h(d_q)/2}$, one has

$$\begin{aligned}
& shift(i) \\
=& \lceil bottom_B(d_q)\rceil_{h(d_q)/2} + \sum_{g_k \in G, d_q < g_k \leq g_j} h(g_k) \\
+& \sum_{g_k \in G, d_q \leq g_k < g_j} shift_k^{\max} + shift_j(i) \\
\leq& top_B(d_q) + \sum_{g_k \in G, d_q < g_k \leq g_j} h(g_k) \\
+& \sum_{g_k \in G, d_q \leq g_k < g_j} top_{B_k^G}(g_{k+1} - 1) + top_{B_j^G}(i) \\
=& top_B(i).
\end{aligned}$$

An analogous chain of inequalities shows that $shift(g_j) \leq top_B(g_j)$ for any $g_j \in G \setminus D'$. A similar proof works for the special case $base_q = 0$. □

It remains to show that the final set of values of $top(i) = shift(i)$ has the desired properties. This is the most delicate part of our analysis. We define a set $\mathscr{C}_{top}^r$ of candidate top coordinates recursively in r.

- Set $\mathscr{C}_{top}^1$ contains, for any item $j \in \mathscr{R}_{hor}$, and any integer $1 \leq a \leq 4n^2$, the value $a \cdot \frac{h(j)}{2n}$.
- Set $\mathscr{C}_{top}^r$, for $r > 1$ is defined recursively w.r.t. to $\mathscr{C}_{top}^{r-1}$. For any item j, any integer $0 \leq a \leq 2n - 1$, any tuple of $b \leq 1/\varepsilon - 1$ items $j(1), \ldots, j(b)$, and any tuple of $c \leq 1/\varepsilon$ values $s(1), \ldots, s(c) \in \mathscr{C}_{top}^{r-1}$, $\mathscr{C}_{top}^r$ contains the sum $a \cdot \frac{h(j)}{2} + \sum_{k=1}^{b} h(j(k)) + \sum_{k=1}^{c} s(k)$.

Note that sets $\mathscr{C}_{top}^r$ can be computed based on the input only (without knowing OPT). It is easy to show that $\mathscr{C}_{top}^r$ has polynomial size for $r = O_\varepsilon(1)$.

Lemma 54. *For any integer $r \geq 1$, $|\mathscr{C}_{top}^r| \leq (2n)^{\frac{r+2+(r-1)\varepsilon}{\varepsilon^{r-1}}}$.*

Proof. We prove the claim by induction on r. The claim is trivially true for $r = 1$ since there are n choices for item j and $4n^2$ choices for the integer a, hence altogether at most $n \cdot 4n^2 < 8n^3$ choices. For $r > 1$, the number of possible values of $\mathscr{C}_{top}^r$ is at most

$$\begin{aligned}
& n \cdot 2n \cdot \left(\sum_{b=0}^{1/\varepsilon - 1} n^b\right) \cdot \left(\sum_{c=0}^{1/\varepsilon} |\mathscr{C}_{top}^{r-1}|^c\right) \leq 4n^2 \cdot n^{\frac{1}{\varepsilon} - 1} \cdot |\mathscr{C}_{top}^{r-1}|^{\frac{1}{\varepsilon}} \\
& \leq (2n)^{\frac{1}{\varepsilon}+1}\left((2n)^{\frac{r+1+(r-2)\varepsilon}{\varepsilon^{r-2}}}\right)^{\frac{1}{\varepsilon}} \leq (2n)^{\frac{r+2+(r-1)\varepsilon}{\varepsilon^{r-1}}}.
\end{aligned}$$

□

Next lemma shows that the values of *shift* returned by `delete&shift` for round parameter r belong to $\mathscr{C}^r_{top}$, hence the final top coordinates belong to $\mathscr{C}_{top} := \mathscr{C}^{r_{hor}}_{top}$.

Lemma 55. *Let* $(D, shift)$ *be the output of some execution of* `delete&shift`(B, r)*. Then, for any* $i \in B \setminus D$*,* $shift(i) \in \mathscr{C}^r_{top}$*.*

Proof. We prove the claim by induction on r. For the case $r = 1$, recall that for any $i \in B^G_j$ one has

$$
\begin{aligned}
shift(i) = &\lceil bottom_B(g_j) \rceil_{h(g_j)/2} \\
&+ \sum_{k \in B^G_j, k \le i} \lceil h(k) \rceil_{h(g_j)/(2n)}.
\end{aligned}
$$

By Lemma 50, $bottom_B(g_j) = \sum_{k \in B, k < g_j} h(k) \le (n-1) \cdot h(g_j)$. By the same lemma, $\sum_{k \in B^G_j, k \le i} h(k) \le (n-1) \cdot h(g_j)$. It follows that

$$
\begin{aligned}
shift(i) &\le 2(n-1) \cdot h(g_j) + \frac{h(g_j)}{2} + (n-1) \cdot \frac{h(g_j)}{2n} \\
&\le 4n^2 \cdot \frac{h(g_j)}{2n}.
\end{aligned}
$$

Hence $shift(i) = a \cdot \frac{h(g_j)}{2n}$ for some integer $1 \le a \le 4n^2$, and $shift(i) \in \mathscr{C}^1_{top}$ for $j = g_j$ and for a proper choice of a.

Assume next that the claim is true up to $r - 1 \ge 1$, and consider the case r. Consider any $i \in B^G_j$, and assume $0 < base_q = \lceil bottom_B(d_q) \rceil_{h(d_q)/2}$. One has:

$$
\begin{aligned}
shift(i) = \lceil bottom_B(d_q) \rceil_{h(d_q)/2} &+ \sum_{g_k \in G, d_q < g_k \le g_j} h(g_k) \\
+ \sum_{g_k \in G, d_q \le g_k < g_j} & shift^{\max}_k + shift_j(i).
\end{aligned}
$$

By Lemma 50, $bottom_B(d_q) \le (n-1) h(d_q)$, therefore $\lceil bottom_B(d_q) \rceil_{h(d_q)/2} = a \cdot \frac{h(d_q)}{2}$ for some integer $1 \le a \le 2(n-1) + 1$. By Proposition 51, $|\{g_k \in G, d_q < g_k \le g_j\}| \le 1/\varepsilon - 1$. Hence $\sum_{g_k \in G, d_q < g_k \le g_j} h(g_k)$ is a value contained in the set of sums of $b \le 1/\varepsilon - 1$ item heights. By the inductive hypothesis $shift^{\max}_k, shift_j(i) \in \mathscr{C}^{r-1}_{top}$. Hence by a similar argument the value of $\sum_{g_k \in G, d_q \le g_k < g_j} shift^{\max}_k + shift_j(i)$ is contained in the set of sums of $c \le 1/\varepsilon - 1 + 1$ values taken from $\mathscr{C}^{r-1}_{top}$. Altogether, $shift(i) \in \mathscr{C}^r_{top}$. A similar argument, without the term $shift_j(i)$, shows that

$shift(g_j) \in \mathscr{C}^r_{top}$ for any $g_j \in G \backslash D'$. The proof works similarly in the case $base_q = 0$ by setting $a = 0$. The claim follows. □

Proof of Lemma 49. We apply the procedure `delete&shift` to OPT_{hor} as described before, and a symmetric procedure to OPT_{ver}. In particular the latter procedure computes a set $D_{ver} \subseteq OPT_{ver}$ of deleted items, and the remaining items are shifted to the left so that their right coordinate belongs to a set $\mathscr{C}_{bottom} := \mathscr{C}^{r_{ver}}_{bottom}$, defined analogously to the case of $\mathscr{C}_{top} := \mathscr{C}^{r_{hor}}_{top}$, for some integer $r_{ver} \in \{1, \ldots, 1/\varepsilon\}$ (possibly different from r_{hor}, though by averaging this is not critical).

It is easy to see that the profit of non-deleted items satisfies the claim by Lemma 52 and its symmetric version. Similarly, the sets $\mathscr{C}_{top}$ and $\mathscr{C}_{bottom}$ satisfy the claim by Lemmas 54 and 55, and their symmetric versions. Finally, w.r.t. the original packing non-deleted items in OPT_{hor} and OPT_{ver} can be only shifted to the bottom and to the left, resp., by Lemma 53 and its symmetric version. This implies that the overall packing is feasible. □

This concludes the proof of Theorem 38. In what follows we will see how to bound the profit of the best structured solutions and hence obtain approximation algorithms in each case due to the fact that we can now search for L&C packings almost optimally as described in Section 4.2.1.

4.4 2DK without Rotations

As discussed in Section 4.2, it just remains to provide bounds for the profit of the best L&C packing satisfying that:

- the number of containers is constant;
- the possible sizes for the containers are polynomially many and can be computed efficiently;
- the possible sizes of the L-shaped region are polynomially many and can be computed efficiently;
- it is possible to include back large items to our solutions.

This section is devoted to provide such bounds for the setting without rotations, first in the unweighted case and then in the weighted case.

4.4.1 A Simple Improved Approximation for Cardinality 2DK

We start by presenting a simple improved approximation for the cardinality case of 2DK. The argumentation was already sketched in Section 4.0.1 but now we show the proof in full detail. Notice first that, regarding large items, we can assume that the optimal solution $OPT \subseteq \mathcal{R}$ satisfies that $|OPT| \geq 1/\varepsilon^3$ since otherwise we can solve the problem optimally by brute force in time $n^{O(1/\varepsilon^3)}$. Therefore, we can discard from the input all *large* items with both sides larger than $\varepsilon \cdot N$: any feasible solution can contain at most $1/\varepsilon^2$ such items, and discarding them leaves a solution of cardinality at least $(1-\varepsilon)|OPT|$. Let OPT' denote this slightly sub-optimal solution obtained by removing large items.

We will need the following technical lemma, that holds also in the weighted case (see also Fig.4.1.(b)-(d)).

Lemma 56. *Let $\mathcal{H}$ and $\mathcal{V}$ be given subsets of items from some feasible solution with width and height strictly larger than $N/2$ respectively. Then there exists an L-packing of a set $APX \subseteq \mathcal{H} \cup \mathcal{V}$ with $p(APX) \geq \frac{3}{4}(p(\mathcal{H}) + p(\mathcal{V}))$ into the area $L = ([0,N] \times [0,h(\mathcal{H})]) \cup ([0,w(\mathcal{V})] \times [0,N])$.*

Proof. Let us consider the packing of $\mathcal{H} \cup \mathcal{V}$. Consider each $i \in \mathcal{H}$ that has no $j \in \mathcal{V}$ to its top (resp., to its bottom) and shift it up (resp. down) until it hits another $i' \in \mathcal{H}$ or the top (resp, bottom) side of the knapsack. Note that, since $h(j) > N/2$ for any $j \in \mathcal{V}$, one of the two cases above always applies. We iterate this process as long as possible to move any such i. We perform a symmetric process on $\mathcal{V}$. At the end of the process all items in $\mathcal{H} \cup \mathcal{V}$ are stacked on the 4 sides of the knapsack[4].

Next we remove the least profitable of the 4 stacks: by a simple permutation argument we can guarantee that this is the top or right stack. We next discuss the case that it is the top one, the other case being symmetric. We show how to repack the remaining items in a boundary L of the desired size by permuting items in a proper order. In more detail, suppose that the items packed on the left (resp., right and bottom) have a total width of w_l (resp., total width of w_r and total height of h_b). We next show that there exists a packing into $L' = ([0,N] \times [0,h_b]) \cup ([0,w_l + w_r] \times [0,N])$. We prove the claim by induction. Suppose that we have proved it for all packings into left, right and bottom stacks with parameters w_l', w_r', and h' such that $h' < h_b$ or $w_l' + w_r' < w_l + w_r$ or $w_l' + w_r' = w_l + w_r$ and $w_r' < w_r$.

[4]It is possible to permute items in the left stack so that items appear from left to right in non-increasing order of height, and symmetrically for the other stacks. This is not crucial for this proof, but we implemented this permutation in Fig.4.1.(c).

In the considered packing we can always find a guillotine cut ℓ, such that one side of the cut contains precisely one *lonely* item among the leftmost, rightmost and bottom-most items. Let ℓ be such a cut. First assume that the lonely item j is the bottom-most one. Then by induction the claim is true for the part above ℓ since the part of the packing above ℓ has parameters w_l, w_r, and $h - h(j)$. Thus, it is also true for the entire packing. A similar argument applies if the lonely item j is the leftmost one.

It remains to consider the case that the lonely item j is the rightmost one. We remove j temporarily and shift *all* other items by $w(j)$ to the right. Then we insert j at the left (in the space freed by the previous shifting). By induction, the claim is true for the resulting packing since it has parameters $w_l + w(j)$, $w_r - w(j)$ and h, resp. □

For our algorithm, we consider the following three solutions. The first uses an L that occupies the full knapsack, i.e., $w_L = h_L = N$. Let $OPT_{long} \subseteq OPT$ be the items in OPT with height or width strictly larger than $N/2$ and define $OPT_{short} = OPT \setminus OPT_{long}$. We apply Lemma 56 to OPT_{long} and hence obtain a packing for this L with a profit of at least $\frac{3}{4}p(OPT_{long})$.

For the other two packings we will employ one-sided resource augmentation PTAS (Lemma 19). We apply it to the slightly reduced knapsacks $[0,N] \times [0,N/(1+\varepsilon)]$ and $[0,N/(1+\varepsilon)] \times [0,N]$ such that in both cases it outputs a solution that fits in the full knapsack $[0,N] \times [0,N]$ and whose profit is by at most a factor $1 + O(\varepsilon)$ worse than the optimal solution for the respective reduced knapsacks. We will prove that one of these solutions yields a profit of at least $(\frac{1}{2} - O(\varepsilon))p(OPT) + (\frac{1}{4} - O(\varepsilon))p(OPT_{short})$ and hence one of our packings yields a $(\frac{16}{9} + \varepsilon)$-approximation.

Lemma 57. *There exists an L&C packing of profit at least $(\frac{9}{16} + \varepsilon)OPT$ for the cardinality case of 2DK.*

Proof. Let OPT be the considered optimal solution with $opt = p(OPT)$. Recall that there are no large items. Let also $Ver \subseteq OPT$ be the (*vertical*) items with height more than $\varepsilon \cdot N$ (hence with width at most $\varepsilon \cdot N$), and $Hor = OPT \setminus Ver$ (*horizontal* items). Note that with this definition both sides of a horizontal item might have a length of at most $\varepsilon \cdot N$. We let $opt_{long} = p(OPT_{long})$ and $opt_{short} = p(OPT_{short})$.

As mentioned above, we already know that the best L-packing has a total profit of at least $(\frac{3}{4} - O(\varepsilon))opt_{long}$ which can be seen by applying Lemma 56 with $H = OPT_{long} \cap Hor$ and $V = OPT_{long} \cap Ver$. In order to show that the other two packings yield a good profit, consider a *random horizontal strip* $S = [0,N] \times [a, a + \varepsilon \cdot N]$ (fully contained in the knapsack) where $a \in [0,(1-\varepsilon)N)$ is chosen unformly at random. We remove all items from OPT intersecting S. Each item in Hor and $OPT_{short} \cap Ver$ is deleted with probability at most 3ε and $\frac{1}{2} + 2\varepsilon$ respectively. Therefore the total profit of the remaining items is in expectation at least $(1 - 3\varepsilon)p(Hor) + (\frac{1}{2} - 2\varepsilon)p(OPT_{short} \cap Ver)$. Observe that the resulting solution can be packed into a restricted knapsack of size $[0, N] \times [0, N/(1+\varepsilon)]$ by shifting down the items above the horizontal strip. Therefore, when we apply Lemma 19 to the knapsack $[0, N] \times [0, N/(1+\varepsilon)]$, up to a factor $1-\varepsilon$, we show that a container packing of at least the same profit exists. In other terms, this profit is at least $(1-4\varepsilon)p(Hor) + (\frac{1}{2} - \frac{5}{2}\varepsilon)p(OPT_{short} \cap Ver)$.

By a symmetric argument, we show that a container packing of profit at least $(1-4\varepsilon)p(Ver) + (\frac{1}{2} - \frac{5}{2}\varepsilon)p(OPT_{short} \cap Hor)$ exists when we apply Lemma 19 to the knapsack $[0, N/(1+\varepsilon)] \times [0, N]$. Thus the best of the latter two solutions has profit at least $(\frac{1}{2} - 2\varepsilon)opt_{long} + (\frac{3}{4} - \frac{13}{4}\varepsilon)opt_{short} = (\frac{1}{2} - 2\varepsilon)opt + (\frac{1}{4} - \frac{5}{4}\varepsilon)opt_{short}$. The best of our three solutions has therefore value at least $(\frac{9}{16} - O(\varepsilon))opt$ where the worst case is achieved for roughly $opt_{long} = 3 \cdot opt_{short}$. □

Thanks to this existential lemma and the algorithm in Section 4.2.1 we conclude the following theorem. Notice that there is no need to include back small items as they are already packed into containers due to Lemma 19, and also the requirements for the number of containers and the possible sizes of the containers and the L-shaped region are satisfied.

Theorem 58. *There is a* $\left(\frac{16}{9} + \varepsilon\right)$*-approximation for the cardinality case of 2DK.*

In the above result we use either an L-packing or a container packing. If we also consider solutions that combine both kind of packings it is possible to obtain an improved approximation ratio as stated in Theorem 39. In particular, we consider configurations were long items (or a subset of them) can be packed into a relatively small L, and pack part of the remaining short items in the complementary rectangular region using Steinberg theorem (Theorem 5) and related techniques. This proof is deferred to Appendix C.

4.4.2 Weighted Case without Rotations

Now we proceed with the weighted case of 2DK. Our main goal is to prove the following theorem:

Lemma 59. *There exists an L&C packing of profit at least* $(\frac{9}{17} + \varepsilon)OPT$ *for the weighted case of 2DK.*

In order to prove this we will first describe a method to "process" corridors from Lemma 44 and further decompose them into rectangular boxes with items inside, and then analyze different candidate packings to prove the desired bound.

For the sake of simplicity we will first assume that we are able to drop $O_\varepsilon(1)$ items at no cost. Then in Appendix B we will explain how to get rid of this assumption.

Partitioning Corridors into Rectangular Boxes

As explained in Section 2.4 we can temporarily remove small items OPT_{small} and then include them back almost optimally as long as we ensure the required area guarantees for the considered candidate solutions. Also, since $|OPT_{large}| \leq O(1/\varepsilon_{large}^2)$ we can assume that there are no large items in the instance, hence we may now assume that we need to pack only the skewed items from OPT_{skew}.

We proceed to describe a routine to partition the corridors into rectangular boxes such that each item is contained in one such box. We remark that to achieve this partitioning we sometimes have to sacrifice a large fraction of OPT_{corr}, hence we do not achieve a $(1 + \varepsilon)$-approximation as in the work of Adamaszek and Wiese [2015]. On the positive side, we generate only a constant (rather than polylogarithmic) number of boxes. This is crucial to obtain polynomial time algorithms in the later steps.

Recall that each $i \in OPT_{corr}$ is univocally associated with the only subcorridor that fully contains it. We say that we *delete* a sub-corridor, when we delete all items univocally associated with the subcorridor. Note that in the deletion of a sub-corridor we do not delete items that are partially contained in that sub-corridor but completely contained in a neighbor sub-corridor. Given a corridor, we sometimes *delete* some of its subcorridors, and consider the *residual* corridors (possibly more than one) given by the union of the remaining subcorridors. Note that removing any subcorridor from a closed corridor turns it into an open corridor. We implicitly assume that items associated with a deleted subcorridor are also removed (and consequently the corresponding area can be used to pack other items).

Given a corridor, we partition its area into a constant number of boxes as follows (see also Figure 4.5, and Adamaszek and Wiese [2015] for a more detailed description of an analogous construction). Let S be one of its boundary subcorridors (if any), or the central subcorridor of a U-bend. Note that one such S must exist (trivially for an open corridor, otherwise by Lemma 45.2). In the corridor partition, there might be several subcorridors fulfilling the latter condition. We will explain later in which order to process the subcorridors, here we explain

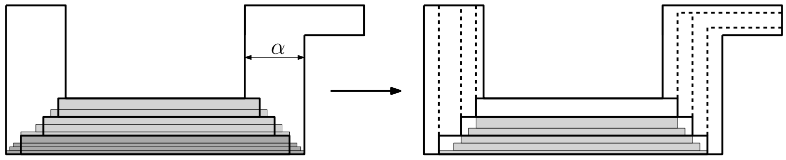


Figure 4.5. Our operation that divides a corridor into $O_\varepsilon(1)$ boxes and $O_\varepsilon(1)$ shorter corridors. The dark gray items show thin items that are removed in this operation. The light gray items are fat items that are shifted to the box below their respective original box. The value α denotes the width of the depicted corridor.

only how to apply our routine to *one* subcorridor, which we call *processing* of a subcorridor.

Suppose that S is horizontal with height b, with the shorter horizontal associated edge being the top one. The other cases are symmetric. Let $\varepsilon_{box} > 0$ be a sufficiently small constant to be defined later. If S is the only subcorridor in the considered corridor, S forms a box and all its items are marked as *fat*. Otherwise, we draw $1/\varepsilon_{box}$ horizontal lines that partition the private region of S into subregions of height $\varepsilon_{box} b$. We mark as *thin* the items of the bottom-most (i.e., the widest) such subregion, and as *killed* the items of the subcorridor cut by these horizontal lines. All the remaining items of the subcorridor are marked as *fat*.

For each such subregion, we define an associated (horizontal) box as the largest axis-aligned box that is contained in the subregion. Given these boxes, we partition the rest of the corridor into $1/\varepsilon_{box}$ corridors as follows. Let S' be a corridor next to S, say to its top-right. Let P be the set of corners of the boxes contained in the boundary curve between S and S'. We project P vertically on the boundary curve of S' not shared with S, hence getting a set P' of $1/\varepsilon_{box}$ points. We iterate the process on the pair (S', P'). At the end of the process, we obtain a set of $1/\varepsilon_{box}$ boxes from the starting subcorridor S, plus a collection of $1/\varepsilon_{box}$ new (open) corridors each one having one less bend with respect to the original corridor. Later, we will also apply this process on the latter corridors. Each newly created corridor will have one bend less than the original corridor and thus this process eventually terminates. Note that, since initially there are $O_{\varepsilon,\varepsilon_{large}}(1)$ corridors, each one with $O(1/\varepsilon)$ bends, the final number of boxes is $O_{\varepsilon,\varepsilon_{large},\varepsilon_{box}}(1)$.

Remark 60. *Assume that we execute the above procedure on the subcorridors until there is no subcorridor left on which we can apply it. Then we obtain a partition of OPT_{corr} into disjoint sets OPT_{thin}, OPT_{fat} and OPT_{kill} of thin, fat, and killed items, respectively. Note that each order to process the subcorridors leads to different such* *partition. We will define this order carefully in our analysis.*

Remark 61. *By a simple shifting argument, there exists a packing of OPT_{fat} into the boxes. Intuitively, in the above construction each subregion is fully contained in the box associated with the subregion immediately below (when no lower subregion exists, the corresponding items are thin).*

We will from now on assume that the shifting of items as described in Remark 61 has been done. The following lemma summarises some of the properties of the boxes and of the associated partition of OPT_{corr} (independently from the way ties are broken). Let $\mathcal{R}_{hor}$ and $\mathcal{R}_{ver}$ denote the set of horizontal and vertical input items, respectively.

Lemma 62. *The following properties hold:*

1. $|OPT_{kill}| = O_{\varepsilon,\varepsilon_{large},\varepsilon_{box}}(1)$;
2. *For any given constant $\varepsilon_{ring} > 0$, there is a sufficiently small $\varepsilon_{box} > 0$ such that the total height (resp., width) of items in $OPT_{thin} \cap \mathcal{R}_{hor}$ (resp., $OPT_{thin} \cap \mathcal{R}_{ver}$) is at most $\varepsilon_{ring} N$.*

Proof. (1) Each horizontal (resp., vertical) line in the construction can kill at most $1/\varepsilon_{large}$ items, since those items must be horizontal (resp., vertical). Hence we kill $O_{\varepsilon,\varepsilon_{large},\varepsilon_{box}}(1)$ items in total.

(2) The mentioned total height/width is at most $\varepsilon_{box} N$ times the number of subcorridors, which is $O_{\varepsilon,\varepsilon_{large}}(1)$. The claim follows for ε_{box} small enough. □

A profitable L&C packing

In this section we prove that there exists an L&C packing with enough profit. Note that in the previous processing of the corridors we did not specify in which order we partition the subcorridors into boxes. In this section, we give several such orders which will then result in different packings. The last such packing is special since we will modify it a bit to gain some space and then reinsert the thin items that were removed in the process of partitioning the corridors into containers. Afterwards, we will show that one of the resulting packings will yield an approximation ratio of $17/9 + \varepsilon$.

In the remainder of this section we prove Lemma 59, but as mentioned before assuming that we can drop $O_{\varepsilon}(1)$ items at no cost. Hence, formally we will prove that there is an L&C packing for a set $\mathcal{R}'$ of items and a set of $O_{\varepsilon}(1)$ items $\mathcal{R}_{drop} \subseteq$

$\mathscr{R} \setminus \mathscr{R}'$ such that $p(\mathscr{R}') + p(\mathscr{R}_{drop}) \geq (\frac{9}{17} - O(\varepsilon))p(OPT)$. In Appendix B we will prove Lemma 59 in full generality (without dropping any items).

The proof of Lemma 59 involves some case analysis. Recall that we classify subcorridors into short and long, and horizontal and vertical. We further partition short subcorridors as follows: let $S_1, \ldots, S_{k'}$ be the subcorridors of a given corridor, and let $S_1^s, \ldots, S_{k''}^s$ be the subsequence of short subcorridors (if any). Mark S_i^s as *even* if i is so, and *odd* otherwise. Note that corridors are subdivided into several other corridors during the box construction process (see Figure 4.5), and these new corridors might have fewer subcorridors than the initial corridor. However, the marking of the subcorridors (short, long, even, odd, horizontal, vertical) is inherited from the marking of the original subcorridor.

We will describe now 7 different ways to partition the subcorridors into boxes, for some of them we delete some of the subcorridors. Each of these different processing orders will give different sets OPT_{thin}, OPT_{kill} and OPT_{fat}^{cont}, and based on these, we will partition the items into three sets. We will then prove three different lower bounds on $p(OPT_{L\&C})$ w.r.t. the sizes of these three sets using averaging arguments about the seven cases.

Cases 1a, 1b, 2a, 2b: Short horizontal/short vertical subcorridors. We delete either all vertical short (case 1) or all horizontal short subcorridors (case 2). We first process all short subcorridors, then either all vertical (subcases a) or horizontal long ones (subcases b), and finally the remaining (horizontal or vertical, resp.) long ones. We can start by processing all short corridors. Indeed, any such corridor cannot be the center of a Z-bend by Lemma 45 since its two sides would be long, hence it must be boundary or the center of a U-bend. After processing short subcorridors, by the same argument the residual (long) subcorridors are the boundary or the center of a U-bend. So we can process the long subcorridors in any order. This gives in total four cases (see Fig. 4.6 for a depiction of these cases).

Cases 3a, 3b: Even/odd short subcorridors. We delete the odd (resp., even) short subcorridors and then process even (resp., odd) short subcorridors last. We exploit the fact that each residual corridor contains at most one short subcorridor. Then, if there is another (long) subcorridor, there is also one which is boundary (trivially for an open corridor) or the center of a U-bend (by Lemma 45, Property 2). Hence we can always process some long subcorridor leaving the unique short subcorridor as last one. This gives two cases.

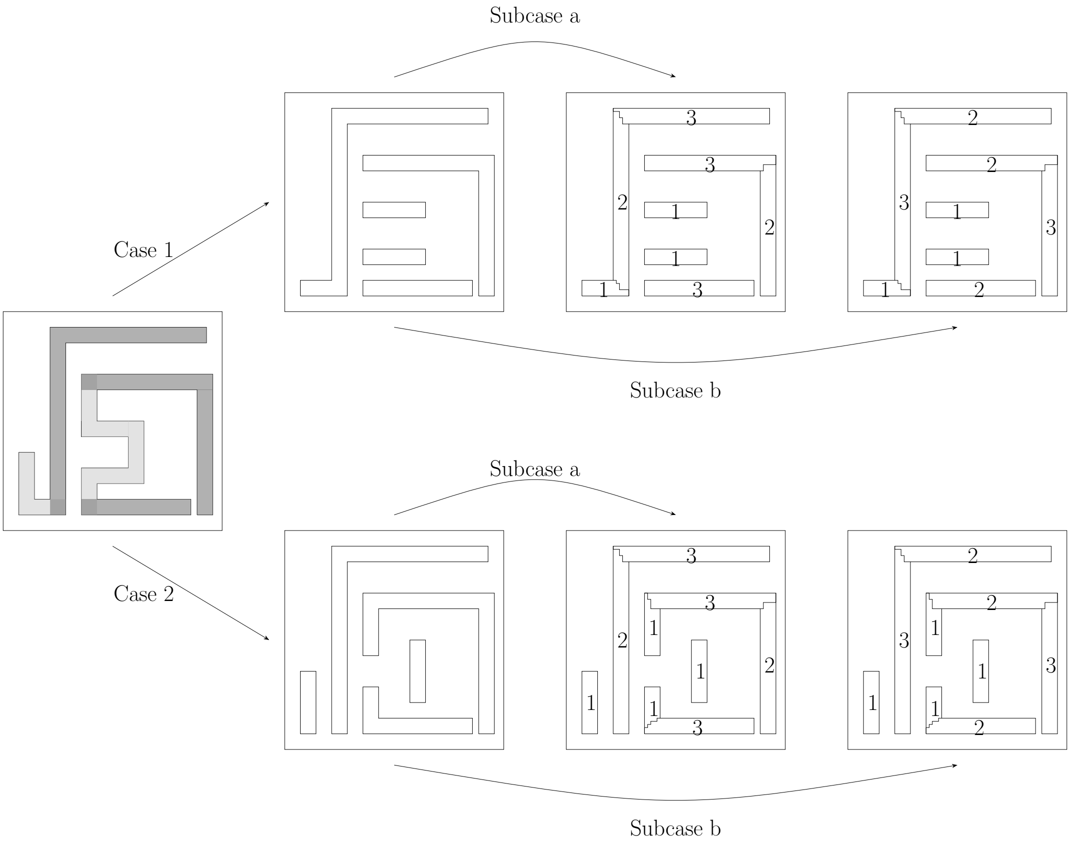


Figure 4.6. Figure for Case 1 and 2. The knapsack on the left contains two corridors, where short subcorridors are marked light grey and long subcorridors are marked dark grey. In case 1, we delete vertical short subcorridors and then consider two processing orders in subcases a and b. In case 2, we delete horizontal short subcorridors and again consider two processing orders in subcases a and b.

Case 4: Fat only. Do not delete any short subcorridor. Process subcorridors in any feasible order.

In each of the cases, we apply the procedure described at the beginning of this section to partition each box into $O_\varepsilon(1)$ containers, which generates a constant number of discarded items. These discarded items will be included into OPT_{kill}. We next label items as follows. Consider the classification of items into OPT_{fat}^{cont}, OPT_{thin}, and OPT_{kill} in each one of the 7 cases above. Then:

- OPT_T is the set of items which are in OPT_{thin} *in at least one case*;

- OPT_K is the set of items which are in OPT_{kill} *in at least one case*;
- OPT_F is the set of items which are in OPT_{fat}^{cont} *in all the cases*.

Remark 63. *Consider the subcorridor of a given corridor that is processed last in any of the above cases. None of its items are assigned to OPT_{thin} in that case and thus essentially all its items are packed in one of the constructed containers. In particular, for an item in set OPT_T, in some of the above cases it might be in such a subcorridor and thus marked fat and packed into a container.*

Lemma 64. *It holds that*

$$p(OPT_F \cup OPT_T) + p(OPT_K) + p(OPT_{large}) + p(OPT_{corr}^{cross}) \geq (1 - O(\varepsilon))p(OPT).$$

Proof. Let us initialize $OPT_F = OPT_{fat}^{cont}$, $OPT_T = OPT_{thin}$, and $OPT_K = OPT_{kill}$ by considering one of the above cases. Next we consider the aforementioned cases, hence moving some items in OPT_F to either OPT_T or OPT_K. Note that initially $p(OPT_F \cup OPT_T) + p(OPT_{kill}) + p(OPT_{large}) + p(OPT_{corr}^{cross}) \geq (1 - O(\varepsilon))p(OPT)$ by Lemma 44 and hence we keep this property. □

Let $\mathscr{R}_{lc}$ and $\mathscr{R}_{sc}$ denote the items in long and short corridors, respectively. We also let $OPT_{LF} = \mathscr{R}_{lc} \cap OPT_F$, and define analogously OPT_{SF}, OPT_{LT}, and OPT_{LF}. The next three lemmas provide a lower bound on the case of a degenerate L.

Lemma 65. $p(OPT_{L\&C}) \geq p(OPT_{LF}) + p(OPT_{SF})$.

Proof. Follows immediately since we pack a superset of OPT_F in case 4. □

Lemma 66. $p(OPT_{L\&C}) \geq p(OPT_{LF}) + \frac{p(OPT_{LT})}{2} + \frac{p(OPT_{SF})}{2}$.

Proof. Consider the sum of the profit of the packed items corresponding to the four subcases of cases 1 and 2. Each $i \in OPT_{LF}$ appears 4 times in the sum (as items in OPT_F are fat in all cases and all long subcorridors get processed), and each $i \in OPT_{LT}$ at least twice by Remark 63: If a long subcorridor $\mathfrak{L}$ neighbors a short subcorridor, the short subcorridor is either deleted or processed first. Further, all neighboring long subcorridors are processed first in case 1a and 2a (if $\mathfrak{L}$ is horizontal, then its neighbors are vertical) or 1b and 2b (if $\mathfrak{L}$ is vertical). Thus, $\mathfrak{L}$ is the last processed subcorridor in at least two cases. Additionally, each item $i \in OPT_{SF}$ also appears twice in the sum, as it gets deleted either in case 1 (if it is vertical) or in case 2 (if it is horizontal), and is fat otherwise. The claim follows by an averaging argument. □

Lemma 67. $p(OPT_{L\&C}) \geq p(OPT_{LF}) + \frac{p(OPT_{SF})}{2} + \frac{p(OPT_{ST})}{2}$.

Proof. Consider the sum of the number of packed items corresponding to cases 3a and 3b. Each $i \in OPT_{LF}$ appears twice in the sum as it is fat and all long subcorridors get processed. Each $i \in OPT_{SF} \cup OPT_{ST}$ appears at least once in the sum by Remark 63: An item $i \in OPT_{SF}$ is deleted in one of the two cases (depending on whether it is in an even or odd subcorridor) and otherwise fat. An item $i \in OPT_{ST}$ is also deleted in one of the two cases and otherwise its subcorridor is processed last. The claim follows by an averaging argument. $\square$

There is one last (and slightly more involving) case to be considered, corresponding to a non-degenerate L.

Lemma 68. $p(OPT_{L\&C}) \geq \frac{3}{4}p(OPT_{LT}) + p(OPT_{ST}) + \frac{1-O(\varepsilon)}{2}p(OPT_{SF})$.

Proof. Recall that $\varepsilon_{large}N$ is the maximum width of a corridor. We consider an execution of the algorithm with a boundary L of width $N' = \varepsilon_{ring}N$, and threshold length $\ell = (\frac{1}{2} + 2\varepsilon_{large})N$. We remark that this length guarantees that items in $\mathscr{R}_{long}$ are not contained in short subcorridors.

By Lemma 56, we can pack a subset of $OPT_T \cap \mathscr{R}_{long}$ of profit at least $\frac{3}{4}p(OPT_T \cap \mathscr{R}_{long})$ in a boundary L of width $\varepsilon_{ring}N$. By Lemma 62 the remaining items in OPT_T can be packed in two containers of size $\ell \times \varepsilon_{ring}N$ and $\varepsilon_{ring}N \times \ell$ that we place on the two sides of the knapsack not occupied by the boundary L.

In the free area we can identify a square region K'' of width and height $(1-\varepsilon)N$. We next show that there exists a feasible solution $OPT'_{SF} \subseteq OPT_{SF}$ with $p(OPT'_{SF}) \geq (1-O(\varepsilon))p(OPT_{SF})/2$ that can be packed in a square of side length $(1-3\varepsilon)N$. We can then apply the Resource Augmentation (Lemma 19) to obtain a container packing of $OPT''_{SF} \subseteq OPT'_{SF}$ having profit $p(OPT''_{SF}) \geq (1-O(\varepsilon))p(OPT'_{SF})$ inside a central square region of width and height $(1-3\varepsilon)(1+\varepsilon_{ra})N \leq (1-2\varepsilon)N$.

Consider the packing of OPT_{SF} as in the optimum solution. Choose a random vertical (resp., horizontal) strip in the knapsack of width (resp., height) $3\varepsilon N$. Delete from OPT_{SF} all the items intersecting the vertical and horizontal strips: clearly the remaining items OPT'_{SF} can be packed into a square of side length $(1-3\varepsilon)N$. Consider any $i \in OPT_{SF}$, and assume i is horizontal (the vertical case being symmetric). Recall that it has height at most $\varepsilon_{small}N \leq \varepsilon N$ and width at most $\ell \leq \frac{1}{2} + 2\varepsilon$. Therefore i intersects the horizontal strip with probability at most 5ε and the vertical strip with probability at most $\frac{1}{2} + 8\varepsilon$. Thus by the union bound the probability that item i belongs to OPT'_{SF} can be bounded below by $\frac{1}{2} - 13\varepsilon$. The claim follows by linearity of expectation. $\square$

Lemma 69. *In the packings due to Lemmas 65, 66, 67, and 68 it is possible to cover the items in the solution using $O_\varepsilon(1)$ rectangular regions of total area at most*

$\min\{(1-2\varepsilon)N^2, a(OPT_{corr}) + \varepsilon_{ra}N^2\}$ *while discarding a set* OPT_{disc} *of items of constant cardinality.*

Proof. We will first prove the following claim: Let $\mathscr{R}' \subseteq OPT_{skew}$ be a collection of items that can be packed into the knapsack and decomposed into boxes such that no box contain simultaneously horizontal and vertical items, and $\varepsilon_{cont} > 0$ be a given constant. Then there exists a container packing of $\mathscr{R}'' \subseteq \mathscr{R}'$ into the knapsack and a set $\mathscr{R}_{disc} \subseteq \mathscr{R}' \setminus \mathscr{R}''$ such that:

1. $p(\mathscr{R}'') + p(\mathscr{R}_{disc}) \geq (1 - O(\varepsilon))p(\mathscr{R}')$;
2. The number of containers is $O_{\varepsilon,\varepsilon_{large},\varepsilon_{cont}}(1)$ and their sizes belong to a set of cardinality $n^{O_{\varepsilon,\varepsilon_{large},\varepsilon_{cont}}(1)}$ that can be computed in polynomial time;
3. $|\mathscr{R}_{disc}| \in O_{\varepsilon,\varepsilon_{large},\varepsilon_{cont}}(1)$; and
4. The total area of the containers is upper-bounded by $\min\{(1-2\varepsilon)N^2, a(\mathscr{R}') + \varepsilon_{cont}N^2\}$.

To prove the claim let us focus on a specific box of size $a \times b$, and on the items $\mathscr{R}_{box} \subseteq \mathscr{R}'$ inside it. If $|\mathscr{R}_{box}| = O_\varepsilon(1)$ then we can simply include them in $\mathscr{R}_{disc}$ and think of the box as empty. Otherwise, assume w.l.o.g. that this box (hence its items) is horizontal. We obtain a set $\overline{\mathscr{R}_{box}}$ by removing from $\mathscr{R}_{box}$ all items intersecting a proper horizontal strip of height $3\varepsilon b$. Clearly these items can be repacked in a box of size $a \times (1-3\varepsilon)b$. By a simple averaging argument, it is possible to choose the strip so that the items fully contained in it have total profit at most $O(\varepsilon)p(\mathscr{R}_{box})$. Furthermore, there can be at most $O(1/\varepsilon_{large})$ items that partially overlap with the strip (since items are skewed). We add these items to $\mathscr{R}_{disc}$ and do not pack them.

At this point we can use the Resource Augmentation Lemma (Lemma 19) to pack a large profit subset $\mathscr{R}'_{box} \subseteq \overline{\mathscr{R}_{box}}$ into $O_{\varepsilon_{cont}}(1)$ containers that can be packed in a box of size $a \times (1-3\varepsilon)(1+\varepsilon_{ra})b \leq a \times (1-2\varepsilon)b$. We perform the above operation on each box of the previous construction and define $\mathscr{R}''$ to be the union of the respective $\mathscr{R}'_{box}$. Now it just remains to prove the area guarantees of the claim. Notice first that from each box we either deleted all the items inside or packed a subset of the items into a reduced box having at least $2\varepsilon ab$ area less. It follows that the total area of the containers is at most $(1-2\varepsilon)N^2$. On the other hand, due to the area guarantees of Lemma 19, we know that the total area of the containers is at most $a(\mathscr{R}') + \varepsilon_{cont}N^2$, proving the claim.

Now, in the cases of Lemmas 65, 66 and 67 the rectangular regions correspond to the containers and the required guarantees follow directly from the previous claim.

On the other hand, in the case of Lemma 68, we add to the containers coming from the previous claim four rectangular regions: one of size $\varepsilon_{ring}N \times N$ plus another of size $N \times \varepsilon_{ring}N$ to cover the boundary L, and the two containers of size $\varepsilon_{ring}N \times \ell$ and $\ell \times \varepsilon_{ring}N$ to pack items from OPT_T. Notice that there is a region not occupied by the boundary L nor by the containers of area at least $4\varepsilon N^2 - 4\varepsilon^2 N^2 - 4\varepsilon_{ring}N^2 \geq 2\varepsilon N^2$ for ε_{ring} small enough, e.g., $\varepsilon_{ring} \leq \varepsilon^2$ suffices. Hence the first inequality follows. For the other inequality, from the claim at the beginning of the proof we have that the total area of the rectangular regions is at most

$$a(OPT_{corr}) + \varepsilon_{ra}(1-2\varepsilon)^2 N^2 + 4\varepsilon_{ring}N^2 \leq a(OPT_{corr}) + \varepsilon_{ra}N^2$$

for ε_{ring} small enough. □

Combining the above Lemmas 64, 65, 66, 67, and 68 we achieve the desired approximation factor, assuming that the (dropped) $O_\varepsilon(1)$ items in $OPT_{kill} \cup OPT_{large} \cup OPT_{corr}^{cross} \cup OPT_{disc}$ have zero profit, and due to the guarantees in Lemma 69 we know that such a solution can be found by our algorithm. Notice that the worst case is obtained, up to $1-O(\varepsilon)$ factors, for $p(OPT_{LT}) = p(OPT_{SF}) = p(OPT_{ST})$ and $p(OPT_{LF}) = \frac{5}{4}p(OPT_{LT})$. This gives $p(OPT_{LT}) = \frac{4}{17} \cdot p(OPT_T \cup OPT_F)$ and a total profit of $\frac{9}{17} \cdot p(OPT_T \cup OPT_F)$.

This concludes the proof of Theorem 40 assuming that we can drop $O(1)$ rectangles at no cost. It is possible to drop this assumption by applying a standard shifting argumentation whose details we defer to Appendix B.

4.5 2DK with Rotations

This section is devoted to provide bounds on the profit of the best structured solution for two-dimensional Geometric Knapsack with rotations (2DKR), first in the unweighted case and then in the weighted case. As opposed to the previous section, we will now focus only on purely container-based solutions, which can be handled by our algorithm described in Chapter 2. It is an interesting open question whether it is possible to solve the L-packing problem with 90 degree rotations.

4.5.1 Cardinality 2DKR

Now we proceed to present a polynomial time $\left(\frac{4}{3} + \varepsilon\right)$-approximation algorithm for 2DKR for the cardinality case. In order to do so we will show that there exists a container-based packing for OPT_{skew} having the claimed profit and then argue about small and large items analogously to the approach in Section 4.4.1. Our existential result crucially exploits the following resource contraction lemma, which is our main new idea in the setting with rotations. Along this section we will assume w.l.o.g. that $\varepsilon, \varepsilon_{small} > 0$ are sufficiently small constants.

Lemma 70. *(Resource Contraction Lemma) Suppose that there exists a feasible packing of a set of items $\mathcal{R}$, with $|\mathcal{R}| \geq 1/\varepsilon_{small}^3$. Then it is possible to pack a subset $\mathcal{R}' \subseteq \mathcal{R}$ of cardinality at least $\frac{2}{3}(1-O(\varepsilon))|\mathcal{R}|$ into the region $[0, \left(1-\varepsilon^{\frac{1}{2\varepsilon}+1}\right)N] \times [0, N]$ if rotations are allowed.*

We will first assume that this lemma holds to prove that a profitable structured packing exists, and then we will prove Lemma 70.

Existence of a Profitable Packing

Similarly to the case without rotations, we will first show the existence of a profitable structured solution. More in detail we will prove the following result.

Lemma 71. *There exists a container packing of total profit at least $\left(\frac{3}{4} - O(\varepsilon)\right)|OPT|$ for the cardinality case of 2DKR.*

Proof. As in Section 4.4.1, we assume that there are no large items and temporarily remove small items. Hence, from now on we will assume that there are only skewed items in OPT. We start with the corridor partition from Lemma 44 and also define *thin*, *fat* and *killed* items according to the procedure described in Section 4.4.2. Killed items can be safely discarded since there are only $O_\varepsilon(1)$ of them as stated in Lemma 62. Let T and F be the set of thin and fat items respectively.

A first candidate solution that we can consider packs $(1-\varepsilon)|F|$ items: After the removal of T, we can get a container packing for almost all items in F as discussed in Lemma 65 in Section 4.4.2.

A second candidate solution we consider can pack $(1-O(\varepsilon))(|T|+\frac{2}{3}|F|)$ many items, and here we will make use of our resource contraction lemma (Lemma 70). First of all, thanks to Lemma 62 we can ensure that the total height of the items in T is at most $\frac{\varepsilon^{\frac{1}{2\varepsilon}+1}N}{2}$, and hence we can pack them in a single vertical container of width $\frac{\varepsilon^{\frac{1}{2\varepsilon}+1}N}{2}$ by rotating the horizontal items in T.

Now we consider two possible cases: if $|F| \geq \frac{1}{\varepsilon_{small}^3}$, then we can apply Lemma 70 to show that there exists $F' \subseteq F$ of cardinality at least $\frac{2}{3}(1-O(\varepsilon))|F|$ that can be packed inside $K' := [0, \left(1-\varepsilon^{\frac{1}{2\varepsilon}+1}\right)N] \times [0, N]$. Then we can use the resource augmentation PTAS (Lemma 19) to get a container packing of $(2/3 - O(\varepsilon))|F|$ many items in the region $K'' := [0, \left(1-\varepsilon^{\frac{1}{2\varepsilon}+1}/2\right)N] \times [0, N]$ with the required area guarantees so that small items can be packed later by means of Lemma 18, and place the vertical container for items in T to the right of K'' in the region $[\left(1-\varepsilon^{\frac{1}{2\varepsilon}+1}/2\right)N, 1] \times [0, N]$. This already proves the claim in this first case

On the other hand, if $|F| < \frac{1}{\varepsilon_{small}^3}$, then either $|T| < \frac{1}{\varepsilon_{small}^4}$ in which case we can pack $|F \cup T| < \frac{2}{\varepsilon_{small}^4}$ items just by brute-force, or else $|T| \geq \frac{1}{\varepsilon_{small}^4} \geq \frac{1}{\varepsilon_{small}}|F|$, in which case we pack at least$|T| \geq (1 - O(\varepsilon))(|T| + |F|)$ many items, proving the claim also in this case.

By combining both candidate solutions, the claimed bound on the profit of the best container packing holds. Up to a factor $(1 - O(\varepsilon))$, the worst case is obtained when $|F| = 3|T|$ which gives a total profit of $\frac{3}{4}|OPT|$. □

As discussed before, the required guarantees hold in order to include back almost $|OPT_{small}|$ many items to the solution, and hence by applying our algorithm described in Section 4.2.1 we can conclude the proof of Theorem 42 provided that Lemma 70 is true.

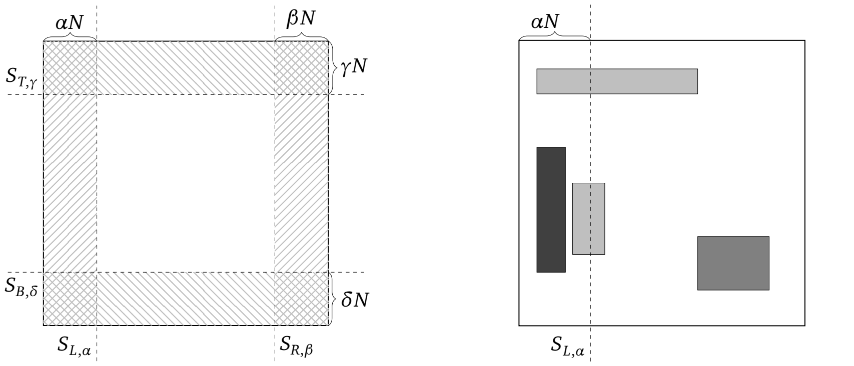
Proof of the Resource Contraction Lemma

Now we will prove Lemma 70. Let us first remove from the set of items $\mathcal{R}$ all the ones having both height and width larger than $\varepsilon_{small}N$. Let $\mathcal{R}_2$ be the resulting set: observe that $|\mathcal{R}_2| \geq (1 - \varepsilon_{small})|\mathcal{R}|$. We next show how to remove from $\mathcal{R}_2$ a set of cardinality at most $\varepsilon|\mathcal{R}_2|$ such that the remaining items $\mathcal{R}_{final}$ are either *very tall* or *not too tall*, where the exact meaning of this will be given next.

Lemma 72. *There exists a value $i \in \{1, \ldots, \lceil 1/(2\varepsilon) \rceil\}$ such that the set of items from $\mathcal{R}_2$ having height $h(i) \in ((1-2\varepsilon^i)N, (1-\varepsilon^{i+1})N]$ has cardinality at most $\varepsilon|\mathcal{R}_2|$.*

Proof. For each $i = 1, \ldots, \lceil 1/(2\varepsilon) \rceil$, let K_i be the set of items in $\mathcal{R}_2$ with height in $((1-2\varepsilon^i)N, (1-\varepsilon^{i+1})N]$. An item can belong to at most two such sets if ε is small enough. Thus, the smallest such set has cardinality at most $\varepsilon|\mathcal{R}_2|$. □

We remove from $\mathcal{R}_2$ the elements from the set K_i of minimum cardinality guaranteed by the above lemma, and let $\mathcal{R}_{final}$ be the resulting set. We also define $\varepsilon_s = \varepsilon^i$ for the same i. Thus, $\varepsilon_s \geq \varepsilon^{1/2\varepsilon} > \varepsilon_{small}/\varepsilon$. Note that the items in $\mathcal{R}_{final}$ have height either at most $(1-2\varepsilon_s)N$ or larger than $(1-\varepsilon \cdot \varepsilon_s)N$.



(a) Depiction of strips $S_{L,\alpha}, S_{R,\beta}, S_{B,\delta}, S_{T,\gamma}$

(b) $C_{L,\alpha}, D_{L,\alpha}$ are dark and light gray items resp.

Figure 4.7. Definitions for cardinality 2DK with rotations.

Now we require the following technical definitions. For any $\delta > 0$ we denote the strips of width N and height δN at the top and bottom of the knapsack by $S_{T,\delta} := [0, N] \times [(1-\delta)N, N]$ and $S_{B,\delta} := [0, N] \times [0, \delta N]$ respectively. Similarly, we denote the strips of height N and width δN to the left and right of the knapsack by $S_{L,\delta} := [0, \delta N] \times [0, N]$ and $S_{R,\delta} := [(1-\delta)N, N] \times [0, N]$ respectively (see Figure 4.7(a)). The set of items in $\mathscr{R}_{final}$ intersected by a strip $S_{K,\delta}$ are denoted by $E_{K,\delta}$ and the set of items fully contained in such a strip are denoted by $C_{K,\delta}$. Obviously $C_{K,\delta} \subseteq E_{K,\delta}$, and we define, $D_{K,\delta} = E_{K,\delta} \setminus C_{K,\delta}$ (see Figure 4.7(b)).

To prove the resource contraction lemma we will make use of the following technical lemmas which provide area guarantees for the items intersected by some of the strips and a way to pack them in a slightly reduced knapsack.

Lemma 73. *Either* $a(E_{L,\varepsilon_s} \cup E_{R,\varepsilon_s}) \leq \frac{(1+8\varepsilon_s)}{2} N^2$ *or* $a(E_{T,\varepsilon_s} \cup E_{B,\varepsilon_s}) \leq \frac{(1+8\varepsilon_s)}{2} N^2$.

Proof. Let us define $V := E_{L,\varepsilon_s} \cup E_{R,\varepsilon_s}$ the set of items intersected by at least one of the vertical strips of width $\varepsilon_s N$ and $H := E_{T,\varepsilon_s} \cup E_{B,\varepsilon_s}$ the set of items intersected by at least one of the horizontal strips of height $\varepsilon_s N$. Note that, $a(V) + a(H) = a(V \cup H) + a(V \cap H)$. Clearly $a(V \cup H) \leq N^2$ since all items fit into the knapsack. On the other hand, except possibly four items (the ones by a vertical and a horizontal strip at the same time) all other items in $V \cap H$ lie entirely within the union of the four strips. Thus $a(V \cap H) \leq 4\varepsilon_s N^2 + 4\varepsilon_{small} N^2 \leq 8\varepsilon_s N^2$, as $\varepsilon_{small} \leq \varepsilon_s$. We can conclude that

$$\min\{a(V), a(H)\} \leq \frac{a(V) + a(H)}{2} = \frac{a(V \cup H) + a(V \cap H)}{2} \leq \frac{(1 + 8\varepsilon_s)}{2} N^2$$

. □

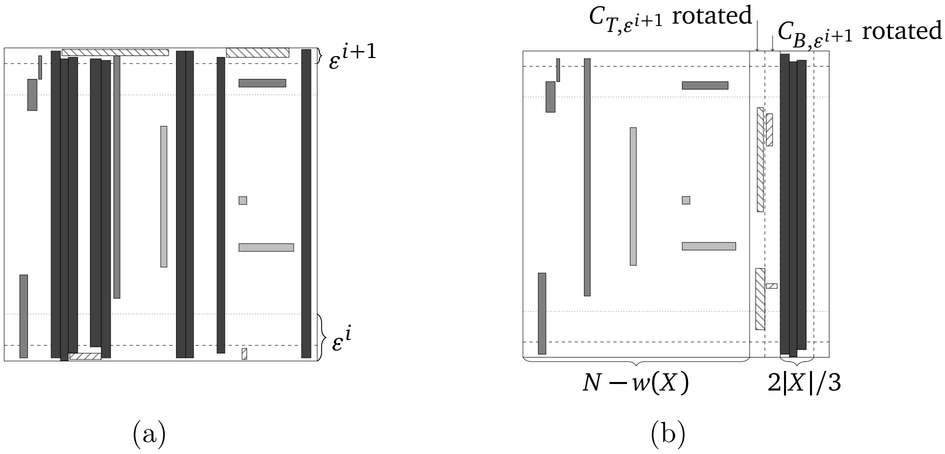


Figure 4.8. Case A for cardinality 2DKR. Dark gray rectangles are X, light gray rectangles are Z, gray rectangles plus hatched rectangles are Y, hatched rectangles are $C_{T,\varepsilon^{i+1}}$ and $C_{B,\varepsilon^{i+1}}$. (a): original packing in $N \times N$. (b): modified packing leaving space for resource contraction on the right.

Lemma 74. *Let $0 < \varepsilon_a < 1/2$ be a constant and $\tilde{\mathcal{R}} := \{1, \dots, k\}$ a set of items satisfying $w(i) \le \varepsilon_{small} N$ for all $i \in \tilde{\mathcal{R}}$. If $a(\tilde{\mathcal{R}}) \le (1/2 + \varepsilon_a)N^2$, then there exists a set of items $S \subseteq \tilde{\mathcal{R}}$ of cardinality at least $(1-2\varepsilon_s-2\varepsilon_a)|\tilde{\mathcal{R}}|$ that can be packed into $[0,(1-\varepsilon_s)N] \times [0,N]$.*

Proof. Let us assume that the items in $\tilde{\mathcal{R}}$ are given in non-decreasing order according to their area. Note that $a(i) \le \varepsilon_{small} N^2 \le \frac{\varepsilon_s}{2} N^2$ for any $i \in \tilde{\mathcal{R}}$. Let $S := \{1, \dots, j\}$ be such that $\frac{(1-2\varepsilon_s)}{2} N^2 \le \sum_{i=1}^{j} a(i) \le \frac{(1-\varepsilon_s)}{2} N^2$ and $\sum_{i=1}^{j+1} a(i) > \frac{(1-\varepsilon_s)}{2} N^2$. Then by using Steinberg theorem (Theorem 5), S can be packed into $[0,(1-\varepsilon_s)N] \times [0,N]$. As we considered items sorted non-decreasingly by area, we have that $\frac{|S|}{|\tilde{\mathcal{R}}|} \ge \frac{(\frac{1}{2}-\varepsilon_s)}{(\frac{1}{2}+\varepsilon_a)}$. Thus, $|S| \ge \left(1 - \frac{(\varepsilon_a+\varepsilon_s)}{(\frac{1}{2}+\varepsilon_a)}\right)|\tilde{\mathcal{R}}| > (1-2\varepsilon_a-2\varepsilon_s)|\tilde{\mathcal{R}}|$. □

Now we have all the required elements to proceed with the proof of Lemma 70. Let us assume that $a(E_{T,\varepsilon_s} \cup E_{B,\varepsilon_s}) \le \frac{(1+8\varepsilon_s)}{2} N^2$ thanks to Lemma 73 (the other case being symmetric). Let us partition $\mathcal{R}_{final}$ into three sets in order to bound the profit of our candidate solutions: Let X be the set of items in $\mathcal{R}_{final}$ intersecting both horizontal strips S_{T,ε_s} and S_{B,ε_s}, $Y := \{E_{T,\varepsilon_s} \cup E_{B,\varepsilon_s}\} \setminus X$ the items intersecting only one of the horizontal strips and $Z := \mathcal{R}_{final} \setminus \{X \cup Y\}$ be the remaining items (see Figure 4.8(a) and 4.9(a) for examples). Now let us consider the following two cases.

- **Case A:** $w(X) \ge 12\varepsilon \cdot \varepsilon_s N$.

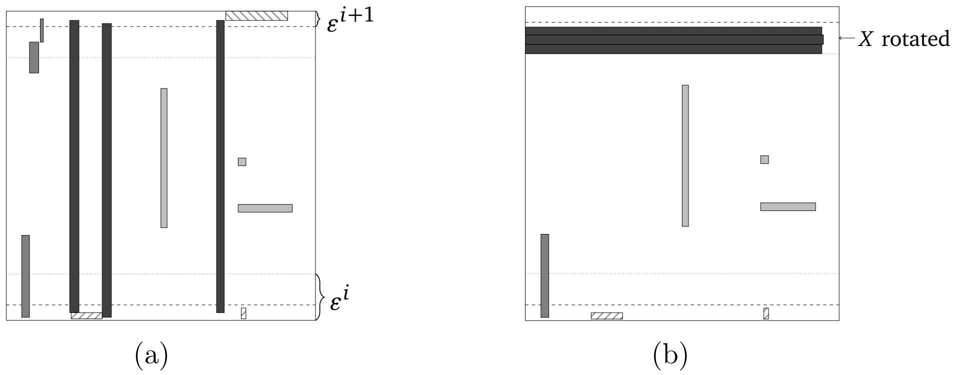


Figure 4.9. Case B for cardinality 2DKR. Dark gray rectangles are X, light gray rectangles are Z, gray rectangles plus hatched rectangles are Y, hatched rectangles are $C_{T,\varepsilon^{i+1}}$ and $C_{T,\varepsilon^{i+1}}$. (a): original packing. (b): modified packing leaving space for resource contraction on the top.

From Lemma 72, all items in X must intersect also $S_{T,\varepsilon\cdot\varepsilon_s}$ and $S_{B,\varepsilon\cdot\varepsilon_s}$. This implies that the removal of $X \cup C_{T,\varepsilon\cdot\varepsilon_s} \cup C_{B,\varepsilon\cdot\varepsilon_s}$ creates a set of empty strips of height N and total width $w(X)$. By a simple permutation argument, all items in $Y \cup Z$ can be packed inside $[0, N - w(X)] \times [0, N]$, leaving an empty vertical strip of width $w(X)$ in the right side of the knapsack. Next we rotate $C_{T,\varepsilon\cdot\varepsilon_s}$ and $C_{B,\varepsilon\cdot\varepsilon_s}$ and pack them in two vertical strips, each one having width $\varepsilon \cdot \varepsilon_s N$. Note that $w(i) \leq \varepsilon \cdot \varepsilon_s N$ for all $i \in X$. Now pick items from X, sorted non-decreasingly by width, until their total width is in $[w(X) - 4\varepsilon \cdot \varepsilon_s N, w(X) - 3\varepsilon \cdot \varepsilon_s N]$ and pack them into another vertical strip. The cardinality of this set is at least $\frac{(w(X)-4\varepsilon\cdot\varepsilon_s N)}{w(X)}|X| \geq \frac{2}{3}|X|$, where the last inequality follows by our assumption on $w(X)$ (see Figure 4.8 for a depiction of the final packing). Hence, at least $\frac{2}{3}|X| + |Y| + |Z| \geq \frac{2}{3}|\mathscr{R}_{final}|$ items can be packed into $[0, (1 - \varepsilon \cdot \varepsilon_s)N] \times [0, N]$.

- **Case B:** $w(X) < 12\varepsilon \cdot \varepsilon_s N$.

  Observe that $Y = (E_{T,\varepsilon_s} \setminus X)\dot{\cup}(E_{B,\varepsilon_s} \setminus X)$, hence $|Y| = |E_{T,\varepsilon_s} \setminus X| + |E_{B,\varepsilon_s} \setminus X|$. Assume w.l.o.g. that $|E_{B,\varepsilon_s} \setminus X| \geq |Y|/2 \geq |E_{T,\varepsilon_s} \setminus X|$. Let us remove E_{T,ε_s}. We can pack X on top of $\mathscr{R} \setminus E_{T,\varepsilon_s}$ as $12\varepsilon \cdot \varepsilon_s \leq \varepsilon_s - \varepsilon \cdot \varepsilon_s$ for ε small enough. This gives a packing of $|X| + |Z| + \frac{|Y|}{2}$ many items. On the other hand, as $a(X \cup Y) = a(E_{T,\varepsilon_s} \cup E_{B,\varepsilon_s}) \leq \frac{(1+8\varepsilon_s)}{2}N^2$, by using Lemma 74 it is possible to pack at least $(1 - 2\varepsilon_s - 8\varepsilon_s)|X \cup Y| \geq (1 - 10\varepsilon_s)(|X| + |Y|)$ many items into $[0, (1 - \varepsilon \cdot \varepsilon_s)N] \times [0, N]$.

  Thus we can always pack a set of items having cardinality at least

$$\begin{aligned} & \max\left\{(1-10\varepsilon_s)(|X|+|Y|), |X|+|Z|+\tfrac{|Y|}{2}\right\} \\ \geq\ & \tfrac{1}{3}(1-10\varepsilon_s)(|X|+|Y|)+\tfrac{2}{3}\left(|X|+|Z|+\tfrac{|Y|}{2}\right) \\ \geq\ & \tfrac{2}{3}(1-10\varepsilon_s)(|X|+|Y|+|Z|) \\ =\ & \tfrac{2}{3}(1-10\varepsilon_s)|\mathcal{R}_{final}|. \end{aligned}$$

This concludes the proof of Lemma 70 as $|\mathcal{R}_{final}| \geq (1-O(\varepsilon))|\mathcal{R}|$.

4.5.2 Weighted 2DKR

In this section we present a polynomial time $\left(\frac{3}{2}+\varepsilon\right)$-approximation algorithm for the weighted version of 2DKR. Differently from the cardinality case, where it is possible to remove a constant number of items at a negligible cost, the same is not possible in the weighted case, where a single item could have a big profit.

As we will see, a problematic case for our approach occurs when one very large item is present in the optimal solution. We call an item i *massive* if $w(i) \geq (1-\varepsilon)N$ and $h(i) \geq (1-\varepsilon)N$. The presence of such a big item in the optimal solution requires a different analysis, that we present below. In both the cases, we can show that there exists a container packing with roughly 2/3 of the profit of the optimal solution.

We will prove the following result:

Theorem 75. *There exists a container packing of total profit at least* $\left(\frac{2}{3}-O(\varepsilon)\right)p(OPT)$ *for the weighted case of 2DKR.*

To achieve the claimed result, we will first prove that in the presence of a massive item it is possible to find such a container packing, and then when there is no massive item we will follow a similar approach to the one developed in Section 4.5.1, proving that it is possible to pack enough profit into a reduced knapsack and then use the previously described techniques to get the required candidate solutions.

Massive item case

We start by analyzing the case where the optimal solution has a massive item m. We will assume, without loss of generality, that $1/(3\varepsilon)$ is an integer. In this specific case we will consider three candidate solutions.

Consider first the items in $OPT \setminus \{m\}$. Clearly, each of them has width or height at most εN; moreover, $a(OPT \setminus \{m\}) \leq (1-(1-\varepsilon)^2)N^2 = (2\varepsilon-\varepsilon^2)N^2 \leq \frac{N^2}{2(1+\varepsilon)}$ if ε is small enough. Thus, by possibly rotating each element so that the height is smaller than ε, by Steinberg theorem (Theorem 5) all the items in $OPT \setminus \{m\}$ can be packed in a box of size $N \times \frac{N}{1+\varepsilon}$; then, by Lemma 19, there is a feasible container packing for a subset of $OPT \setminus \{m\}$ of profit at least $(1-O(\varepsilon))p(OPT \setminus \{m\})$.

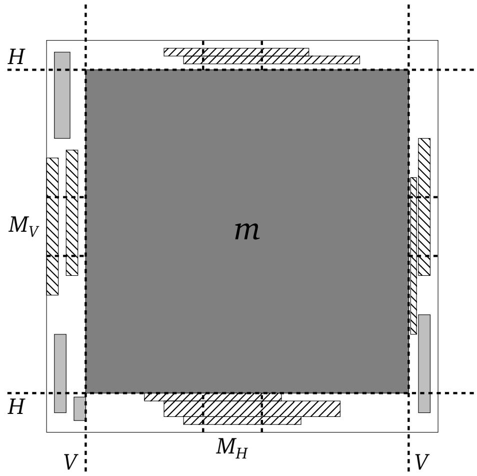
Consider now the whole solution. From the definition of massive item, the region $[\varepsilon N, (1-\varepsilon)N]^2$ is entirely covered by the massive item m. Let us partition the region with x-coordinate between εN and $(1-\varepsilon)N$ into $k = 1/(3\varepsilon)$ strips $S_1, \ldots, S_k$ of width $3\varepsilon(1-2\varepsilon)N \geq 2\varepsilon N$ and height N each. Let $OPT(S_i)$ be the set of items such that their left or right edge (or both) is contained in the interior of strip S_i. Since each item belongs to at most two of these sets, there exists i such that $p(OPT(S_i)) \leq 6\varepsilon p(OPT)$. Symmetrically, we define k horizontal strips $T_1, \ldots, T_k$, obtaining an index j such that $p(OPT(T_j)) \leq 6\varepsilon p(OPT)$. If we remove from the solution the items in $(OPT(S_i) \cup OPT(T_j))$, obtaining OPT', every remaining item is either disjoint from S_i or T_j, or completely crosses one of them. Furthermore, we have that $p(OPT') \geq (1-12\varepsilon)p(OPT)$. Let M_V be the set of items in $OPT' \setminus \{m\}$ that overlap T_j, and let M_H be the set of items in $OPT' \setminus \{m\}$ that overlap S_i (see Figure 4.10). Clearly, the items in M_H can be packed inside an horizontal container of width N and height $N - h(m)$, and the items in M_V can be packed in a vertical container of width $N - w(m)$ and height N.

Let H be the set of items of $OPT' \setminus M_H$ that are completely above or completely below the massive item m; symmetrically, let V be the set of items of $OPT' \setminus M_V$ that are completely to the left or completely to the right of m. We will now show that there is a container packing for $M_H \cup V \cup \{m\}$. Since all the elements overlapping T_j have been removed, V can be packed in a box of size $(N - w(m)) \times (1-2\varepsilon)N$ by a simple shifting argument. Since $(1-2\varepsilon)N \cdot (1+\varepsilon) < (1-\varepsilon)N \leq h(m)$, Lemma 19 implies that there is a container packing of a subset of V with profit at least $(1-O(\varepsilon))p(V)$ in a bin of size $(N - w(m)) \times h(m)$ and using $O_\varepsilon(1)$ containers; thus, by adding an horizontal container of the same size as m and an horizontal container of size $N \times (N - h(m))$, we obtain a container packing for $M_H \cup V \cup \{m\}$ with $O_\varepsilon(1)$ containers and profit at least $(1-O(\varepsilon))p(M_H \cup V \cup \{m\})$. Symmetrically, there is a container packing for a subset of $M_V \cup H \cup \{m\}$ with profit at least $(1-O(\varepsilon))p(M_V \cup H \cup \{m\})$ and $O_\varepsilon(1)$ containers.

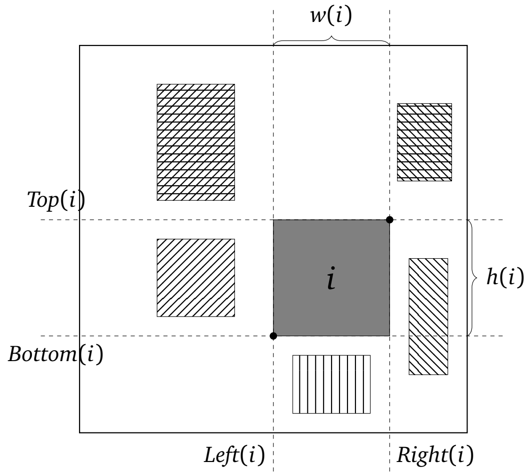
Putting these candidate solutions together we can prove the following.

Lemma 76. *Suppose that there is a massive item $m \in OPT$. Then, there exists a container packing having profit at least $\left(\frac{2}{3} - O(\varepsilon)\right)p(OPT)$.*

Proof. Consider the slightly suboptimal solution OPT' defined above, and notice that $OPT' = \{m\} \cup M_H \cup H \cup M_V \cup V$. By the discussion above, there is a container packing with $O_\varepsilon(1)$ containers and profit at least $(1-O(\varepsilon))\max\{p(OPT' \setminus \{m\}), p(M_H \cup V \cup \{m\}), p(M_V \cup H \cup \{m\})\}$. Since each element in OPT' is contained in at least two of the above solutions, it follows that the profit of the best



(a) Massive item case. Items intersecting strips M_H and M_V (hatched rectangles) cross these stripes completely.



(b) *Bottom*(i), *Top*(i), *Left*(i), *Right*(i) correspond to vertical, horizontal, north east and north west patterns respectively.

Figure 4.10

container packing is at least $\left(\frac{2}{3} - O(\varepsilon)\right)p(OPT)$. □

No massive item case

If there is no massive item in the optimal solution, we will show two candidate container packings and show that the maximum of them always packs at least a $\left(\frac{2}{3} - O(\varepsilon)\right)$ fraction of the optimal profit, bounding then $p(OPT_{L\&C})$. In order to achieve this, similarly to the approach developed in Section 4.5.1, we will prove that it is possible to pack enough profit in a reduced knapsack and then use Lemma 19 to obtain a structured solution. More in detail, we will prove the following resource contraction lemma.

Lemma 77. *(Weighted Resource Contraction Lemma) If a set of items $\mathcal{R}$ does not contain a massive item and can be packed into a box of size $N \times N$, then it is possible to pack a set $\mathcal{R}'$ of profit at least $\frac{1}{2}p(\mathcal{R})$ into a box of size $N \times (1 - \frac{\varepsilon}{2})N$ (into a box of size $(1 - \frac{\varepsilon}{2})N \times N$ resp.) if rotations are allowed.*

Let us assume by now that this lemma holds and show the existence of a profitable packing in this case. Later in the end of this section we prove the resource contraction lemma.

Lemma 78. *Suppose that there is no massive item in OPT. Then, there exists a container packing having profit at least* $\left(\frac{2}{3} - O(\varepsilon)\right)p(OPT)$.

Proof. First, we consider the corridor decomposition from Lemma 44 and the classification of items as in Section 4.4.2 to define sets LF, SF, LT, ST and OPT_{small}. Let $T := LT \cup ST$ be the set of thin items.

Then similarly to Lemma 88, we can show that the profit of the best container packing is at least $(1-\varepsilon)(p(LF) + p(SF) + p(OPT_{small}))$. Thus,

$$p(OPT_{L\&C}) \geq (1-\varepsilon)p(OPT) - p(T).$$

In the second case, we can use our Resource Contraction Lemma (Lemma 77), to show that it is possible to pack $\frac{1}{2}$ of the remaining profit from the optimal solution, i.e., $p(OPT \setminus T)/2$ into a box of size $N \times (1 - \varepsilon/2)N$. Now, we can pack T in a horizontal container of height $\frac{\varepsilon}{4}N$ and using Lemma 77 and resource augmentation we can pack $p(OPT \setminus T)/2$ in the remaining region $N \times (1-\varepsilon/4)N$. Thus,

$$p(OPT_{L\&C}) \geq p(T) + \frac{1-\varepsilon}{2}(p(OPT) - p(T)).$$

Hence, up to $(1 - O(\varepsilon))$ factor, we pack at least

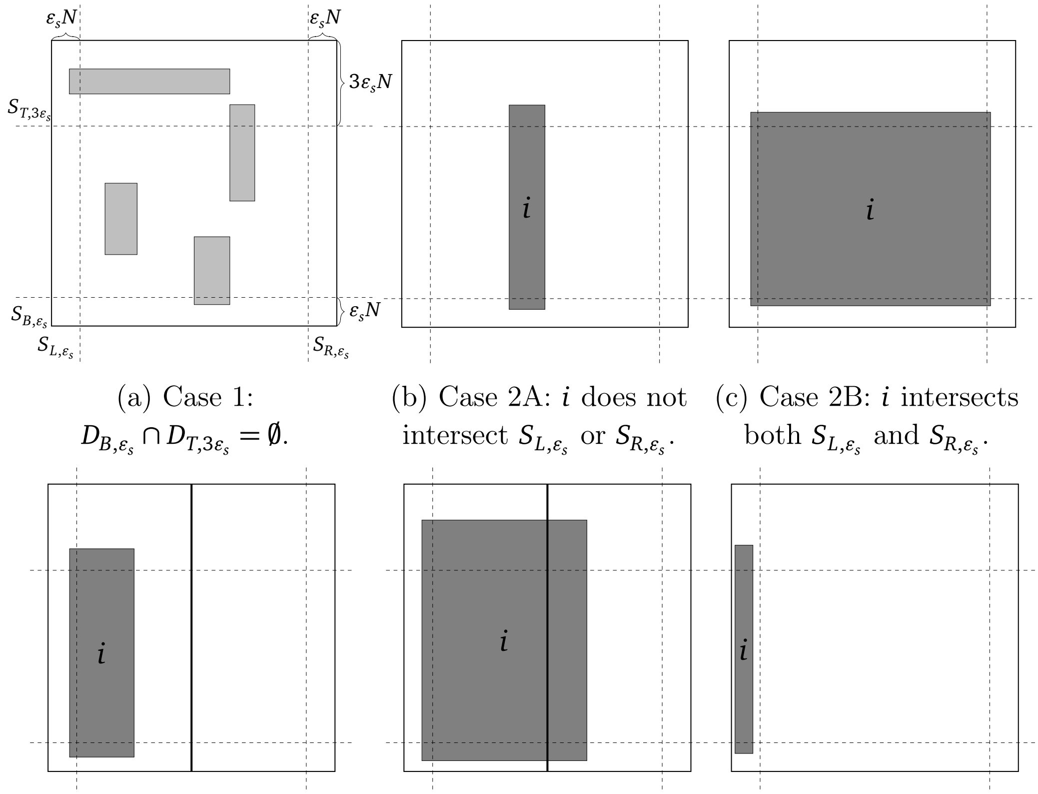
$$\max\{\frac{1}{2}(p(T) + p(OPT \setminus T)), p(OPT \setminus T)\} \geq \frac{2}{3}p(OPT).$$

□

This concludes the proof of Theorem 75. Note that now we can just apply our PTAS for container packings to get the desired approximation algorithm, hence completing the proof of Theorem 41.

Proof of Lemma 77 Let $\varepsilon_s = \varepsilon/2$. We will consider different cases in order to partition $\mathcal{R}$ into two sets $\mathcal{R}_1, \mathcal{R} \setminus \mathcal{R}_1$ and show that each such set can be packed into the reduced knapsack $N \times (1 - \varepsilon_s)N$.

In a packing of a set of items $\mathcal{R}$, for item i we define the sets $Left(i) := \{k \in \mathcal{R} : right(k) \leq left(i)\}$, $Right(i) := \{k \in \mathcal{R} : left(k) \geq right(i)\}$, $Top(i) := \{k \in \mathcal{R} : bottom(k) \geq top(i)\}$, $Bottom(i) := \{k \in \mathcal{R} : top(k) \leq bottom(i)\}$, i.e., the items that lie completely to the left, right, above and below of i respectively. We also consider the following four strips $S_{T,3\varepsilon_s}, S_{B,\varepsilon_s}, S_{L,\varepsilon_s}, S_{R,\varepsilon_s}$ (see Figure 4.11). Recall that for any $\delta > 0$ we denote the strips of width N and height δN at the top and bottom of the knapsack by $S_{T,\delta} := [0,N] \times [(1-\delta)N, N]$ and $S_{B,\delta} := [0,N] \times [0, \delta N]$ respectively. Similarly, we denote the strips of height N and width



(a) Case 1: $D_{B,\varepsilon_s} \cap D_{T,3\varepsilon_s} = \emptyset$.

(b) Case 2A: i does not intersect S_{L,ε_s} or S_{R,ε_s}.

(c) Case 2B: i intersects both S_{L,ε_s} and S_{R,ε_s}.

(d) Case 2C: $i \in D_{L,\varepsilon_s}$ and $right(i) \leq N/2$.

(e) Case 2C: $i \in D_{L,\varepsilon_s}$ and $right(i) > N/2$.

(f) Case 2C: $i \in C_{L,\varepsilon_s}$.

Figure 4.11. Cases for the Resource Contraction Lemma (Lemma 77).

δN to the left and right of the knapsack by $S_{L,\delta} := [0, \delta N] \times [0, N]$ and $S_{R,\delta} := [(1-\delta)N, N] \times [0, N]$ respectively. The set of items in the solution intersected by a strip $S_{K,\delta}$ are denoted by $E_{K,\delta}$, the set of items fully contained in such a strip are denoted by $C_{K,\delta}$and $D_{K,\delta} = E_{K,\delta} \setminus C_{K,\delta}$.

Now we will analyze the different possible cases.

Case 1: No item intersects both S_{B,ε_s} and $S_{T,3\varepsilon_s}$. Let us define in this case $\mathscr{R}_1 := E_{T,3\varepsilon_s}$. As items in $\mathscr{R}_1$ do not intersect S_{B,ε_s}, $\mathscr{R}_1$ can be packed into a $(N, N(1-\varepsilon_s))$ bin. On the other hand, if we keep $\mathscr{R} \setminus (\mathscr{R}_1 \cup C_{L,\varepsilon_s} \cup C_{R,\varepsilon_s})$ in its position, rotate C_{L,ε_s} and C_{R,ε_s} and pack them on top of the previous packing using two strips of height $\varepsilon_s N$ and width N, we get a packing of total height at most $(1 - 3\varepsilon_s + 2\varepsilon_s)N \leq (1-\varepsilon_s)N$.

Case 2: There is some item i intersecting both S_{B,ε_s} and $S_{T,3\varepsilon_s}$. In this case we distinguish three subcases:

Case 2A: Item i does not intersect neither S_{L,ε_s} nor S_{R,ε_s}.
In this case item i partitions $\mathscr{R}_1 \setminus (C_{T,3\varepsilon_s} \cup C_{B,\varepsilon_s} \cup \{i\})$ into two sets: $\mathit{Left}(i)$ and $\mathit{Right}(i)$. W.l.o.g., assume left$(i) \leq 1/2$. If we let $\mathscr{R}_1 = \mathit{Right}(i) \cup \{i\}$, as item i does not intersect the strip S_{L,ε_s}, we can pack $\mathscr{R}_1$ into a box of height N and width $(1-\varepsilon_s)N$. On the other hand, we can pack $\mathscr{R} \setminus \mathscr{R}_1$ in the reduced knapsack as follows: we can rotate $C_{T,3\varepsilon_s}$ and C_{B,ε_s} and pack them to the right of $\mathit{Left}(i)$. This packing of $\mathscr{R} \setminus \mathscr{R}_1$ takes height N and width $\mathit{left}(i) + 4\varepsilon_s N \leq (\frac{1}{2} + 4\varepsilon_s)N \leq (1-\varepsilon_s)N$ if ε_s is small enough.

Case 2B: Item i intersects both S_{L,ε_s} and S_{R,ε_s}.
Let $\mathscr{R}_1$ to be $C_{L,\varepsilon_s} \cup C_{R,\varepsilon_s} \cup \mathit{Top}(i)$. We can pack $\mathit{Top}(i)$ and then rotate C_{L,ε_s} and C_{R,ε_s} to be packed on top. These items can be packed into height $(1 - \mathit{top}(i) + 2\varepsilon_s)N \leq 5\varepsilon_s N \leq (1-\varepsilon_s)N$ if ε_s is small enough. On the other hand, as there is no massive item, $\mathscr{R} \setminus \mathscr{R}_1$ can be packed using height $(1-\varepsilon_s)N$ and width N.

Case 2C: Item i intersects only S_{L,ε_s} (symmetrically only S_{R,ε_s}).
Consider first the case $i \in D_{L,\varepsilon_s}$. If right$(i) \leq N/2$, let $\mathscr{R}_1 := \mathit{Right}(i)$ which can be packed into a box of width $(1-\varepsilon_s)N$ and height N. On the other hand, we can pack $\mathscr{R} \setminus \{\mathscr{R}_1 \cup C_{T,3\varepsilon_s} \cup C_{B,\varepsilon_s}\}$ in its original position and $C_{T,3\varepsilon_s} \cup C_{B,\varepsilon_s}$ on its side by rotating them. The total width of this packing is at most $(1/2 + 3\varepsilon_s + \varepsilon_s)N \leq (1-\varepsilon_s)N$ if ε_s is small enough.
On the other hand, if right$(i) > N/2$, let $\mathscr{R}_1 := \mathit{Left}(i) \cup i$ which can clearly be packed into a reduced knapsack as i does not intersect S_{R,ε_s}. Now, consider $\mathscr{R} \setminus \{\mathscr{R}_1 \cup C_{T,3\varepsilon_s} \cup C_{B,\varepsilon_s}\}$, rotate $C_{T,3\varepsilon_s} \cup C_{B,\varepsilon_s}$ and pack them on its left. The total width of this packing is at most $(1/2 + 4\varepsilon_s)N \leq (1-\varepsilon_s)N$.
Finally, if the previous case does not hold, then every item intersecting both $S_{B,\varepsilon_s} \cap S_{T,3\varepsilon_s}$ is completely contained in S_{L,ε_s}. Thus, we can consider $\mathscr{R}_1 = E_{T,3\varepsilon_s} \setminus (C_{L,\varepsilon_s} \cup C_{R,\varepsilon_s})$ which can be packed in the reduced knapsack as items do not intersect S_{B,ε_s}. In order to pack $\mathscr{R} \setminus \mathscr{R}_1$ we can rotate C_{L,ε_s} and C_{R,ε_s} and pack them on top of $\mathscr{R} \setminus (\mathscr{R}_1 \cup C_{L,\varepsilon_s} \cup C_{R,\varepsilon_s})$ as in Case 1.

This concludes the proof of Lemma 77.

4.6 Open Problems

The most important open question here is to close the gap for the approximability of all the considered cases. A way to make progress in this direction is to generalize our techniques in order to handle multiple L-packing instances simultaneously, or also to handle ring-shaped instances of the problem for items with one large dimension and the remaining one small (similar to the L-case described before). A way to handle these special cases would lead to improved approximation factors in the settings without rotations. On the other hand, finding a way to handle L-packing instances when rotations are allowed is also an interesting and important open question.

We can also consider parameterized algorithms for Geometric Knapsack. To the best of our knowledge, the only result in this direction is a recent PTAS in FPT time (considering the number of items in the optimal solution as parameter) for the case with rotations and uniform weights, together with a hardness result for all the cases (Grandoni et al. [2019]). Developing improved parameterized approximation algorithms for the remaining cases may help to devise new algorithmic approaches to tackle the general problem.

Another interesting related problem is the geometric variant of the *Unsplittable Flow on a Path problem (UFP)*, known as the *Storage Allocation problem (SAP)*. In this problem we are given a strip of integral width W and a set of n items $\{1, \dots, n\}$, each one characterized by an integral height $h(i)$, an integral starting x-coordinate $s(i)$ and an integral ending x-coordinate $t(i)$ satisfying $0 \le s(i) < t(i) \le W$, plus an integral profit $p(i)$. We are also given a *capacity profile* which in some sense defines a “knapsack” where to assign a maximum subset of items. More formally, we look at the strip as a path with W edges, where each edge e is equipped with an integral capacity $c(e)$. The goal is to select a subset of the items of maximum profit and find a packing of them (i.e. deciding vertical positions $bottom(i)$ for each selected item i) such that the starting and ending coordinates of each item are respected, they form a feasible non-overlapping packing and for each edge e of the path and item i containing that edge we have that $top(i) \le c(e)$ (in other words, it is similar to two-dimensional Geometric Knapsack, but the horizontal position of each item is fixed and the knapsack is not necessarily a square, it is defined by the capacity profile). This problem is known to be NP-hard and the best known approximation factor for it is $(2 + \varepsilon)$ (Mömke and Wiese [2015]).

Appendix A

Two-Dimensional Geometric Knapsack with Resource Augmentation

In this section we prove that it is possible to pack a high profit subset of items into containers, if we are allowed to augment one side of a knapsack by a small fraction.

The result was essentially proved by Jansen and Solis-Oba [2009], although we introduced some modifications and extensions to obtain additional useful features. For the sake of completeness, we provide a full proof, which follows in spirit the proof of the original result, from which we also borrow most of the notation.

We say that a container C' is smaller than a container C if $w(C') \leq w(C)$ and $h(C') \leq h(C)$. Given a container C and a positive $\varepsilon < 1$, we say that an item R_j is ε-small for C if $w(R_j) \leq \varepsilon w(C)$ and $h(R_j) \leq \varepsilon h(C)$.

Lemma 79 (Resource Augmentation Packing Lemma). *Let $\mathcal{R}'$ be a collection of items that can be packed into a box of size $a \times b$, and $\varepsilon_{ra} > 0$ be a given constant. Then there exists a container packing of $\mathcal{R}'' \subseteq \mathcal{R}'$ inside a box of size $a \times (1+\varepsilon_{ra})b$ (resp., $(1+\varepsilon_{ra})a \times b$) such that:*

1. *$p(\mathcal{R}'') \geq (1 - O(\varepsilon_{ra}))p(\mathcal{R}')$;*
2. *the number of containers is $O_{\varepsilon_{ra}}(1)$ and their sizes belong to a set of cardinality $n^{O_{\varepsilon_{ra}}(1)}$ that can be computed in polynomial time;*
3. *the total area of the containers is at most $a(\mathcal{R}') + \varepsilon_{ra}ab$.*

Note that we do not allow rotations, that is, items are packed with the same orientation as in the original packing. However, as an existential result we can apply it also to the case with rotations. Moreover, since Theorem 16 gives a PTAS for approximating container packings, this implies a simple algorithm that does not need to solve any LP to find the solution, in both the cases with and without rotations.

For simplicity, in this section we assume that widths and heights are positive real numbers in $(0,1]$, and $a = b = 1$: in fact, all elements, containers and boxes can be scaled without affecting the property of a packing of being a *container packing* with the above conditions. Thus, without loss of generality, we prove the statement for the augmented $1 \times (1 + \varepsilon_{ra})$ box.

Let $\varepsilon'_{ra} = \varepsilon_{ra}/2 < \varepsilon_{ra}$. We will first obtain a packing where all the elements of each area container C are smaller than the dimensions of C by a factor ε'_{ra}, and in Section A.3 we will obtain the final packing, where the sizes of each container are taken from a polynomially sized set of choices.

We will use the following Lemma, that follows from the analysis in Kenyon and Rémila [2000].

Lemma 80 (Kenyon and Rémila [2000])**.** *Let $\overline{\varepsilon} > 0$, and let $\mathscr{Q}$ be a set of rectangles, each of height and width at most* 1*. Let $\mathscr{L} \subseteq \mathscr{Q}$ be the set of rectangles of width at least $\overline{\varepsilon}$, and let $OPT_{SP}(\mathscr{L})$ be the minimum width such that the rectangles in $\mathscr{L}$ can be packed in a box of size $OPT_{SP}(\mathscr{L}) \times 1$.*

Then $\mathscr{Q}$ can be packed in polynomial time into a box of height 1 *and width $\tilde{w} \leq \max\{OPT_{SP}(\mathscr{L}) + \frac{18}{\overline{\varepsilon}^2} w_{\max}, a(\mathscr{Q})(1 + \overline{\varepsilon}) + \frac{19}{\overline{\varepsilon}^2} w_{\max}\}$. Furthermore, all the rectangles with both width and height less than $\overline{\varepsilon}$ are packed into at most $\frac{9}{\overline{\varepsilon}^2}$ boxes, and all the remaining rectangles into at most $\frac{27}{\overline{\varepsilon}^3}$ vertical containers.*

Note that the boxes containing the rectangles that are smaller than $\overline{\varepsilon}$ are not necessarily packed as containers.

We need the following technical lemma to create a gap on the sizes of the items analogously to Lemmas 26 and 72.

Lemma 81. *Let $\varepsilon > 0$ and let $f(\cdot)$ be any positive increasing function such that $f(x) < x$ for all x. Then, there exist positive constant values $\delta, \mu \in \Omega_\varepsilon(1)$, with $f(\varepsilon) \geq \delta$ and $f(\delta) \geq \mu$ such that the total profit of all the items whose width or height lies in $(\mu, \delta]$ is at most $\varepsilon \cdot p(\mathscr{R}')$.*

Proof. Define $k + 1 = 2/\varepsilon + 1$ constants $\varepsilon_1, \ldots, \varepsilon_{k+1}$, with $\varepsilon_1 = f(\varepsilon)$ and $\varepsilon_i = f(\varepsilon_{i+1})$ for each i. Consider the k ranges of widths and heights of type $(\varepsilon_{i+1}, \varepsilon_i]$. By an averaging argument there exists one index j such that the total profit of the items in $\mathscr{R}'$ with at least one side length in the range $(\varepsilon_{j+1}N, \varepsilon_j N]$ is at most $2\frac{\varepsilon}{2}p(\mathscr{R}')$. It is then sufficient to set $\delta = \varepsilon_j$ and $\mu = \varepsilon_{j+1}$. □

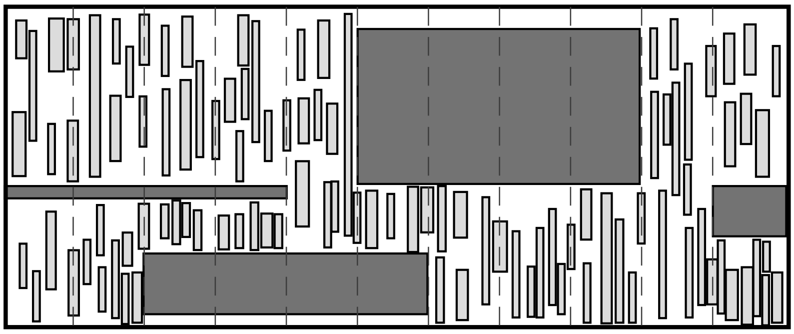


Figure A.1. An example of a packing after the short-narrow items have been removed, and the wide items (in dark grey) have been aligned to the M vertical strips. Note that the short-high items (in light gray) are much smaller than the vertical strips.

We use this lemma with $\varepsilon = \varepsilon'_{ra}$, and we will specify the function f later. By properly choosing the function f, in fact, we can enforce constraints on the value of μ with respect to δ, which will be useful several times; the exact constraints will be clear from the analysis. Thus, we remove from $\mathscr{R}'$ the items that have at least one side length in $(\mu, \delta]$.

We call an item R *wide* if $w(R) > \delta$, *high* if $h(R) > \delta$, *short* if $w(R) \leq \mu$ and *narrow* if $h(R) \leq \mu$.[1] From now on, we will assume that we start with the optimal packing of the items in $\mathscr{R}'$, and we will modify it until we obtain a packing with the desired properties. We remove from $\mathscr{R}'$ all the short-narrow items, initially obtaining a packing. We will show in section A.4 how to use the residual space to pack them, with a negligible loss of profit.

As a first step, we round up the widths of all the *wide* items in $\mathscr{R}'$ to the nearest multiple of δ^2; moreover, we shift them horizontally so that their starting coordinate is an integer multiple of δ^2 (note that, in this process, we might have to shift also the other items in order to make space). Since the width of each wide item is at least δ and $\frac{1}{\delta} \cdot 2\delta^2 = 2\delta$, it is easy to see that it is sufficient to increase the width of the box to $1 + 2\delta$ to perform such a rounding.

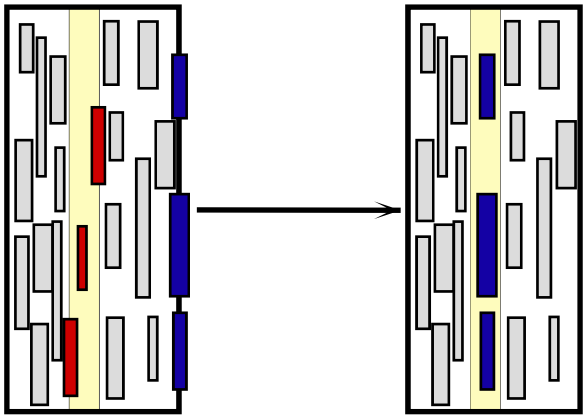


Figure A.2. For each vertical box, we can remove a low profit subset of items (red in the picture), to make space for short-high items that cross the right edge of the box (blue).

A.1 Containers for Short-High Items

We draw vertical lines across the $1 \times (1+2\delta)$ region spaced by δ^2, splitting it into $M := \frac{1+2\delta}{\delta^2}$ vertical strips (see Figure A.1). Consider each maximal rectangular region which is contained in one such strip and does not overlap any wide item; we define a box for each such region that contains at least one short-high item, and we denote the set of such boxes by $\mathcal{B}$.

Note that some short items might intersect the vertical edges of the boxes, but in this case they overlap with exactly two boxes. Using a standard shifting technique, we can assume that no item is cut by the boxes by losing profit at most $\varepsilon'_{ra}OPT$: first, we assume that the items intersecting two boxes belong to the leftmost of those boxes. For each box $B \in \mathcal{B}$ (which has width δ^2 by definition), we divide it into vertical strips of width μ. Since there are $\frac{\delta^2}{\mu} > 2/\varepsilon'_{ra}$ strips and each item overlaps with at most 2 such strips, there must exist one of them such that the profit of the items intersecting it is at most $2\mu p(B) \leq \varepsilon'_{ra}p(B)$, where $p(B)$ is the profit of all the items that are contained in or belong to B. We can remove all the items overlapping such strip, creating in B an empty vertical gap of width μ, and then we can move all the items intersecting the right boundary of B to the empty space (see Figure A.2).

Remark 82. *The number of boxes in $\mathcal{B}$ is at most $\frac{1+2\delta}{\delta^2} \cdot \frac{1}{\delta} \leq \frac{2}{\delta^3}$.*

[1]Note that the classification of the items in this section is different from the ones used in the main results of this thesis, although similar in spirit.

First, by a shifting argument similar to above, we can reduce the width of each box to $\delta^2 - \delta^4$ while losing only an ε'_{ra} fraction of the profit of the items in B. Then, for each $B \in \mathscr{B}$, since the maximum width of the items in B is at most μ, by applying Lemma 80 with $\overline{\varepsilon} = \delta^2/2$ we obtain that the items packed inside B can be repacked into a box B' of height $h(B)$ and width at most $\max\{\delta^2 - \delta^4 + \frac{72}{\delta^4}\mu, (\delta^2 - \delta^4)(1 + \frac{\delta^2}{2}) + \frac{76}{\delta^4}\mu\} \leq \delta^2$, which is true if we make sure that $\mu \leq \delta^{10}/76$. Furthermore, the short-high items in B are packed into at most $\dfrac{216}{\delta^6} \leq \dfrac{1}{\delta^7}$ vertical containers, assuming without loss of generality that $\delta \leq 1/216$. Note that all the items are packed into vertical containers, because items that have both width and height smaller than $\overline{\varepsilon}$ are short-narrow and we already removed them. Summarizing:

Proposition 83. *There is a set $\mathscr{R}^+ \subseteq \mathscr{R}'$ of items with total profit at least $(1 - O(\varepsilon'_{ra})) \cdot p(\mathscr{R}')$ and a corresponding packing for them in a $1 \times (1 + 2\delta)$ region such that:*

- *every wide items in $\mathscr{R}^+$ has its length rounded up to the nearest multiple of δ^2 and it is positioned so that its left side is at a position x which is a multiple of δ^2, and*
- *each box $B \in \mathscr{B}$ storing at least one short-high item has width δ^2, and the items inside are packed into at most $1/\delta^7$ vertical containers.*

A.2 Fractional Packing with $O(1)$ Containers

Let us consider now the set of items $\mathscr{R}^+$ and an almost optimal packing S^+ for them according to Proposition 83. We remove the items assigned to boxes in $\mathscr{B}$ and consider each box $B \in \mathscr{B}$ as a single pseudoitem. Thus, in the new almost optimal solution there are just pseudoitems from $\mathscr{B}$ and wide items with right and left coordinates that are multiples of δ^2. We will now show that we can derive a fractional packing with the same profit, and such that the items and pseudoitems can be (fractionally) assigned to a constant number of containers. By *fractional packing* we mean a packing where horizontal items are allowed to be sliced horizontally (but not vertically); we can think of the profit as being split proportionally to the heights of the slices.

Let $\mathscr{K}$ be a subset of the horizontal items of size K that will be specified later. By extending horizontally the top and bottom edges of the items in $\mathscr{K}$ and the

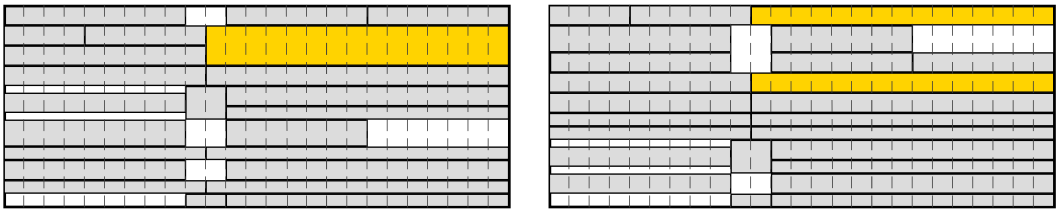


Figure A.3. Rearranging the items in a horizontal stripe. On the right, items are repacked so that regions with the same configuration appear next to each other. Note that the yellow item has been sliced, since it partakes in two regions with different configurations.

pseudoitems in $\mathscr{B}$, we partition the knapsack into at most $2(|K| + |\mathscr{B}|) + 1 \leq 2(K + \frac{2}{\delta^3}) + 1 \leq 2(K + \frac{3}{\delta^3})$ horizontal stripes.

Let us focus on the (possibly sliced) items contained in one such stripe of height h. For any vertical coordinate $y \in [0, h]$ we can define the *configuration* at coordinate y as the set of positions where the horizontal line at distance y from the bottom cuts a vertical edge of an horizontal item which is not in $\mathscr{K}$. There are at most 2^{M-1} possible configurations in a stripe.

We can further partition the stripe into maximal contiguous regions with the same configuration. Note that the number of such regions is not bounded, since configurations can be repeated. But since the items are allowed to be sliced, we can rearrange the regions so that all the ones with the same configuration appear next to each other; see Figure A.3 for an example. After this step is completed, we define up to M horizontal containers per each configuration, where we repack the sliced horizontal items. Clearly, all sliced items are repacked.

Thus, the number of horizontal containers that we defined per each stripe is bounded by $M2^{M-1}$, and the total number overall is at most

$$2\left(K + \frac{3}{\delta^3}\right)M2^{M-1} = \left(K + \frac{3}{\delta^3}\right)M2^M.$$

A.2.1 Existence of an Integral Packing

We will now show the existence of an integral packing, at a small loss of profit. This is similar to the proof of Lemma 17 with the exception that now the profit of the discarded items comes into account.

Consider a fractional packing in N containers. Since each item slice is packed in a container of exactly the same width, it is possible to pack all but at most N items integrally by a simple greedy algorithm: choose a container, and greedily pack in it items of the same width, until either there are no items left for that width, or the next item does not fit in the current container. In this case, we discard this item and close the container, meaning that we do not use it further. Clearly, only one item per container is discarded, and no item is left unpacked.

The only problem is that the total profit of the discarded items can be large. To solve this problem, we use the following shifting argument. Let $\mathcal{K}_0 = \emptyset$ and $K_0 = 0$. For convenience, let us define $f(K) = \left(K + \frac{3}{\delta^3}\right)M2^M$.

First, consider the fractional packing obtained by choosing $\mathcal{K} = \mathcal{K}_0$, so that $K = K_0 = 0$. Let $\mathcal{K}_1$ be the set of discarded items obtained by the greedy algorithm, and let $K_1 = |\mathcal{K}_1|$. Clearly, by the above reasoning, the number of discarded items is bounded by $f(K_0)$. If the profit $p(\mathcal{K}_1)$ of the discarded items is at most $\varepsilon'_{ra}p(OPT)$, then we remove them and there is nothing else to prove. Otherwise, consider the fractional packing obtained by fixing $\mathcal{K} = \mathcal{K}_0 \cup \mathcal{K}_1$. Again, we will obtain a set $\mathcal{K}_2$ of discarded items such that $K_2 := |\mathcal{K}_2| \leq f(K_0 + K_1)$. Since the sets $\mathcal{K}_1, \mathcal{K}_2, \ldots$ that we obtain are all disjoint, the process must stop after at most $1/\varepsilon'_{ra}$ iterations. Setting $p := M2^M$ and $q := \frac{3}{\delta^3}M2^M$, we have that $K_{i+1} \leq p(K_0 + K_1 + \ldots K_i) + q$ for each $i \geq 0$. Crudely bounding it as $K_{i+1} \leq i \cdot pq \cdot K_i$, we immediately obtain that $K_i \leq (pq)^i$. Thus, in the successful iteration, the size of $\mathcal{K}$ is at most $K_{1/\varepsilon'_{ra}-1}$ and the number of containers is at most $K_{1/\varepsilon'_{ra}} \leq (pq)^{1/\varepsilon'_{ra}} = (\frac{3}{\delta^2}M^2 2^{2M})^{1/\varepsilon'_{ra}} = O_{\varepsilon'_{ra},\delta}(1)$.

A.3 Rounding Down Horizontal and Vertical Containers

As per the above analysis, the total number of horizontal containers is at most $(\frac{3}{\delta^2}M^2 2^{2M})^{\varepsilon'_{ra}}$ and the total number of vertical containers is at most $\frac{2}{\delta^3} \cdot \frac{1}{\delta^7} = \frac{2}{\delta^{10}}$.

We will now show that, at a small loss of profit, it is possible to replace each horizontal and each vertical container defined so far with a constant number of smaller containers, so that the total area of the new containers is at most as big as the total area of the items originally packed in the container. Note that in each container we consider the items with the original widths (not rounded up). We use the following lemma:

Lemma 84. *Let C be a horizontal (resp. vertical) container defined above, and let $\mathcal{R}_C$ be the set of items packed in C. Then, it is possible to pack a set $\mathcal{R}'_C \subseteq \mathcal{R}_C$ of profit at least $(1-3\varepsilon'_{ra})p(\mathcal{R}_C)$ in a set of at most $\left\lceil \log_{1+\varepsilon'_{ra}}(\frac{1}{\delta}) \right\rceil / \varepsilon'^2_{ra}$ horizontal (resp. vertical) containers that can be packed inside C and such that their total area is at most $a(\mathcal{R}_C)$.*

Proof. Without loss of generality, we prove the result only for the case of a horizontal container.

Since $w(R_i) \geq \delta$ for each item $R_i \in \mathscr{R}_C$, we can partition the items in $\mathscr{R}_C$ into at most $\left\lceil \log_{1+\varepsilon'_{ra}}(\frac{1}{\delta}) \right\rceil$ groups $\mathscr{R}_1, \mathscr{R}_2, \ldots$, so that in each $\mathscr{R}_j$ the widest item has width bigger than the smallest by a factor at most $1 + \varepsilon'_{ra}$; we can then define a container C_j for each group $\mathscr{R}_j$ that has the width of the widest item it contains and height equal to the sum of the heights of the contained items.

Consider now one such C_j and the set of items $\mathscr{R}_j$ that it contains, and let $P := p(\mathscr{R}_j)$. Clearly, $w(C_j) \leq (1+\varepsilon'_{ra})w(R_i)$ for each $R_i \in \mathscr{R}_j$, and so $a(C_j) \leq (1+\varepsilon'_{ra})a(\mathscr{R}_j)$. If all the items in $\mathscr{R}_j$ have height at most $\varepsilon'_{ra}h(C_j)$, then we can remove a set of items with total height at least $\varepsilon'_{ra}h(C)$ and profit at most $2\varepsilon'_{ra}p(\mathscr{R}_j)$. Otherwise, let $\mathscr{Q}$ be the set of items of height larger than $\varepsilon'_{ra}h(C_j)$, and note that $a(Q) \geq \varepsilon'_{ra}h(C_j)w(C_j)/(1+\varepsilon'_{ra})$. If $p(\mathscr{Q}) \leq \varepsilon'_{ra}P$, then we remove the items in $\mathscr{Q}$ from the container C_j and reduce its height as much as possible, obtaining a smaller container C'_j; since $a(C'_j) \leq a(C_j) - \varepsilon'_{ra}a(C_j) = (1-\varepsilon'_{ra})a(C_j) \leq (1-\varepsilon'_{ra})(1+\varepsilon'_{ra})a(\mathscr{R}_j) < a(\mathscr{R}_j)$, then the proof is finished. Otherwise, we define one container for each of the items in $\mathscr{Q}$ (which are at most $1/\varepsilon'_{ra}$) of exactly the same size, and we still shrink the container with the remaining items as before; note that there is no lost area for each of the newly defined container. Since at every non-terminating iteration a set of items with profit larger than $\varepsilon'_{ra}P$ is removed, the process must end within $1/\varepsilon'_{ra}$ iterations.

Note that the total number of containers that we produce for each initial container C_j is at most $1/\varepsilon'^2_{ra}$, and this concludes the proof. □

Thus, by applying the above lemma to each horizontal and each vertical container, we obtain a modified packing where the total area of the horizontal and vertical containers is at most the area of the items of $\mathscr{R}'$ (without the short-narrow items, which we will take into account in the next subsection), while the number of containers increases at most by a factor $\left\lceil \log_{1+\varepsilon'_{ra}}(\frac{1}{\delta}) \right\rceil / \varepsilon'^2_{ra}$.

A.4 Packing Short-Narrow Items

This is similar to the procedure shown in Section 3.2.4 but we need to provide extra area guarantees in this case.

Consider the integral packing obtained from the previous subsection, which has at most $K' := \left(\frac{2}{\delta^{10}} + (\frac{3}{\delta^2}M^2 2^{2M})^{\varepsilon'_{ra}}\right)\left\lceil \log_{1+\varepsilon'_{ra}}(\frac{1}{\delta}) \right\rceil / \varepsilon'^2_{ra}$ containers. We can create a non-uniform grid extending each side of the containers until they hit another container or the boundary of the knapsack. Moreover, we also add horizontal and vertical lines spaced at distance ε'_{ra}. We call *free cell* each face defined by the above lines that does not overlap a container of the packing; by construction, no free cell has a side bigger than ε'_{ra}. The number of free cells in this grid plus the existing containers is bounded by $K_{TOTAL} = (2K' + 1/\varepsilon'_{ra})^2 = O_{\varepsilon'_{ra},\delta}(1)$. We crucially use the fact that this number does not depend on value of μ.

Note that the total area of the free cells is no less than the total area of the short-narrow items, as a consequence of the guarantees on the area of the containers introduced so far. We will pack the short-narrow items into the free cells of this grid using NFDH, but we only use cells that have width and height at least $\frac{8\mu}{\varepsilon'_{ra}}$; thus, each short-narrow item will be assigned to a cell whose width (resp. height) is larger by at least a factor $8/\varepsilon'_{ra}$ than the width (resp. height) of the item. Each discarded cell has area at most $\frac{8\mu}{\varepsilon'_{ra}}$, which implies that the total area of discarded cells is at most $\frac{8\mu K_{TOTAL}}{\varepsilon'_{ra}}$. Now we consider the selected cells in an arbitrary order and pack short narrow items into them using NFDH, defining a new area container for each cell that is used. Thanks to Lemma **??**, we know that each new container C (except maybe the last one) that is used by NFDH contains items for a total area of at least $(1-\varepsilon'_{ra}/4)a(C)$. Thus, if all items are packed, we remove the last container opened by NFDH, and we call S the set of items inside, that we will repack elsewhere; note that $a(S) \leq \varepsilon'^2_{ra} \leq \varepsilon'_{ra}/3$, since all the items in S were packed in a free cell. Instead, if not all items are packed by NFDH, let S be the residual items. In this case, the area of the unpacked items is $a(S) \leq \frac{8\mu K_{TOTAL}}{\varepsilon'_{ra}} + \varepsilon'_{ra}/4 \leq \varepsilon'_{ra}/3$, assuming that $\mu \leq \frac{\varepsilon'^2_{ra}}{96K_{TOTAL}}$.

In order to repack the items of S, we define a new area container C_S of height 1 and width $\varepsilon'_{ra}/2$. Since $a(C_S) = \varepsilon'_{ra}/2 \geq (\varepsilon'_{ra}/3)/(1-2\varepsilon'_{ra})$, all elements from S are packed in C_S by NFDH, and the container can be added to the knapsack by further enlarging its width from $1+2\delta$ to $1+2\delta+\varepsilon'_{ra}/2 < 1+\varepsilon'_{ra}$.

The last required step is to guarantee the necessary constraint on the total area of the area containers, similarly to what was done in Section A.3 for the horizontal and vertical containers.

Let D be any full area container (that is, any area container except for C_S). We know that the area of the items R_D in D is $a(R_D) \geq (1-\varepsilon'_{ra})a(D)$, since each item R_i inside D has width less than $\varepsilon'_{ra}w(D)/2$ and height less than $\varepsilon'_{ra}h(D)/2$, by construction. We remove items from R_D in non-decreasing order of profit/area ratio, until the total area of the residual items is between $(1-4\varepsilon'_{ra})a(D)$ and $(1-3\varepsilon'_{ra})a(D)$ (this is possible, since each element has area at most $\varepsilon'^2_{ra}a(D)$); let R'_D be the resulting set. We have that $p(R'_D) \geq (1-4\varepsilon'_{ra})p(R_D)$, due to the greedy choice. Let us define a container D' of width $w(D)$ and height $(1-\varepsilon'_{ra})h(D)$. It is easy to verify that each item in R_D has width (resp. height) at most $\varepsilon'_{ra}w(D')$ (resp.

$\varepsilon'_{ra}h(D'))$. Moreover, since $a(R'_D) \leq (1-3\varepsilon'_{ra})a(D) \leq (1-2\varepsilon'_{ra})(1-\varepsilon'_{ra})a(C) \leq (1-2\varepsilon'_{ra})a(C')$, then all elements in R'_D are packed in D'. By applying this reasoning to each area container (except C_S), we obtain property (3) of Lemma 79.

Note that the constraints on μ and δ that we imposed are $\mu \leq \frac{\delta^{10}}{76}$ (from Section A.1), and $\mu \leq \frac{\varepsilon'^2_{ra}}{96K_{TOTAL}}$. It is easy to check that both of them are satisfied if we choose $f(x) = (\varepsilon'_{ra}x)^C$ for a big enough constant C that depends only on δ and ε'_{ra}.

Appendix B

Proof of Lemma 59 without Extra Assumptions

We remove now the assumption that we can drop $O_\varepsilon(1)$ items from OPT. We will add a couple of shifting steps to the argumentation above to prove Lemma 59 without that assumption.

It is no longer true that we can neglect the large items OPT_{large} since they might contribute a large amount towards the objective, even though their total number is guaranteed to be small. Also, in the process of constructing the boxes, we killed up to $O_\varepsilon(1)$ items (the items in OPT_{kill}). Similarly, we can no longer drop the constantly many items in OPT_{corr}^{cross}. Therefore, we apply some careful shifting arguments in order to ensure that we can still use a similar construction as above, while losing only a factor $1 + O(\varepsilon)$ due to some items that we will discard.

The general idea is as follows: For $t = 0, \ldots, k$ (we will later argue that $k < 1/\varepsilon$), we define disjoint sets $K(t)$ recursively, each containing at most $O_\varepsilon(1)$ items. Each set $\mathscr{K}(t) = \bigcup_{j=0}^{t} K(j)$ is used to define a grid $G(t)$ in the knapsack. Based on an item classification that depends on this grid, we identify a set of skewed items and create a corridor partition w.r.t. these skewed items as described in Lemma 44. We then create a partition of the knapsack into corridors and a constant (depending on ε) number of containers (see Section B.1). Next, we decompose the corridors into boxes (Section B.2) and these boxes into containers (section B.3) similarly as we did in Section 4.4.2 (but with some notable changes as we did not delete small items from the knapsack and thus need to handle those as well). In the last step, we add small items to the packing (Section B.4). During this whole process, we define the set $K(i + 1)$ of items that are somehow "in our way" during the decomposition process (e.g., items that are not fully contained in some corridor of the corridor partition), but which we cannot delete directly as they might have large profit. These items are similar to the killed items in the previous argumentation. However, using a shifting argument we can simply show that after at most $k < 1/\varepsilon$ steps of this process, we encounter a set $K(k)$ of low total profit, that we can remove, at which point we can apply almost the same argumentation as in Lemmas 65, 66, and 67 to obtain lower bounds on the profit of an optimal L&C packing (Section B.5).

We initiate this iterative process as follows: Denote by $K(0)$ a set containing all items that are killed in at least one of the cases arising in Section 4.4.2 (the set OPT_K in that section) and additionally the large items OPT_{large} and the $O_\varepsilon(1)$ items in OPT_{corr}^{cross} (in fact $OPT_{large} \subseteq OPT_{corr}^{cross}$). Note that $|K(0)| \leq O_\varepsilon(1)$. If $p(K(0)) \leq \varepsilon \cdot p(OPT)$ then we can simply remove these items (losing only a factor of $1+\varepsilon$) and then apply the remaining argumentation exactly as above and we are done. Therefore, from now on suppose that $p(K(0)) > \varepsilon \cdot p(OPT)$.

B.1 Definition of Grid and Corridor Partition

Assume we are in round t of this process, i.e., we defined $K(t)$ in the previous step (unless $t = 0$, then $K(t)$ is defined as specified above) and assume that $p(K(t)) > \varepsilon OPT$ (otherwise, see Section B.5). We are now going to define the non-uniform grid $G(t)$ and the induced partition of the knapsack into a collection of cells $\mathscr{C}_t$. The x-coordinates (y-coordinates) of the grid cells are the x-coordinates (y-coordinates, respectively) of the items in $\mathscr{K}(t)$. This yields a partition of the knapsack into $O_\varepsilon(1)$ rectangular cells, such that each item of $\mathscr{K}(t)$ covers one or multiple cells. Note that an item might intersect many cells.

Similarly as above, we define constants $1 \geq \varepsilon_{large} \geq \varepsilon_{small} \geq \Omega_\varepsilon(1)$ and apply a shifting step such that we can assume that for each item $i \in OPT$ touching some cell C we have that $w(i \cap C) \in (0, \varepsilon_{small} w(C)] \cup (\varepsilon_{large} w(C), w(C)]$ and $h(i \cap C) \in (0, \varepsilon_{small} h(C)] \cup (\varepsilon_{large} h(C), h(C)]$, where $h(C)$ and $w(C)$ denote the height and the width of the cell C and $w(i \cap C)$ and $h(i \cap C)$ denote the height and the width of the intersection of the item i with the cell C, respectively. For a cell C denote by $OPT(C)$ the set of items that intersect C in OPT. We obtain a partition of $OPT(C)$ into $OPT_{small}(C)$, $OPT_{large}(C)$, $OPT_{hor}(C)$, and $OPT_{ver}(C)$:

- $OPT_{small}(C)$ contains all items $i \in OPT(C)$ with $h(i \cap C) \leq \varepsilon_{small} h(C)$ and $w(i \cap C) \leq \varepsilon_{small} w(C)$,
- $OPT_{large}(C)$ contains all items $i \in OPT(C)$ with $h(i \cap C) > \varepsilon_{large} h(C)$ and $w(i \cap C) > \varepsilon_{large} w(C)$,

- $OPT_{hor}(C)$ contains all items $i \in OPT(C)$ with $h(i \cap C) \leq \varepsilon_{small} h(C)$ and $w(i \cap C) > \varepsilon_{large} w(C)$, and
- $OPT_{ver}(C)$ contains all items $i \in OPT(C)$ with $h(i \cap C) > \varepsilon_{large} h(C)$ and $w(i \cap C) \leq \varepsilon_{small} w(C)$.

We call an item i *intermediate* if there is a cell C such that $w(i \cap C) \in (\varepsilon_{small} w(C), \varepsilon_{large} w(C)]$ or $h(i \cap C) \in (\varepsilon_{small} w(C), \varepsilon_{large} w(C)]$. Note that an item i is intermediate if and only if the last condition is satisfied for one of the at most four cells that contain a corner of i. As in Section 4.1.1 it is possible to define ε_{small} and ε_{large} so that intermediate items can be discarded at negligible cost.

For each cell C that is not entirely covered by some item in $K(t)$ we add all items in $OPT_{large}(C)$ that are not contained in $\mathcal{K}(t)$ to $K(t+1)$. Note that here, in contrast to before, we do *not* remove small items from the packing but keep them.

Based on the items $OPT_{skew}(\mathcal{C}_t) := \cup_{C \in \mathcal{C}_t} OPT_{hor}(C) \cup OPT_{ver}(C)$ we create a corridor decomposition and consequently a box decomposition of the knapsack. To make this decomposition clearer, we assume that we first stretch the non-uniform grid into a uniform $[0,1] \times [0,1]$ grid. After this operation, for each cell C and for each element of $OPT_{hor}(C) \cup OPT_{ver}(C)$ we know that its height or width is at least $\varepsilon_{large} \cdot \frac{1}{1+2|\mathcal{K}(t)|}$. We apply Lemma 44 on the set $OPT_{skew}(\mathcal{C}_t)$ which yields a decomposition of the $[0,1] \times [0,1]$ square into at most $O_{\varepsilon, \varepsilon_{large}, \mathcal{K}(t)}(1) = O_{\varepsilon, \varepsilon_{large}}(1)$ corridors. The decomposition for the stretched $[0,1] \times [0,1]$ square corresponds to the decomposition for the original knapsack, with the same items being intersected. Since we can assume that all items of OPT are placed within the knapsack so that they have integer coordinates, we can assume that the corridors of the decomposition also have integer coordinates. We can do that, because shifting the edges of the decomposition to the closest integral coordinate will not make the decomposition worse, i.e., no new items of OPT will be intersected.

We add all items in $OPT_{skew}(\mathcal{C}_t)$ that are not contained in a corridor (at most $O_\varepsilon(1)$ many) and that are not contained in $\mathcal{K}(t)$ to $K(t+1)$. The corridor partition has the following useful property: we started with a fixed (optimal) solution OPT for the overall problem with a *fixed placement* of the items in this solution. Then we considered the items in $OPT_{skew}(\mathcal{C}_t)$ and obtained the sets $OPT_{corr} \subseteq OPT_{skew}(\mathcal{C}_t)$ and $OPT_{corr}^{cross} \subseteq OPT_{corr}$. With the mentioned fixed placement, apart from the $O_\varepsilon(1)$ items in OPT_{corr}^{cross}, each item in OPT_{corr} is contained in one corridor. In particular, the items in OPT_{corr} do not overlap the items in $\mathcal{K}(t)$. We construct now a partition of the knapsack into $O_\varepsilon(1)$ corridors and $O_\varepsilon(1)$ containers where each container contains exactly one item from $\mathcal{K}(t)$. The main

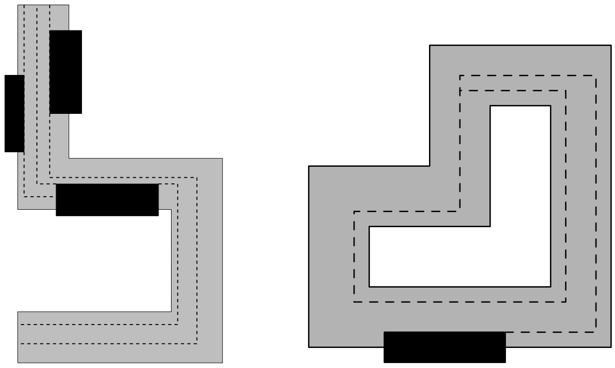


Figure B.1. Circumventing the items in $\mathscr{R}'$, shown in black. The connected components between the dashed lines show the resulting new corridors.

obstacle is that there can be an item $i \in \mathscr{K}(t)$ that overlaps a corridor (see Figure B.1). We solve this problem by applying the following lemma on each such corridor.

Lemma 85. *Let S be an open corridor with $b(S)$ bends. Let $\mathscr{R}' \subseteq OPT$ be a collection of items which intersect the boundary of S with $\mathscr{R}' \cap OPT_{skew}(\mathscr{C}_t) = \emptyset$. Then there is a collection of $|\mathscr{R}'| \cdot b(S)$ line segments $\mathscr{L}$ within S which partition S into corridors with at most $b(S)$ bends each such that no item from $\mathscr{R}'$ is intersected by $\mathscr{L}$ and there are at most $O_\varepsilon(|\mathscr{R}'| \cdot b(S))$ items of $OPT_{skew}(\mathscr{C}_t)$ intersected by line segments in $\mathscr{L}$.*

Proof. Let $i \in \mathscr{R}'$ and assume w.l.o.g. that i lies within a horizontal subcorridor S_i of the corridor S. If the top or bottom edge e of i lies within S_i, we define a horizontal line segment ℓ which contains the edge e and which is maximally long so that it does not intersect the interior of any item in $\mathscr{R}'$, and such that it does not cross the boundary curve between S_i and an adjacent subcorridor, or an edge of the boundary of S (we can assume w.l.o.g. that e does not intersect the boundary curve between S_i and some adjacent subcorridor). We say that ℓ *crosses* a boundary curve c (or an edge e of the boundary of S) if $c \setminus \ell$ (or $e \setminus \ell$) has two connected components.

We now “extend” each end-point of ℓ which does not lie at the boundary of some other item of $\mathscr{R}'$ or at the boundary of S, we call such an end point a *loose end*. For each loose end x of ℓ lying on the boundary curve c_{ij} partitioning the subcorridors S_i and S_j, we introduce a new line ℓ' perpendicular to ℓ, starting at x and crossing the subcorridor S_j such that the end point of ℓ' is maximally far away from x subject to the constraint that ℓ' does not cross an item in $\mathscr{R}'$, another boundary curve inside S, or the boundary of S. We continue iteratively. Since the corridor has $b(S)$ bends, after at most $b(S)$ iterations this operation will finish. We repeat the above operation for every item $i \in \mathcal{R}'$, and we denote by $\mathcal{L}$ the resulting set of line segments, see Figure B.1 for a sketch. Notice that $|\mathcal{L}| = b(S) \cdot |\mathcal{R}'|$. By construction, if an item $i \in OPT_{skew}(\mathcal{C}_t)$ is intersected by a line in $\mathcal{L}$ then it is intersected parallel to its longer edge. Thus, each line segment in $\mathcal{L}$ can intersect at most $O_\varepsilon(1)$ items of $OPT_{skew}(\mathcal{C}_t)$. Thus, in total there are at most $O_{\varepsilon,\varepsilon_{large}}(|\mathcal{R}'| \cdot b(S))$ items of $OPT_{skew}(\mathcal{C}_t)$ intersected by line segments in $\mathcal{L}$. □

We apply Lemma 85 to each open corridor that intersects an item in $\mathcal{K}(t)$. We add all items of $\mathcal{R}_{skew}(\mathcal{C}_t)$ that are intersected by line segments in $\mathcal{L}$ to $K(t+1)$. This adds $O_\varepsilon(1)$ items in total to $K(t+1)$ since $|\mathcal{K}(t)| \in O_\varepsilon(1)$ and $b(S) \leq 1/\varepsilon$ for each corridor S. For closed corridors we have the following analogous lemma.

Lemma 86. *Let S be a closed corridor with $b(S)$ bends. Let $OPT_{skew}(S)$ denote the items in $OPT_{skew}(\mathcal{C}_t)$ that are contained in S. Let $\mathcal{R}' \subseteq OPT$ be a collection of items which intersect the boundary of S with $\mathcal{R}' \cap OPT_{skew}(\mathcal{C}_t) = \emptyset$. Then there is a collection of $O_\varepsilon(|\mathcal{R}'|^2/\varepsilon)$ line segments $\mathcal{L}$ within S which partition S into a collection of closed corridors with at most $1/\varepsilon$ bends each and possibly an open corridor with $b(S)$ bends such that no item from $\mathcal{R}'$ is intersected by $\mathcal{L}$ and there is a set of items $OPT'_{skew}(S) \subseteq OPT_{skew}(S)$ with $|OPT'_{skew}(S)| \leq O_\varepsilon(|\mathcal{R}'|^2)$ such that the items in $OPT_{skew}(S) \setminus OPT'_{skew}(S)$ intersected by line segments in $\mathcal{L}$ have a total profit of at most $O(\varepsilon) \cdot p(OPT_{skew}(\mathcal{C}_t))$.*

Proof. Similarly as for the case of open corridors, we take each item $i \in \mathcal{R}'$ whose edge e is contained in S, and we create a path containing e that partitions S. Here the situation is a bit more complicated, as our newly created paths could extend over more than $\frac{1}{\varepsilon}$ bends inside S. In this case we will have to do some shortcutting, i.e., some items contained in S will be crossed parallel to their shorter edge and we cannot guarantee that their total number will be small. However, we will ensure that the total weight of such items is small. We proceed as follows (see Figure B.1 for a sketch).

Consider any item $i \in \mathcal{R}'$ and assume w.l.o.g. that i intersects a horizontal subcorridor S_i of the closed corridor S. Let e be the edge of i within S_i. For each endpoint of e we create a path p as for the case of closed corridors. If after at most $b(S) \leq 1/\varepsilon$ bends the path hits an item of $\mathcal{R}'$ (possibly the same item i), the boundary of S or another path created earlier, we stop the construction of the path. Otherwise, if p is the first path inside of S that did not finish after at most $b(S)$ bends, we proceed with the construction of the path, only now at each bend we check the total weight of the items of $OPT_{skew}(S)$ that would be crossed parallel to their shorter edge, if, instead of bending, the path would continue at the bend to hit itself. From the construction of the boundary curves in the intersection of two subcorridors we know that for two bends of the constructed path, the sets of items that would be crossed at these bends of the path are pairwise disjoint. Thus, after $O(|\mathscr{R}'|/\varepsilon)$ bends we encounter a collection of items $OPT''_{skew}(S) \subseteq OPT_{skew}(S)$ such that $p(OPT''_{skew}(S)) \leq \frac{\varepsilon}{|\mathscr{R}'|} p(OPT_{skew}(S))$, and we end the path p by crossing the items of $OPT''_{skew}(S)$. This operation creates an open corridor with up to $O(|\mathscr{R}'|/\varepsilon)$ bends. We divide it into up to $O(|\mathscr{R}'|)$ corridors with up to $1/\varepsilon$ bends each. Via a shifting argument we can argue that this loses at most a factor of $1+\varepsilon$ in the profit due to these items. When we perform this operation for each item $i \in \mathscr{R}'$ the total weight of items intersected parallel to their shorter edge (i.e., due to the above shortcutting) is bounded by $|\mathscr{R}'| \cdot \frac{\varepsilon}{|\mathscr{R}'|} p(OPT_{skew}(S)) = \varepsilon \cdot p(OPT_{skew}(S))$. This way, we introduce at most $O(|\mathscr{R}'|^2/\varepsilon)$ line segments. Denote them by $\mathscr{L}$. They intersect at most $O_\varepsilon(|\mathscr{R}'|^2)$ items parallel to their respective longer edge, denote them by $OPT'_{skew}(S)$. Thus, the set $\mathscr{L}$ satisfies the claim of the lemma. □

Similarly as for Lemma 85 we apply Lemma 86 to each closed corridor. We add all items in the respective set $OPT'_{skew}(S)$ to the set $K(t+1)$ which yields $O_\varepsilon(1)$ many items. The items in $OPT_{skew}(S) \setminus OPT'_{skew}(S)$ are removed from the instance, as their total profit is small.

B.2 Partitioning Corridors into Boxes

Then we partition the resulting corridors into boxes according to the different cases described in Section 4.4.2. There is one difference to the argumentation above: we define that the set OPT_{fat} contains not only skewed items contained in the respective subregions of a subcorridor, but *all* items contained in such a subregion. In particular, this includes items that might have been considered as small items above. Thus, when we move items from one subregion to the box associated to the subregion below (see Remark 61) then we move *every* item that is contained in that subregion. If an item is killed in one of the orderings of the subcorridors to apply the procedure from Section 4.4.2 then we add it to $K(t+1)$. Note that $|K(t+1)| \in O_{\varepsilon,\varepsilon_{large},\varepsilon_{box}}(1)$ and $\mathscr{K}(t) \cap K(t+1) = \emptyset$. Also note here that we ignore for the moment small items that cross the boundary curves of the subcorridors; they will be taken care of in Section B.4.

B.3 Partitioning Boxes into Containers

Then we subdivide the boxes into containers. We apply the same decomposition into containers as in the proof of Lemma 69 to each box with a slight modification. Assume that we apply it to a box of size $a \times b$ containing a set of items $\mathcal{R}_{box}$. Like above we first remove the items in a thin strip of width $3\varepsilon b$ such that via a shifting argument the items (fully!) contained in this strip have a small profit of $O(\varepsilon)p(\mathcal{R}_{box})$. However, in contrast to the setting above the set $\mathcal{R}_{box}$ contains not only skewed items but also small items. We call an item i *small* if there is no cell C such that $i \in OPT_{large}(C) \cup OPT_{hor}(C) \cup OPT_{ver}(C)$ and denote by $OPT_{small}(\mathcal{C}_t)$ the set of small items. When we choose the strip to be removed we ensure that the profit of the removed skewed *and* small items is small. There are $O_\varepsilon(1)$ skewed items that partially (but not completely) overlap the strip whose items we remove. We add those $O_\varepsilon(1)$ items to $K(t+1)$. Small items that partially overlap the strip are taken care of later in Section B.4, we ignore them for the moment. Then we apply Lemma 19. In contrast to the setting above, we do not only apply it to the skewed items but apply it also to small items that are contained in the box. Denote by $OPT'_{small}(\mathcal{C}_t)$ the set of small items that are contained in some box of the box partition.

Thus, we obtain an L&C packing for the items in $\mathcal{K}(t)$, for a set of items $OPT'_{skew}(\mathcal{C}_t) \subseteq OPT_{skew}(\mathcal{C}_t)$, and for a set of items $OPT''_{small}(\mathcal{C}_t) \subseteq OPT'_{small}(\mathcal{C}_t)$ such that

$$p(OPT'_{skew}(\mathcal{C}_t))+p(OPT''_{small}(\mathcal{C}_t))+p(K(t+1)) \geq (1-O(\varepsilon))p(OPT_{skew}(\mathcal{C}_t) \cup OPT'_{small}(\mathcal{C}_t)).$$

B.4 Handling Small Items

So far we ignored the small items in $OPT''_{small}(\mathcal{C}_t) := OPT_{small}(\mathcal{C}_t) \setminus OPT'_{small}(\mathcal{C}_t)$. This set consists of small items that in the original packing intersect a line segment of the corridor partition, the boundary of a box, or a boundary curve within a corridor. We describe now how to add them into the empty space of the so far computed packing. First, we assign each item in $OPT''_{small}(\mathcal{C}_t)$ to a grid cell. We assign each small item $i \in OPT''_{small}(\mathcal{C}_t)$ to the cell C such that in the original packing i intersects with C and the area of $i \cap C$ is not smaller than $i \cap C'$ for any cell C' ($i \cap C'$ denotes the part of i intersecting C' in the original packing for any grid cell C').

Consider a grid cell C and let $OPT''_{small}(C)$ denote the small items in $OPT''_{small}(\mathcal{C}_t)$ assigned to C. Intuitively, we want to pack them into the empty space in the cell

C that is not used by any of the containers, similarly as above. The following lemma is an analog of Lemma 69 of the setting above.

Lemma 87. *Let C be a cell. The total area of C occupied by containers is at most $(1-2\varepsilon)a(C)$.*

Proof. In our construction of the boxes we moved some of the items (within a corridor). In particular, it can happen that we moved some items into C that were originally in some other grid cell C'. This reduces the empty space in C for the items in $OPT''_{small}(C)$. Assume that there is a horizontal subcorridor H intersecting C such that some items or parts of items within H were moved into C that were not in C before. Then such items were moved vertically and the corridor containing H must intersect the upper or lower boundary of C. The part of this subcorridor lying within C has a height of at most $\varepsilon_{large} \cdot h(C)$. Thus, the total area of C lost in this way is bounded by $O(\varepsilon_{large}a(C))$ which includes analogous vertical subcorridors.

Like in Lemma 69 we argue that in each horizontal box of size $a \times b$ we remove a horizontal strip of height $3\varepsilon b$ and then the created containers lie in a box of height $(1-3\varepsilon)(1+\varepsilon_{ra})b$. In particular, if the box does not intersect the top or bottom edge of C then within C its containers use only a box of dimension $a' \times (1-3\varepsilon)(1+\varepsilon_{ra})b$ where a' denotes the width of the box within C, i.e., the width of the intersection of the box with C. If the box intersects the top or bottom edge of C then we cannot guarantee that the free space lies within C. However, the total area of such boxes is bounded by $O(\varepsilon_{large}a(C))$. We can apply a symmetric argument to vertical boxes. Then, the total area of C used by containers is at most $(1-3\varepsilon)(1+\varepsilon_{ra})a(C)+O(\varepsilon_{large}a(C)) \leq (1-2\varepsilon)a(C)$. This gives the claim of the lemma. □

Next, we argue that the items in $OPT''_{small}(C)$ have very small total area. Recall that they are the items intersecting C that are not contained in a box. The total number of boxes and boundary curves intersecting C is $O_{\varepsilon,\varepsilon_{large}}(1)$ and in particular, this quantity does not depend on ε_{small}. Hence, by choosing ε_{small} sufficiently small, we can ensure that $a(OPT''_{small}(C)) \leq \varepsilon a(C)$. Then, similarly as in Lemma 18 we can argue that if ε_{small} is small enough then we can pack the items in $OPT''_{small}(C)$ using NFDH into the empty space within C.

B.5 L&C Packings

We iterate the above construction, obtaining pairwise disjoint sets $K(1), K(2), \ldots$ until we find a set $K(k)$ such that $p(K(k)) \leq \varepsilon \cdot OPT$. Since the sets $K(0), K(1), \ldots$

are pairwise disjoint there must be such a value k with $k \leq 1/\varepsilon$. Thus, $|\mathscr{K}(k-1)| \leq O_\varepsilon(1)$. We build the grid given by the x- and y-coordinates of $\mathscr{K}(k-1)$, giving a set of cells $\mathscr{C}_k$. As described above we define the corridor partition, the partition of the corridors into boxes (with the different orders to process the subcorridors as described in Section 4.4.2) and finally into containers. Denote by $OPT_{small}(\mathscr{C}_k)$ the resulting set of small items.

We consider the candidate packings based on the grid $\mathscr{C}_k$. For each of the six candidate packings with a degenerate L we can pack almost all small items of the original packing. We define $\mathscr{R}_{lc}$ and $\mathscr{R}_{sc}$ the sets of items in long and short subcorridors in the initial corridor partition, respectively. Exactly as in the cardinality case, a subcorridor is long if it is longer than $N/2$ and short otherwise. As before we divide the items into fat and thin items and define the sets OPT_{SF}, OPT_{LT}, and OPT_{ST} accordingly. Moreover, we define the set OPT_{LF} to contain all items in $\mathscr{R}_{lc}$ that are fat in all candidate packings *plus* the items in $\mathscr{K}(k-1)$.

Thus, we obtain the respective claims of Lemmas 65, 66, and 67 in the weighted setting. For the following lemma let $OPT_{small} := OPT_{small}(\mathscr{C}_k)$.

Lemma 88. *Let $OPT_{L\&C}$ the most profitable solution that is packed by an L&C packing.*

(a) $p(OPT_{L\&C}) \geq (1-O(\varepsilon))(p(OPT_{LF}) + p(OPT_{SF}) + p(OPT_{small}))$

(b) $p(OPT_{L\&C}) \geq (1-O(\varepsilon))(p(OPT_{LF}) + \frac{p(OPT_{SF})}{2} + \frac{p(OPT_{LT})}{2} + p(OPT_{small}))$

(c) $p(OPT_{L\&C}) \geq (1-O(\varepsilon))(p(OPT_{LF}) + \frac{p(OPT_{SF})}{2} + \frac{p(OPT_{ST})}{2} + p(OPT_{small}))$.

For the candidate packing for the non-degenerate-L case (Lemma 68 in Section 4.4.2) we first add the small items as described above. Then we remove the items in $\mathscr{K}(k-1)$. Then, like above, with a random shift we delete items touching a horizontal and a vertical strip of width $3\varepsilon N$. Like before, each item i is still contained in the resulting solution with probability $1/2 - 15\varepsilon$ (note that we cannot make such a claim for the items in $\mathscr{K}(k-1)$). For each small item we can even argue that it still contained in the resulting solution with probability $1-O(\varepsilon)$ (since it is that small in both dimensions). We proceed with the construction of the boundary L and the assignment of the items into it like in the unweighted case.

Lemma 89. *For the solution $OPT_{L\&C}$ we have that $p(OPT_{L\&C}) \geq (1-O(\varepsilon))(\frac{3}{4}p(OPT_{LT}) + p(OPT_{ST}) + \frac{1-O(\varepsilon)}{2}p(OPT_{SF}) + p(OPT_{small}))$.*

When we combine Lemmas 88 and 89 we conclude that $p(OPT_{L\&C}) \geq (17/9 + O(\varepsilon))p(OPT)$. Similarly as before, the worst case is obtained, up to $1-O(\varepsilon)$ factors, when we have that $p(OPT_{LT}) = p(OPT_{SF}) = p(OPT_{ST})$, $p(OPT_{LF}) = \frac{5}{4}p(OPT_{LT})$, and $p(OPT_{small}) = 0$. This completes the proof of Lemma 59, and consequently the proof of Theorem 40.

Appendix C

Improved Approximation Algorithm for Cardinality 2DK without rotations

In this section, we present a refined approximation algorithm for the cardinality case when rotations are not allowed. More in detail, we prove the following result.

Theorem 90. *There is a polynomial-time* $\frac{558}{325} + \varepsilon < 1.72$ *approximation algorithm for cardinality 2DK.*

Along this section, since the profit of each item is equal to 1, instead of $p(\mathcal{R})$ for a set of items $\mathcal{R}$ we will just write $|\mathcal{R}|$. We will use most of the notation defined in Section 4.4.2. Recall that for two given constants $0 < \varepsilon_{small} < \varepsilon_{large} \leq 1$, we partition the instance into:

- $\mathcal{R}_{small}$, the set of items with $h(R), w(R) \leq \varepsilon_{small}N$, and we denote them as *small* items;
- $\mathcal{R}_{large}$, the set of items with $h(R), w(R) > \varepsilon_{large}N$, and we denote them as *large* items;
- $\mathcal{R}_{hor}$, the set of items with $w(R) > \varepsilon_{large}N$ and $h(R) \leq \varepsilon_{small}N$, and we denote them as *horizontal* items;
- $\mathcal{R}_{ver}$, the set of items with $h(R) > \varepsilon_{large}N$ and $w(R) \leq \varepsilon_{small}N$, and we denote them as *vertical* items;
- $\mathcal{R}_{int}$, the set of remaining items, and we denote them as *intermediate* items.

The corresponding intersection with OPT defines the sets OPT_{small}, OPT_{large}, OPT_{hor}, OPT_{ver} and OPT_{int}, respectively. As discussed in Section 4.4.1, since any feasible solution contains at most $\frac{1}{\varepsilon_{large}^2}$ large items, we can assume in this case that $OPT_{large} = \emptyset$. Furthermore, thanks to Lemma 43, ε_{small} and ε_{large} can be chosen in such a way that $\varepsilon_{small} \leq \varepsilon_{large} \leq \varepsilon$, ε_{small} differs from ε_{large} by a large factor and $|OPT_{int}| \leq \varepsilon|OPT|$. Building upon the corridors decomposition from Adamaszek and Wiese [2013], we will again consider OPT_T (thin items), OPT_F (fat items) and OPT_K (killed items) as defined in Section 4.4.2. Thanks to Lemma 62, $|OPT_K| = O_\varepsilon(1)$ and all the involved parameters can be fixed in such a way that the total height (resp. width) of $OPT_T \cap \mathcal{R}_{hor}$ (resp. $OPT_T \cap \mathcal{R}_{ver}$) is at most εN. Recall that a subcorridor is called *long* if its shortest edge has length at least $\frac{N}{2}$ and short otherwise. In the analysis of the algorithm we will again use sets OPT_{LF}, OPT_{LT}, OPT_{SF} and OPT_{ST} as defined in Section 4.4.2, corresponding to items from OPT_F inside long corridors, items from OPT_T inside long corridors, items from OPT_F inside short corridors and items from OPT_T inside short corridors respectively. For a given $\ell \in (\frac{N}{2}, N]$, we let $\mathcal{R}_{long} \subseteq \mathcal{R}$ be the items whose longest side has length longer than ℓ and $\mathcal{R}_{short} = \mathcal{R} \setminus \mathcal{R}_{long}$. We will assume as in the proof of Lemma 68 that $\ell = \left(\frac{1}{2} + 2\varepsilon_{large}\right)N$. That way we make sure that no long item belongs to a short subcorridor (however it is worth remarking that long corridors may contain short items).

Let us define $OPT_{long} := \mathcal{R}_{long} \cap OPT$ and $OPT_{short} := \mathcal{R}_{short} \cap OPT$. Let us define $\varepsilon_L = \sqrt{\varepsilon}$. Note that $\varepsilon_L \geq \varepsilon \geq \varepsilon_{large} \geq \varepsilon_{small}$. For simplicity and readability of the section, sometimes we will slightly abuse the notation and for any small constant depending on $\varepsilon, \varepsilon_{large}, \varepsilon_{small}$, we will just use $O(\varepsilon_L)$. Now we give a brief informal overview of the refinement and the cases before we go to the details.

C.1 Overview of the Refined Packing

For the refined packing we will consider several L&C packings in order to bound $|OPT_{L\&C}|$, the cardinality of the optimal L&C packing for the instance. Some of the candidate solutions are just extensions of previous constructions (e.g. from Theorem 58 and Lemma 88). Then we consider several new L&C packings where an L-region is packed with items from $\mathcal{R}_{long}$ and the remaining region is used for packing items from $\mathcal{R}_{short}$ using Steinberg theorem (Theorem 5). Note that in the definition of L&C packing in Section 4.4.2, we assumed the height of the horizontal part of the L-region and the width of the vertical part of the L-region to be the same. However, we will consider L-regions where the height of the horizontal part and width of vertical part may differ. Now several cases arise depending on the structure and profit of the L-region. To pack items in OPT_{short} we have three options:

1. We can pack items in $\mathcal{R}_{short}$ using Steinberg theorem into one rectangular region, in which case we need both sides of the region to be greater than $\frac{1}{2} + 2\varepsilon_{large}$.

2. We can pack items in $\mathcal{R}_{short}$ using Steinberg theorem such that vertical and horizontal items are packed separately into different vertical and horizontal rectangular regions inside the knapsack.

3. If $a(OPT_{short})$ is large, we might pack only a small region with items in OPT_{long}, and use the remaining larger space in the knapsack to pack a significant fraction of profit from OPT_{short}.

Now depending on the structure of the L-packing and $a(OPT_{short})$, we arrive at several different cases. If the L-region has very small width and height, we have case (1). Else if the L-region has very large width (or height), we have case (2B), where we pack nearly $\frac{1}{2}|OPT_{long}|$ in the L-region and then pack items from $\mathcal{R}_{short}$ in one large rectangular region. Otherwise, we have case (2A), where either we pack only items from $OPT_{long} \cap OPT_T$ (See Lemma 93, used in case: (2Ai)) or nearly $\frac{3}{4}|OPT_{long}|$ (See Lemma 94, used in cases (2Aii), (2Aiiia)) or in another case, we pack the vertical and horizontal items in OPT_{short} in two different regions through a more complicated packing (See case (2Aiiib)). The details of these cases can be found in the proof of Theorem 90.

C.2 Design of Candidate Solutions

Let us first start with some extensions of previous solutions. Note that by using analogous arguments as in the proof of Theorem 58, we can derive the following inequalities which lead to a $\left(\frac{16}{9} + O(\varepsilon_L)\right)$-approximation algorithm.

$$|OPT_{L\&C}| \geq \frac{3}{4}|OPT_{long}| \tag{C.1}$$

$$|OPT_{L\&C}| \geq \left(\frac{1}{2} - O(\varepsilon_L)\right)|OPT_{long}| + \left(\frac{3}{4} - O(\varepsilon_L)\right)|OPT_{short}| \tag{C.2}$$

Now from Lemma 62, items in $OPT_{short} \cap OPT_T$ can be packed into two containers of size $\ell \times \varepsilon N$ and $\varepsilon N \times \ell$. We can adapt part of the results in Lemma 88 to obtain the following inequalities.

Proposition 91. *The following inequalities hold:*

$$|OPT_{L\&C}| \geq (1-O(\varepsilon_L))(|OPT_{long} \setminus OPT_T| + |OPT_{short} \setminus OPT_T|). \tag{C.3}$$

$$|OPT_{L\&C}| \geq (1-O(\varepsilon_L))(|OPT_{long} \setminus OPT_T| + \frac{1}{2}(|OPT_{short} \setminus OPT_T| + |OPT_{long} \cap OPT_T|)). \tag{C.4}$$

Proof. Inequality (C.3) follows directly from Lemma 88 since $OPT_{LF} \cup OPT_{SF} \cup OPT_{small} = (OPT_{long} \setminus OPT_T) \cup (OPT_{short} \setminus OPT_T)$ and both sets are disjoint. Inequality (C.4) follows from Lemma 66: if we consider the sum of the number of packed items corresponding to the 4 subcases associated with the case "short horizontal/short vertical", then every $i \in OPT_{long} \setminus OPT_T \subseteq OPT_{LF}$ appears four times, every $i \in OPT_{short} \cap OPT_{LF}$ appears four times, every $i \in OPT_{SF}$ appears twice and every $i \in OPT_{long} \cap OPT_T$ appears twice. After including a $(1-O(\varepsilon_L))$ fraction of OPT_{small}, and since $(OPT_{short} \cap OPT_{LF}) \cup OPT_{SF} \cup OPT_{small} = OPT_{short} \setminus OPT_T$, the inequality follows by averaging the cardinality of the four packings. □

We will make use of Steinberg theorem (Theorem 5) to pack items from OPT_{short} in order to obtain better solutions. The following is a simple application of the theorem.

Corollary 92. *Let $\mathcal{R}'$ be a set of items such that $\max_{i \in \mathcal{R}'} h(i) \leq \left(\frac{1}{2} + 2\varepsilon_{large}\right)N$ and $\max_{i \in \mathcal{R}'} w(i) \leq \left(\frac{1}{2} + 2\varepsilon_{large}\right)N$. Then for any $\alpha, \beta \leq \frac{1}{2} - 2\varepsilon_{large}$, all of $\mathcal{R}'$ can be packed into a knapsack of width $(1-\alpha)N$ and height $(1-\beta)N$ if*

$$a(\mathcal{R}') \leq \left(\frac{1}{2} - (\alpha+\beta)\left(\frac{1}{2} + 2\varepsilon_{large}\right) - 8\varepsilon_{large}^2\right)N^2.$$

Now we prove a stronger version of Lemma 68 for the cardinality case.

Lemma 93. *If $a(OPT_{short} \setminus OPT_T) \leq \gamma N^2$ for any $\gamma \leq 1$, then*

$$|OPT_{L\&C}| \geq \frac{3}{4}|OPT_{long} \cap OPT_T| + |OPT_{short} \cap OPT_T| + \min\left\{1, \frac{1-O(\varepsilon_L)}{2\gamma}\right\}|OPT_{short} \setminus OPT_T|.$$

Proof. As in Lemma 68, we can pack $\frac{3}{4}|OPT_{long} \cap OPT_T| + |OPT_{short} \cap OPT_T|$ many items in a boundary L-region plus two boxes on the other two sides of the knapsack and then a free square region with side length $(1-3\varepsilon)N$ can be used to pack items from $OPT_{short} \setminus OPT_T$. From Corollary 92, any subset of items of $OPT_{short} \setminus OPT_T$ with total area at most $(1-O(\varepsilon_L))N^2/2$ can be packed into that square region of length $(1-3\varepsilon)N$. Thus we sort items from $OPT_{short} \setminus OPT_T$ in the order of nondecreasing area and iteratively pick them until their total area reaches $(1-O(\varepsilon_L)-\varepsilon_{small})N^2/2$. Using Steinberg theorem, there exists a packing of the selected items. If $2\gamma \leq 1-O(\varepsilon_L)-\varepsilon_{small}$ then the profit of this packing is $|OPT_{short} \setminus OPT_T|$, and otherwise the total profit is at least $\frac{1-O(\varepsilon_L)}{2\gamma}|OPT_{short} \setminus OPT_T|$. We can finally obtain an L&C packing similarly to Lemma 68. □

Now the following lemma will be useful when $a(OPT_{short})$ is large.

Lemma 94. *If $a(OPT_{short}) > \gamma N^2$ for any $\gamma \geq \frac{3}{4}+\varepsilon+\varepsilon_{large}$, then*

$$|OPT_{L\&C}| \geq \frac{3}{4}|OPT_{long}| + \frac{(3\gamma-1-O(\varepsilon_L))}{4\gamma}|OPT_{short}|.$$

Proof. Similarly to Lemma 56 in Section 4.4.1, we start from the optimal packing and move all items in OPT_{long} to the boundary such that all of them are contained in a boundary ring. Note that unlike the case when we only pack $OPT_{long} \cap OPT_T$ in the boundary region, the boundary ring formed by OPT_{long} may have width or height larger than εN. Let us call the 4 stacks in the ring to be *subrings*. Let us assume that left and right subrings have width $\alpha_{left}N$ and $\alpha_{right}N$ respectively, and bottom and top subrings have height $\beta_{bottom}N$ and $\beta_{top}N$ respectively. By possibly killing one of the long items, subrings can be arranged such that $\alpha_{left}, \alpha_{right}, \beta_{bottom}, \beta_{top} \leq \frac{1}{2}$: If no vertical item intersects the vertical line $x = \frac{N}{2}$ and no horizontal item intersects the horizontal line $y = \frac{N}{2}$ this property holds directly. If one of the previous cases is not satisfied, by deleting such item we can ensure the desired property at a negligible loss of profit, and notice that it is not possible that both cases happen at the same time since items are long.

As $a(OPT_{short}) > \gamma N^2$, then $a(OPT_{long}) < (1-\gamma)N^2$. Let us define $\alpha = \alpha_{left} + \alpha_{right}$ and $\beta = \beta_{bottom} + \beta_{top}$. Then $(\alpha+\beta)N \cdot \frac{N}{2} \leq a(OPT_{long})$, which implies that $\frac{\alpha+\beta}{2} < 1-\gamma$. Hence, we get the following two inequalities:

$$(\alpha+\beta) \leq 2(1-\gamma); \tag{C.5}$$

$$a(OPT_{short}) \leq N^2 - a(OPT_{long}) \leq \left(1-\frac{(\alpha+\beta)}{2}\right)N^2. \tag{C.6}$$

Now consider the case when we remove the top horizontal subring and construct a boundary L-region as in Lemma 56. We will assume that items in the L-region are pushed to the left and bottom as much as possible. Then, the boundary L-region has width $(\alpha_{left}+\alpha_{right})N$ and height $\beta_{bottom}N$. We will use Steinberg theorem to show the existence of a packing of items from OPT_{short} in a subregion of the remaining space with width $N-(\alpha_{left}+\alpha_{right}+\varepsilon)N$ and height $N-(\beta_{bottom}+\varepsilon)N$, and use the rest of the area for resource augmentation to get an L&C packing.

Since $\gamma \geq \frac{3}{4} + \varepsilon + \varepsilon_{large}$, we have from (C.5) that $\alpha + \beta + 2\varepsilon \leq 2(1-\gamma) + 2\varepsilon \leq 1/2 - 2\varepsilon_{large}$. So $\alpha + \varepsilon \leq 1/2 - 2\varepsilon_{large}$ and $\beta + \varepsilon \leq 1/2 - 2\varepsilon_{large}$. Thus from Corollary 92, in the region with width $N - (\alpha_{left} + \alpha_{right} + \varepsilon)N$ and height $N(1 - \beta_{bottom} - \varepsilon)$ we can pack items from OPT_{short} of total area at most $\left(\frac{1}{2} - \frac{(\alpha_{left} + \alpha_{right} + \beta_{bottom})}{2} - O(\varepsilon_L)\right)N^2$. Hence, we can take the items in OPT_{short} in the order of non-decreasing area until their total area reaches $\left(\frac{1}{2} - \frac{(\alpha_{left} + \alpha_{right} + \beta_{bottom})}{2} - O(\varepsilon_L) - \varepsilon_{small}\right)N^2$ and pack at least $|OPT_{short}| \cdot \frac{(\frac{1}{2} - \frac{(\alpha_{left} + \alpha_{right} + \beta_{bottom})}{2} - O(\varepsilon_L) - \varepsilon_{small})}{(1 - \frac{(\alpha+\beta)}{2})}$ many items using Steinberg theorem.

If we now consider all the four different cases corresponding to removal of the four different subrings and take the average of profits obtained in each case, we pack at least

$$\begin{aligned}
& \frac{3}{4}|OPT_{long}| + |OPT_{short}| \cdot \left(\frac{(\frac{1}{2} - \frac{3}{8}(\alpha_{left} + \alpha_{right} + \beta_{bottom} + \beta_{top}) - O(\varepsilon_L)}{(1 - \frac{(\alpha+\beta)}{2})} \right) \\
= \; & \frac{3}{4}|OPT_{long}| + |OPT_{short}| \cdot \left(\frac{(\frac{1}{2} - \frac{3}{8}(\alpha + \beta) - O(\varepsilon_L))}{(1 - \frac{(\alpha+\beta)}{2})} \right) \\
\geq \; & \frac{3}{4}|OPT_{long}| + |OPT_{short}| \cdot \frac{3\gamma - 1 - O(\varepsilon_L)}{4\gamma},
\end{aligned}$$

where the last inequality follows from (C.5) and the fact that the expression is decreasing as a function of $(\alpha + \beta)$. □

Now we can start with the proof of Theorem 90.

Proof of Theorem 90. In the refined analysis, we will consider different solutions and show that the best of them always achieves the claimed approximation guarantee. We will pack some items in a boundary L-region (either a subset of only $OPT_{long} \cap OPT_T$ or a subset of OPT_{long}) using the PTAS for L-packings described in Section 4.3, and in the remaining area of the knapsack (outside of the boundary L-region), we will pack a subset of items from OPT_{short}.

Consider the ring as constructed in the beginning of the proof of Lemma 94. Then we remove the least profitable subring and repack the remaining items from OPT_{long} in a boundary L-region. W.l.o.g. assume that the horizontal top subring was the least profitable subring. The other cases are analogous. We will use the same notation as in Lemma 94, and also define $w_L = (\alpha_{left} + \alpha_{right}), h_L = \beta_{bottom}$. Now let us consider two cases (see Figure C.1 for an overview of the subcases of case 2).

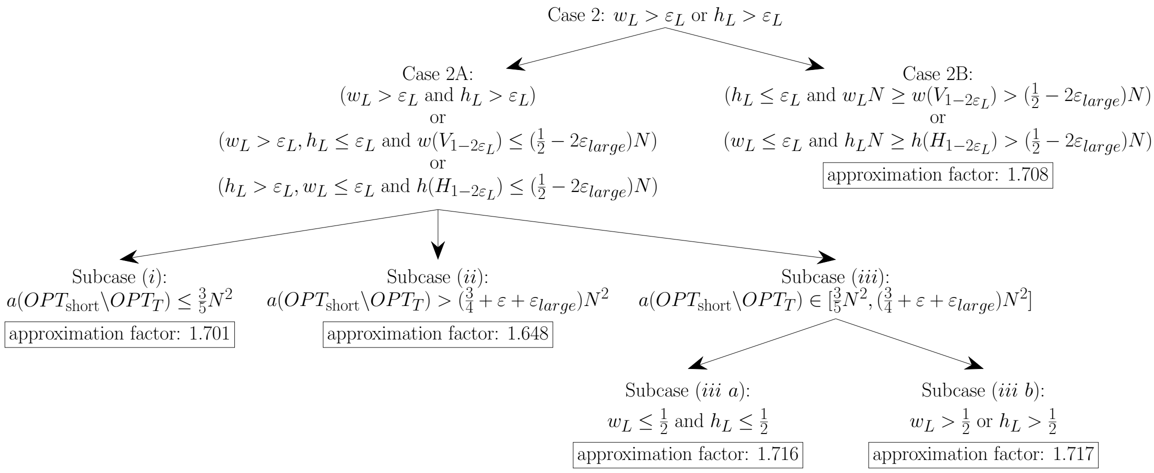
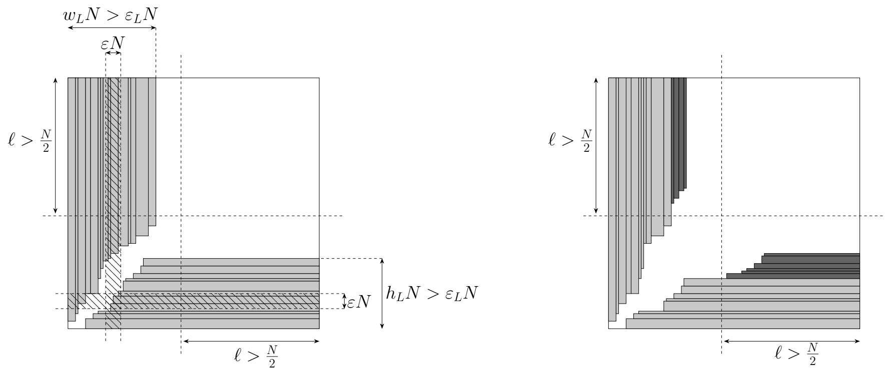


Figure C.1. Summary of the cases.

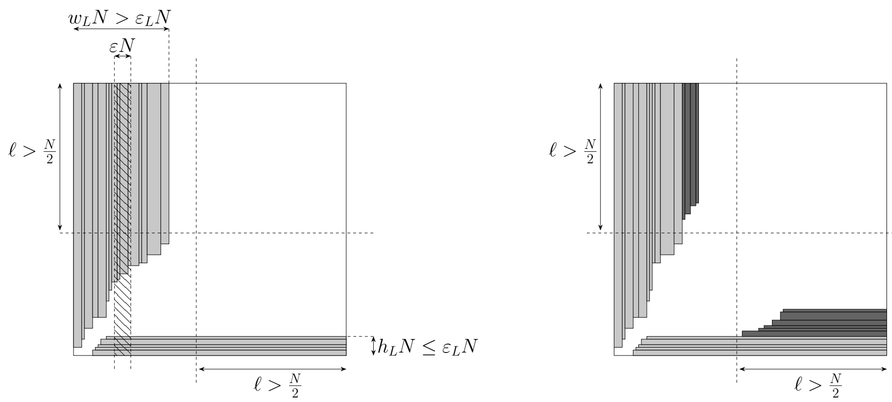
• **Case 1.** $w_L \leq \varepsilon_L, h_L \leq \varepsilon_L$.
In this case, following the proof of Lemma 93 (using $\gamma = 1$), we can pack $\frac{3}{4}|OPT_{long}| + |OPT_{short} \cap OPT_T| + \frac{1-O(\varepsilon_L)}{2}|OPT_{short} \setminus OPT_T|$. This along with inequalities (C.2), (C.3) and (C.4) will give us a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.
• **Case 2.** $w_L > \varepsilon_L$ **or** $h_L > \varepsilon_L$.
Let $V_{1-2\varepsilon_L}$ be the set of vertical items having height strictly larger than $(1-2\varepsilon_L)N$. Similarly, let $H_{1-2\varepsilon_L}$ be the set of horizontal items of width strictly larger than $(1-2\varepsilon_L)N$.
♢ *Case 2A.* $\Big(w_L > \varepsilon_L$ *and* $h_L > \varepsilon_L\Big)$ *or* $\Big(w_L > \varepsilon_L, h_L \leq \varepsilon_L$, *and* $w(V_{1-2\varepsilon_L}) \leq \big(\frac{1}{2} - 2\varepsilon_{large}\big)N\Big)$ *or* $\Big(h_L > \varepsilon_L, w_L \leq \varepsilon_L$, *and* $h(H_{1-2\varepsilon_L}) \leq \big(\frac{1}{2} - 2\varepsilon_{large}\big)N\Big)$.
We will show that if any of the above three conditions is met, then we can pack $\frac{3(1-O(\varepsilon))}{4}|OPT_{long}| + |OPT_{short} \cap OPT_T|$ in a boundary L-region of width close to $w_L N$ and height close to $h_L N$, and then in the remaining area we will pack some items from $OPT_{short} \setminus OPT_T$ using Steinberg theorem and resource augmentation.
Packing of items from $OPT_{long} \cup (OPT_{short} \cap OPT_T)$ **into the** L**-region.**
If $(w_L > \varepsilon_L$ and $h_L > \varepsilon_L)$, we partition the vertical part of the L-region into consecutive strips of width εN. Consider the strip that intersects the least number of vertical items from OPT_{long} among all strips, and we call it to be the *cheapest* εN*-width vertical strip* (See Figure C.2a). Clearly the cheapest εN-width vertical strip intersects at most a $\frac{\varepsilon + 2\varepsilon_{small}}{\varepsilon_L} \leq 3\varepsilon_L$ fraction of the items in the vertical part of the L-region, so we can remove all such vertical items intersected by that strip at a small loss of profit. Similarly, we remove the horizontal items intersected by the cheapest εN-height horizontal strip in the boundary L-region. We now pack



(a) Packing inside the L-region using items from OPT_{long}. Striped strips are cheapest εN-width and cheapest εN-height.

(b) Packing of items in $OPT_{long} \cup (OPT_{short} \cap OPT_T)$. Dark gray items are from $OPT_{short} \cap OPT_T$.

Figure C.2. The case for $w_L > \varepsilon_L$ and $h_L > \varepsilon_L$.



(a) Packing inside the L-region using items from OPT_{long}. The striped strip is the cheapest εN-width strip.

(b) Packing of items in $OPT_{long} \cup (OPT_{short} \cap OPT_T)$. Dark gray items are from $OPT_{short} \cap OPT_T$.

Figure C.3. The case for $w_L > \varepsilon_L$ and $h_L \leq \varepsilon_L$.

the horizontal container for $OPT_{short} \cap OPT_T$ in the free region left by the removed horizontal strip, and the vertical container for $OPT_{short} \cap OPT_T$ in the free region left by the removed vertical strip. Similarly to the proof of Lemma 56 we can sort items in the vertical (resp. horizontal) pile of the L-region according to their height (resp. width), obtaining a feasible $L\&C$ packing (See Figure C.2b).
In the other case ($w_L > \varepsilon_L, h_L \leq \varepsilon_L$ and $w(V_{1-2\varepsilon_L}) \leq \left(\frac{1}{2} - 2\varepsilon_{large}\right)N$), we can again

remove the cheapest εN-width vertical strip in the boundary L-region and pack the vertical container for $OPT_{short} \cap OPT_T$ there (See Figure C.3a). We will now pack items from $OPT_{short} \cap OPT_T$. In the packing of the boundary L-region, we can assume that the vertical items are sorted non-increasingly by height from left to right and pushed upwards until they touch the top boundary. Then, since $w(V_{1-2\varepsilon_L}) \leq \left(\frac{1}{2} - 2\varepsilon_{large}\right)N$ and $(h_L \leq \varepsilon_L)$, the region $\left[\left(\frac{1}{2} - 2\varepsilon_{large}\right)N, N\right] \times [\varepsilon_L N, 2\varepsilon_L N]$ will be completely empty and thus we will have enough space to pack the horizontal container for $OPT_{short} \cap OPT_T$ on top of the horizontal part of the L-region (See Figure C.3b). This leads to a packing in a boundary L-region of width at most $w_L N$ and height at most $(h_L + \varepsilon_L)N$ with total profit at least $\frac{3(1-O(\varepsilon))}{4}|OPT_{long}| + |OPT_{short} \cap OPT_T|$. The last case, when $w_L \leq \varepsilon_L$, is analogous, leading to a packing into a boundary L-region of width at most $(w_L + \varepsilon_L)N$ and height at most $h_L N$ with at least the same profit. Thus,

$$|OPT_{L\&C}| \geq \frac{3(1-O(\varepsilon_L))}{4}|OPT_{long}| + |OPT_{short} \cap OPT_T| \tag{C.7}$$

Packing of items from $OPT_{short} \setminus OPT_T$ into the remaining region.
Note that after packing at least $\frac{3(1-O(\varepsilon))}{4}|OPT_{long}| + |OPT_{short} \cap OPT_T|$ many items in the boundary L-region, the remaining rectangular region of width $(1 - w_L - \varepsilon_L)N$ and height $(1 - h_L - \varepsilon_L)N$ is completely empty. Now we will show the existence of a packing of some items from $OPT_{short} \setminus OPT_T$ in the remaining space of the packing (even using some space from the L-boundary region). Let

$$(OPT_{short} \setminus OPT_T)_{hor} := ((OPT_{short} \setminus OPT_T) \cap \mathscr{R}_{hor}) \cup ((OPT_{short} \setminus OPT_T) \cap \mathscr{R}_{small})$$

and

$$(OPT_{short} \setminus OPT_T)_{ver} := (OPT_{short} \setminus OPT_T) \cap \mathscr{R}_{ver}.$$

Let us assume w.l.o.g. that vertical items are shifted as much as possible to the left and top of the knapsack and horizontal ones are pushed as much as possible to the right and bottom. We divide the analysis in three subcases depending on $a(OPT_{short} \setminus OPT_T)$.
– *Subcase (i).* If $a(OPT_{short} \setminus OPT_T) \leq \frac{3}{5}N^2$, from inequalities (C.2), (C.3), (C.4), (C.7) and Lemma 93, we get a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.
– *Subcase (ii).* If $a(OPT_{short} \setminus OPT_T) > (\frac{3}{4} + \varepsilon + \varepsilon_{large})N^2$, from inequalities (C.2), (C.3), (C.4), (C.7) and Lemma 94, we get a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.
– *Subcase (iii).* Finally, if $\frac{3}{5}N^2 \leq a(OPT_{short} \setminus OPT_T) \leq (\frac{3}{4} + \varepsilon + \varepsilon_{large})N^2$, from inequality (C.5) we get $\alpha + \beta \leq 2(1 - \frac{3}{5}) = \frac{4}{5}$. Now we consider two subcases.

⊙ *Subcase (iii a): $w_L \leq \frac{1}{2}$ and $h_L \leq \frac{1}{2}$.* Note that in this case if $w_L \geq \frac{1}{2} - 2\varepsilon_{large} - 2\varepsilon_L$ (resp., $h_L \geq \frac{1}{2} - 2\varepsilon_{large} - 2\varepsilon_L$), we can remove the cheapest $2(\varepsilon_L + \varepsilon_{large})N$-width vertical (resp., horizontal) strip from the L-region by removing an $O(\varepsilon_L)$ fraction of items in OPT_{long}. Otherwise we have $w_L < \frac{1}{2} - 2\varepsilon_{large} - 2\varepsilon_L$ and $h_L < \frac{1}{2} - 2\varepsilon_{large} - 2\varepsilon_L$. So there is a free rectangular region that has both side lengths at least $N(\frac{1}{2} + 2\varepsilon_{large} + \varepsilon_L)$; we will keep $\varepsilon_L N$ width and $\varepsilon_L N$ height for resource augmentation and use the rest of the rectangular region (with both sides length at least $\left(\frac{1}{2} + 2\varepsilon_{large}\right)N$) to apply Steinberg theorem.

Note that this free rectangular region has area at least $N^2(1 - w_L - 2\varepsilon_L)(1 - h_L - 2\varepsilon_L)$. Now consider items from $(OPT_{short} \setminus OPT_T)_{hor}$ sorted non-decreasingly by area and let us iteratively pick them until their total area becomes at least $\min\{a((OPT_{short} \setminus OPT_T)_{hor}), \frac{N^2(1-w_L-2\varepsilon_L)(1-h_L-2\varepsilon_L)}{2} - \varepsilon_{small}N^2\}$. Thus their total area is at most $\frac{N^2(1-w_L-2\varepsilon_L)(1-h_L-2\varepsilon_L)}{2}$ as the area of any item in $(OPT_{short} \setminus OPT_T)_{hor}$ is at most $\varepsilon_{small}N^2$. Hence, from Steinberg theorem, we can pack these items in the free rectangular region. Similarly, we can pack there items from $(OPT_{short} \setminus OPT_T)_{ver}$ with total area at least

$$\min\{a((OPT_{short} \setminus OPT_T)_{ver}), \frac{N^2(1 - w_L - 2\varepsilon_L)(1 - h_L - 2\varepsilon_L)}{2} - \varepsilon_{small}N^2\}.$$

Since items are sorted non-decreasingly according to their areas, the total profit of the aforementioned packings is bounded below by

$$\min\{1, \left(\frac{(1 - w_L)(1 - h_L)}{2a((OPT_{short} \setminus OPT_T)_{hor})} - O(\varepsilon_L)\right)N^2\}|(OPT_{short} \setminus OPT_T)_{hor}|$$

and

$$\min\{1, \left(\frac{(1 - w_L)(1 - h_L)}{2a((OPT_{short} \setminus OPT_T)_{ver})} - O(\varepsilon_L)\right)N^2\}|(OPT_{short} \setminus OPT_T)_{ver}|$$

respectively. We claim that if we keep the best of the two packings, we can always pack at least $\left(\frac{7}{48} - O(\varepsilon_L)\right)|OPT_{short} \setminus OPT_T|$ many items. To show this we will consider the four possible cases:

- If $\min\{1, \left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short} \setminus OPT_T)_{hor})} - O(\varepsilon_L)\right)N^2\} = \min\{1, \left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short} \setminus OPT_T)_{ver})} - O(\varepsilon_L)\right)N^2\} = 1$, then, by an averaging argument, the best among the two packings has profit at least $\frac{1}{2}(|(OPT_{short} \setminus OPT_T)_{ver}| + |(OPT_{short} \setminus OPT_T)_{hor}|) = \frac{1}{2}|OPT_{short} \setminus OPT_T|$.

- If $\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{hor})}-O(\varepsilon_L)\right)N^2 < 1$ and $\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{ver})}-O(\varepsilon_L)\right)N^2 < 1$, then by an averaging argument we pack at least

$$\begin{aligned}&\frac{N^2}{2}\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{hor})}-O(\varepsilon_L)\right)|(OPT_{short}\setminus OPT_T)_{hor}|\\&+\frac{N^2}{2}\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{ver})}-O(\varepsilon_L)\right)|(OPT_{short}\setminus OPT_T)_{ver}|\\&\geq\frac{N^2}{2}\left(\frac{(1-w_L)(1-h_L)}{2a(OPT_{short}\setminus OPT_T)}-O(\varepsilon_L)\right)|OPT_{short}\setminus OPT_T|\end{aligned}$$

where the inequality follows from the fact that $\frac{a}{b}+\frac{c}{d}\geq\frac{(a+c)}{(b+d)}$ for $a,b,c,d\geq 0$. Since $a(OPT_{short}\setminus OPT_T)\leq(N^2-a(OPT_{long}))\leq(1-\frac{\alpha}{2}-\frac{\beta}{2})N^2\leq(1-\frac{w_L}{2}-\frac{h_L}{2})N^2$ and $w_L+h_L\leq\alpha+\beta\leq\frac{4}{5}$, the amount of items we are packing from $OPT_{short}\setminus OPT_T$ is bounded below by the minimum of

$$f(h_L,w_L)=\left(\frac{(1-w_L)(1-h_L)}{(4-2w_L-2h_L)}-O(\varepsilon_L)\right)N^2|OPT_{short}\setminus OPT_T|$$

over the domain $\{w_L+h_L\leq\frac{4}{5},0\leq w_L\leq\frac{1}{2},0\leq h_L\leq\frac{1}{2}\}$. Since $\frac{\partial f(h_L,w_L)}{\partial h_L}=\frac{-2(1-w_L)^2}{(4-2w_L-2h_L)^2}\leq 0$ and $\frac{\partial f(h_L,w_L)}{\partial w_L}=\frac{-2(1-h_L)^2}{(4-2w_L-2h_L)^2}\leq 0$, the function is decreasing with respect to both its arguments, implying that the minimum value must be attained when $h_L+w_L=\frac{4}{5}$. This in turn implies that the amount of items from $OPT_{short}\setminus OPT_T$ we are packing is bounded below by the minimum of $f(h_L,\frac{4}{5}-h_L)$ over the interval $[\frac{3}{10},\frac{1}{2}]$. Since

$$f(h_L,\frac{4}{5}-h_L)=\left(\frac{5}{12}(1-h_L)(\frac{1}{5}-h_L)-O(\varepsilon_L)\right)N^2|OPT_{short}\setminus OPT_T|$$

describes a parabola centered at $h_L=\frac{2}{5}$, the minimum value on the aforementioned interval is attained at both limits $h_L=\frac{3}{10}$ and $h_L=\frac{1}{2}$ with a value of $\left(\frac{7}{48}-O(\varepsilon_L)\right)|OPT_{short}\setminus OPT_T|$.

- If $\min\{1,\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{hor})}-O(\varepsilon_L)\right)N^2\}=1$ and $\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{ver})}-O(\varepsilon_L)\right)N^2<1$ (the remaining case being analogous), then we are packing at least

$$\begin{aligned}&\frac{1}{2}\left(|(OPT_{short}\setminus OPT_T)_{hor}|+(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{ver})}-O(\varepsilon_L))N^2|(OPT_{short}\setminus OPT_T)_{ver}|\right)\\&\geq\frac{N^2}{2}\left(\frac{(1-w_L)(1-h_L)}{2a((OPT_{short}\setminus OPT_T)_{ver})}-O(\varepsilon_L)\right)(|(OPT_{short}\setminus OPT_T)_{hor}|+|(OPT_{short}\setminus OPT_T)_{ver}|)\\&\geq\frac{N^2}{2}\left(\frac{(1-w_L)(1-h_L)}{2a(OPT_{short}\setminus OPT_T)}-O(\varepsilon_L)\right)|OPT_{short}\setminus OPT_T|\\&\geq\left(\frac{7}{48}-O(\varepsilon_L)\right)|OPT_{short}\setminus OPT_T|,\end{aligned}$$

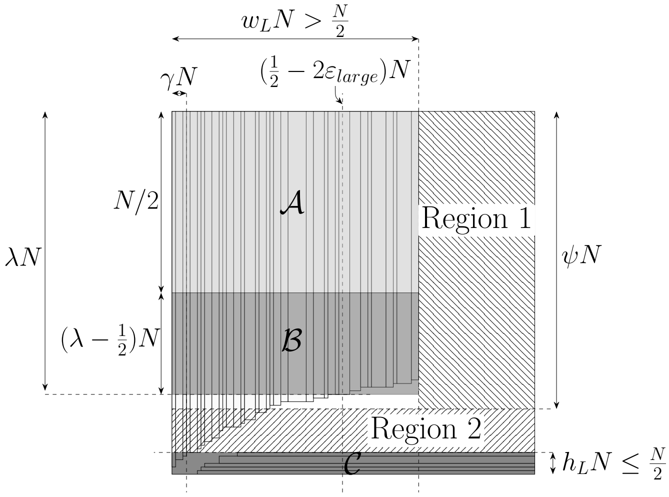


Figure C.4. Case 2A(iii)b in the proof of Theorem 90

where the last inequality comes from the analysis of the previous case.

From this we conclude that

$$|OPT_{L\&C}| \geq \frac{3(1-O(\varepsilon_L))}{4}|OPT_{long}|+|OPT_{short}\cap OPT_T|+\left(\frac{7}{48}-O(\varepsilon_L)\right)|OPT_{short}\setminus OPT_T|.$$

This together with inequalities (C.2), (C.3), (C.4) and Lemma 93 gives us a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.

⊙ *Subcase (iii b):* $w_L > \frac{1}{2}$ (then from inequality (C.5), $h_L \leq \frac{3}{10}$). Note that $a(OPT_{long}) \leq (1-\frac{3}{5})N^2 = \frac{2}{5}N^2$.

Let us define some parameters from the current packing to simplify the calculations. Let λN be the height of the item in the packing that intersects or touches the vertical line $x = \left(\frac{1}{2} - 2\varepsilon_{large}\right)N$ (if two items touch such line, we choose that tallest one) and γN be the total width of vertical items having height greater than $(1-h_L)N$. We define also the following three regions in the knapsack: $\mathscr{A}$, the rectangular region of width $w_L N$ and height $\frac{1}{2}N$ in the top left corner of the knapsack; $\mathscr{B}$, the rectangular region of width $w_L N$ and height $(\lambda - \frac{1}{2})N$ below $\mathscr{A}$ and left-aligned with the knapsack; and $\mathscr{C}$, the rectangular region of width N and height $h_L N$ touching the bottom boundary of the knapsack. Notice that $\mathscr{A}$ is fully occupied by vertical items, $\mathscr{B}$ is almost fully occupied by vertical items except for the right region of width $w_L N - \left(\frac{1}{2} - 2\varepsilon_{large}\right)N$, and at least half of $\mathscr{C}$ is occupied by horizontal items (some vertical items may overlap with this region). Our goal is to pack some items from $OPT_{short} \setminus OPT_T$ in the ⅃-shaped region outside $\mathscr{A} \cup \mathscr{B} \cup \mathscr{C}$. Let $\psi \in [\lambda, 1-h_L]$ be a parameter to be fixed later. We will use, when possible, the following regions for packing items from $OPT_{short} \setminus OPT_T$: Region 1 on the top right corner of the knapsack with width $N(1-w_L)$ and height ψN and Region 2 which is the rectangular region $[0,N] \times [h_L N, (1-\psi)\cdot N]$ (see Figure C.4). Region 1 is completely free but Region 2 may overlap with vertical items.

We will now divide Region 2 into a constant number of boxes such that: they do not overlap with vertical items, the total area inside Region 2 which is neither overlapping with vertical items nor covered by boxes is at most $O(\varepsilon_L)N^2$ and each box has width at least $\left(\frac{1}{2}+2\varepsilon_{large}\right)N$ and height at least εN. That way we will be able to pack items from $(OPT_{short} \setminus OPT_T)_{ver}$ into the box defined by Region 1 and items from $(OPT_{short} \setminus OPT_T)_{hor}$ into the boxes defined inside Region 2 using almost completely its free space. In order to create the boxes inside Region 2 we first create a monotone chain by doing the following: Let $(x_1, y_1) = (\gamma N, h_L)$. Starting from position (x_1, y_1), we draw an horizontal line of length $\varepsilon_L N$ and then a vertical line from bottom to top until it touches a vertical item, reaching position (x_2, y_2). From (x_2, y_2) we start again the same procedure and iterate until we reach the vertical line $x = \left(\frac{1}{2}-2\varepsilon_{large}\right)N$ or the horizontal line $y = (1-\psi)N$.

Notice that the area above the monotone chain and below $y = (1-\psi)N$ that is not occupied by vertical items, is at most $\sum_i \varepsilon_L N(y_{i+1}-y_i) \le \varepsilon_L N^2$. The number of points (x_i, y_i) defined in the previous procedure is at most $1/\varepsilon_L$. By drawing an horizontal line starting from each (x_i, y_i) up to (N, y_i), together with the drawn lines from the monotone chain and the right limit of the knapsack, we define $k \le 1/\varepsilon_L$ boxes. We discard the boxes having height less than εN, whose total area is at most $\frac{\varepsilon}{\varepsilon_L}N^2 = \varepsilon_L N^2$, and have all the desired properties for the boxes.

Note that the area that is occupied by items from OPT_{long} in regions $\mathscr{A}, \mathscr{B}$ and $\mathscr{C}$ is at least $(\frac{1}{2}w_L + (\lambda - \frac{1}{2})(\frac{1}{2}-2\varepsilon_{large}) + \frac{1}{2}h_L)N^2$. Since the total area of items from OPT_{long} is at most $\frac{2}{5}N^2$, the total area occupied by items in OPT_{long} in Region 2 is at most

$$N^2(\frac{2}{5} - \frac{1}{2}w_L - (\lambda - \frac{1}{2})(\frac{1}{2} - 2\varepsilon_{large}) - \frac{1}{2}h_L) \le N^2(\frac{13}{20} - \frac{w_L}{2} - \frac{\lambda}{2} - \frac{h_L}{2} + \varepsilon_{large}).$$

This implies that the total area of the horizontal boxes is at least $N^2(1-\psi-h_L) - N^2(\frac{13}{20} - \frac{w_L}{2} - \frac{\lambda}{2} - \frac{h_L}{2}) - O(\varepsilon_L)N^2$ and the area of the vertical box is $(1-w_L)\psi N^2$. Ignoring the $O(\varepsilon_{large})$-term, these two areas become equal if we set $\psi = \frac{7+10(w_L+\lambda-h_L)}{40-20w_L}$. It is not difficult to verify that in this case $\psi \le 1-h_L$. If $\frac{7+10(w_L+\lambda-h_L)}{40-20w_L} \ge \lambda$, then we set $\psi = \frac{7+10(w_L+\lambda-h_L)}{40-20w_L}$. Otherwise we set $\psi = \lambda$.

First, consider the case $\psi = \frac{7+10(w_L+\lambda-h_L)}{40-20w_L}$. Since $\psi \ge \lambda$, boxes inside Region 2 have width at least $\left(\frac{1}{2}+2\varepsilon_{large}\right)N$ and height at least $\varepsilon N \gg \varepsilon_{small}N$ (recall that

ε_{small} differs by a large factor from $\varepsilon_{large} \leq \varepsilon$), and the box in Region 1 has height at least $\left(\frac{1}{2} + 2\varepsilon_{large}\right)N$ and width at least $\frac{1}{5}N \gg \varepsilon_{small}N$. By using Steinberg theorem, we can always pack in these boxes at least

$$\left(\min\left\{1, \frac{\frac{1}{2}(N - w_L N)\psi N}{a((OPT_{short} \setminus OPT_T)_{hor})} - \varepsilon_{small}N^2\right\}\right)|(OPT_{short} \setminus OPT_T)_{hor}|$$
$$+\left(\min\left\{1, \frac{\frac{1}{2}(N - w_L N)\psi N}{a((OPT_{short} \setminus OPT_T)_{ver})} - \varepsilon_{small}N^2\right\}\right)|(OPT_{short} \setminus OPT_T)_{ver}|.$$

Note that from each box B' of height $h \geq \varepsilon N$ we can remove the cheapest εh-horizontal strip and use resource augmentation to get a container based packing with nearly the same profit as B'. Thus by performing a similar analysis to the one done in Subcase (iii a), and using the fact that $a(OPT_{short} \setminus OPT_T) \leq N^2 - (\frac{\alpha}{2} + (\lambda - \frac{1}{2})\frac{1}{2} + \frac{\beta}{2})N^2 \leq N^2 - N^2(\frac{w_L}{2} + (\lambda - \frac{1}{2})\frac{1}{2} + \frac{h_L}{2})$, we can minimize the whole expression over the domain $\{\frac{w_L}{2} + (\lambda - \frac{1}{2})\frac{1}{2} + \frac{h_L}{2} \leq \frac{2}{5}, \lambda \leq \psi, \frac{1}{2} \leq w_L \leq \frac{4}{5}, \frac{1}{2} \leq \lambda \leq 1, 0 \leq h_L \leq \frac{3}{10}\}$ and prove that this solution has cardinality at least

$$\left(\frac{3 - O(\varepsilon_L)}{4}\right)|OPT_{long}| + |OPT_{short} \cap OPT_T| + \left(\frac{5}{36} - O(\varepsilon_L)\right)|OPT_{short} \setminus OPT_T|. \quad \text{(C.8)}$$

Thus, using the above inequality along with (C.2), (C.3), (C.4) and Lemma 93, we get a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.

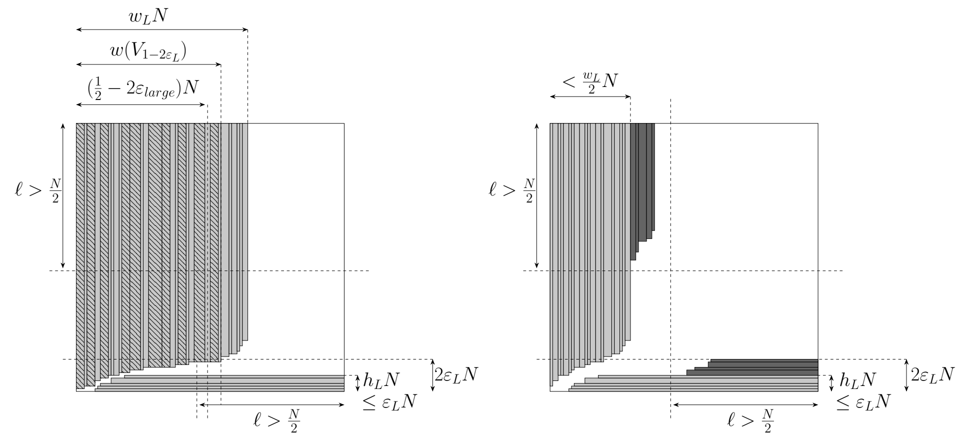
Finally, if $\psi = \lambda > \frac{7+10(w_L+\lambda-h_L)}{40-20w_L}$, the bound for the area of horizontal boxes will not be equal to the area of the vertical box constructed to pack items from $OPT_{short} \setminus OPT_T$. In this case we change the width of the box inside Region 1 to be $w'_L < N(1 - w_L)$ fixed in such a way that the area of this box is equal to the bound we have for the area of the boxes in Region 2, i.e.,

$$N^2(1 - \lambda - h_L) - (\frac{13}{20} - \frac{w_L}{2} - \frac{h_L}{2} - \frac{\lambda}{2} + O(\varepsilon_L))N^2.$$

Performing the same analysis as before, it can be shown that in this case the solution has cardinality at least

$$\left(\frac{(1 - \lambda - h_L)N^2 - (\frac{13}{20} - \frac{w_L}{2} - \frac{h_L}{2} - \frac{\lambda}{2})N^2}{2a(OPT_{short} \setminus OPT_T)} - O(\varepsilon_L)N^2\right)|OPT_{short} \setminus OPT_T|,$$

which is at least $(\frac{1}{6} - O(\varepsilon_L))|OPT_{short} \setminus OPT_T|$ over the domain $\{\frac{w_L}{2} + (\lambda - \frac{1}{2})\frac{1}{2} + \frac{h_L}{2} \leq \frac{2}{5}, \frac{1}{2} \leq w_L \leq \frac{4}{5}, \frac{7+10(w_L+\lambda-h_L)}{40-20w_L} < \lambda \leq 1, 0 \leq h_L \leq \frac{3}{10}\}$ (and this solution leads to a better bound than (C.8)).



(a) Packing inside the L-region using items from OPT_{long}. Striped items are removed.

(b) Packing of items in $OPT_{long} \cup (OPT_{short} \cap OPT_T)$. Dark gray rectangles correspond to $OPT_{short} \cap OPT_T$.

Figure C.5. The case 2B.

◊ *Case 2B.* $\left(h_L \leq \varepsilon_L \text{ and } w_L N \geq w(V_{1-2\varepsilon_L}) > \left(\frac{1}{2} - 2\varepsilon_{large}\right)N\right)$ *or* $\left(w_L \leq \varepsilon_L \text{ and } h_L N \geq h(H_{1-2\varepsilon_L}) > \left(\frac{1}{2} - 2\varepsilon_{large}\right)N\right)$

In the first case, the area of the items in $V_{1-2\varepsilon_L}$ is larger than $\left(\frac{1}{2} - 2\varepsilon_{large}\right)(1 - 2\varepsilon_L)N^2$, and the remaining items in OPT_{long} have area at least $(w_L - \frac{1}{2} + 2\varepsilon_{large})N \cdot \frac{N}{2}$. So, $a(OPT_{long}) > \left(\frac{1}{2} - 2\varepsilon_{large}\right)(1 - 2\varepsilon_L)N^2 + (w_L - \frac{1}{2})N \cdot \frac{N}{2} \geq (\frac{1}{4} + \frac{w_L}{2} - \varepsilon_L - 2\varepsilon_{large})N^2$. Thus $a(OPT_{short} \setminus OPT_T) \leq a(OPT_{short}) < (\frac{3}{4} - \frac{w_L}{2} + \varepsilon_L + 2\varepsilon_{large})N^2$.
Now consider the vertical items in the boundary L-region sorted non-increasingly by width and pick them iteratively until their total width crosses $(\frac{w_L}{2} + 3\varepsilon_L + 2\varepsilon_{large})N$. Remove these items and push the remaining vertical items in the L-region to the left as much as possible. This modified L-region will have profit at least $(\frac{1}{2} - O(\varepsilon_L))|OPT_{long}|$. Now we can put an εN-strip for the vertical items from $OPT_{short} \cap OPT_T$ next to the vertical part of L-region. On the other hand, the horizontal items of $OPT_{short} \cap OPT_T$ can be placed on top of the horizontal part of the L-region. The remaining space will be a free rectangular region of height at least $(1 - 2\varepsilon_L)N$ and width $(1 - \frac{w_L}{2} + 2\varepsilon_L + 2\varepsilon_{large})N$. We will use a part of this rectangular region of height $(1 - 3\varepsilon_L)N$ and width $(1 - \frac{w_L}{2} + \varepsilon_L)N$ to pack items from $OPT_{short} \setminus OPT_T$ and the rest of the region for resource augmentation. Since $\frac{w_L}{2} - \varepsilon_L \leq \frac{1}{2} - \varepsilon_{large}$, we can use Corollary 92 to pack at least $\left(\frac{(1 - \frac{w_L}{2})/2}{\frac{3}{4} - \frac{w_L}{2}} - O(\varepsilon_L)\right)|OPT_{short} \setminus OPT_T| \geq (\frac{3}{4} - O(\varepsilon_L))|OPT_{short} \setminus OPT_T|$ short items in this region

as the expression is increasing with respect to w_L and $w_L > \frac{1}{2} - 2\varepsilon_{large}$. Thus, we get,

$$|OPT_{L\&C}| \geq \left(\frac{1}{2} - O(\varepsilon_L)\right)|OPT_{long}| + |OPT_{short} \cap OPT_T| + \left(\frac{3}{4} - O(\varepsilon_L)\right)|OPT_{short} \setminus OPT_T|. \quad \text{(C.9)}$$

On the other hand, as $a(OPT_{short} \setminus OPT_T) \leq (\frac{3}{4} - \frac{w_L}{2} + \varepsilon_L + 2\varepsilon_{large})N^2$ and $w_L > \frac{1}{2} - 2\varepsilon_{large}$, we get $a(OPT_{short} \setminus OPT_T) \leq (\frac{1}{2} + 3\varepsilon_{large} + \varepsilon_L)N^2$ and thus from Lemma 93 we get

$$\begin{aligned} |OPT_{L\&C}| &\geq \frac{3}{4}|OPT_{long} \cap OPT_T| + |OPT_{short} \cap OPT_T| + (1 - O(\varepsilon_L))|OPT_{short} \setminus OPT_T| \\ &\geq \frac{3}{4}|OPT_{long} \cap OPT_T| + (1 - O(\varepsilon_L))|OPT_{short}|. \quad \text{(C.10)} \end{aligned}$$

From inequalities (C.1), (C.3), (C.4), (C.9) and (C.10) we get a solution with good enough approximation factor. Check Section C.3 and Table C.1 for details.

Now we consider the last case when $w_L \leq \varepsilon_L$ and $h_L N \geq h(H_{1-2\varepsilon_L}) > \left(\frac{1}{2} - 2\varepsilon_{large}\right)N$. Note that since we assumed the cheapest subring was the top subring, after removing it we might be left with only $|OPT_{long} \cap \mathcal{R}_{hor}|/2$ profit in the horizontal part of L-region. So, further removal of items from the horizontal part might not give us a good solution. Thus we show an alternate good packing. We restart with the ring packing and delete the cheapest vertical subring instead of the cheapest subring (i.e., the top subring) and create a new boundary L-region. Here, consider the horizontal items in the boundary L-region in non-increasing order of height and take them until their total height becomes at least $(\frac{\beta_{bottom}+\beta_{top}}{2} + \varepsilon_{small} + \varepsilon)N$. Remove these items and push the remaining horizontal items to the bottom as much as possible. Then, following similar arguments as before, we will obtain the same bounds for the constructed solution.

C.3 Bounding the Approximation Factor

In each one of the cases listed before we are developing a set of different solutions in order to achieve a good approximation factor. Let $z = |OPT_{L\&C}|/|OPT|$, $x_1 = |OPT_{long} \cap OPT_T|/|OPT|$, $x_2 = |OPT_{long} \setminus OPT_T|/|OPT|$, $x_3 = |OPT_{short} \cap OPT_T|/|OPT|$ and $x_4 = |OPT_{short} \setminus OPT_T|/|OPT|$. The following list enumerates all the obtained inequalities in this section, and it is worth remarking that not all of them hold simultaneously.

1. $z \geq \frac{3}{4}x_1 + \frac{3}{4}x_2 + x_3 + \left(\frac{1}{2} - O(\varepsilon_L)\right)x_4$;

2. $z \geq \frac{3}{4}x_1 + \frac{3}{4}x_2$;
3. $z \geq \left(\frac{1}{2} - O(\varepsilon_L)\right)(x_1 + x_2) + \left(\frac{3}{4} - O(\varepsilon_L)\right)(x_3 + x_4)$;
4. $z \geq (1 - O(\varepsilon_L))(x_2 + x_4)$;
5. $z \geq (1 - O(\varepsilon_L))\left(\frac{1}{2}x_1 + x_2 + \frac{1}{2}x_4\right)$;
6. $z \geq \left(\frac{3}{4} - O(\varepsilon_L)\right)(x_1 + x_2) + x_3$;
7. $z \geq \frac{3}{4}x_1 + x_3 + \left(\frac{5}{6} - O(\varepsilon_L)\right)x_4$;
8. $z \geq \frac{3}{4}(x_1 + x_2) + \left(\frac{5}{12} - O(\varepsilon_L)\right)(x_3 + x_4)$;
9. $z \geq \frac{3}{4}x_1 + x_3 + \left(\frac{2}{3} - O(\varepsilon_L)\right)x_4$;
10. $z \geq \frac{3}{4}(x_1 + x_2) + x_3 + \left(\frac{7}{48} - O(\varepsilon_L)\right)x_4$;
11. $z \geq \left(\frac{3}{4} - O(\varepsilon_L)\right)(x_1 + x_2) + x_3 + \left(\frac{5}{36} - O(\varepsilon_L)\right)x_4$;
12. $z \geq \left(\frac{1}{2} - O(\varepsilon_L)\right)(x_1 + x_2) + x_3 + \left(\frac{3}{4} - O(\varepsilon_L)\right)x_4$;
13. $z \geq \frac{3}{4}x_1 + (1 - O(\varepsilon_L))(x_3 + x_4)$.

For each case i, let $\mathscr{A}_i$ be the set of indexes of valid inequalities for case i. Then we can write the following linear program to compute the obtained approximation factor in that case:

$$\begin{array}{ll} \min & z \\ s.t. & \text{Inequalities indexed by } \mathscr{A}_i \\ & \sum_{i=1}^{4} x_i = 1 \\ & z, x_i \geq 0 \qquad \text{for } i = 1,2,3,4. \end{array}$$

Let $c_{j,k}$ be the coefficient accompanying x_k in the constraint $j \in \mathscr{A}_i$, $k = 1,2,3,4$. The dual of the program for case i has the form

$$\begin{array}{ll} \max & -w \\ s.t. & \sum_{j \in \mathscr{A}_i} y_j \leq 1 \\ & \sum_{j \in \mathscr{A}_i} c_{j,k} y_j + w \geq 0 \qquad \text{for } k = 1,2,3,4 \\ & y_j \geq 0 \qquad \text{for } j \in \mathscr{A}_i \\ & w \in \mathbb{R} \end{array}$$

Any feasible solution for the dual program of case i is a lower bound on the fraction of OPT packed in that case. Table C.1 summarizes the analysis described along this section for all the cases, stating the valid inequalities and the approximation factor obtained in each one of them, together with a dual feasible solution. It is not difficult to see that the worst case is 2A(iii)b, implying that $|OPT_{L\&C}| \geq (\frac{325}{558} - O(\varepsilon_L))|OPT|$. Applying the algorithm described in Section 4.2.1 concludes the proof of Theorem 90. □

Case	Valid inequalities	Dual feasible solution	Fraction of OPT packed (w)
1	$1,3,4,5$	$y_1 = \frac{1}{2}, y_3 = \frac{1}{2}, y_4 = 0, y_5 = 0$	$\frac{5}{8} - O(\varepsilon_L)$
$2A(i)$	$3,4,5,6,7$	$y_3 = \frac{17}{54}, y_4 = 0, y_5 = \frac{1}{3}, y_6 = \frac{7}{54}, y_7 = \frac{2}{9}$	$\frac{127}{216} - O(\varepsilon_L)$
$2A(ii)$	$3,4,5,6,8$	$y_3 = \frac{4}{7}, y_4 = 0, y_5 = 0, y_6 = 0, y_8 = \frac{3}{7}$	$\frac{17}{28} - O(\varepsilon_L)$
$2A(iii)a$	$3,4,5,9,10$	$y_3 = \frac{124}{369}, y_4 = 0, y_5 = \frac{1}{3}, y_9 = \frac{2}{9}, y_{10} = \frac{40}{369}$	$\frac{215}{369} - O(\varepsilon_L)$
$2A(iii)b$	$3,4,5,9,11$	$y_3 = \frac{94}{279}, y_4 = 0, y_5 = \frac{1}{3}, y_9 = \frac{2}{9}, y_{11} = \frac{10}{93}$	$\frac{325}{558} - O(\varepsilon_L)$
$2B$	$2,4,5,12,13$	$y_2 = \frac{8}{41}, y_4 = 0, y_5 = \frac{9}{41}, y_{12} = \frac{18}{41}, y_{13} = \frac{6}{41}$	$\frac{24}{41} - O(\varepsilon_L)$

Table C.1. Summary of the case analysis in Theorem 90.

Bibliography

Adamaszek, A., Kociumaka, T., Pilipczuk, M. and Pilipczuk, M. [2017]. Hardness of approximation for strip packing, *ACM Trans. Comput. Theory* **9**(3): 14:1–14:7.

Adamaszek, A. and Wiese, A. [2013]. Approximation schemes for maximum weight independent set of rectangles, *FOCS*, pp. 400–409.

Adamaszek, A. and Wiese, A. [2014]. A QPTAS for maximum weight independent set of polygons with polylogarithmically many vertices., *SODA*, pp. 645–656.

Adamaszek, A. and Wiese, A. [2015]. A quasi-PTAS for the two-dimensional geometric knapsack problem, *SODA*, pp. 1491–1505.

Antoniadis, A., Hoeksma, R., Meißner, J., Verschae, J. and Wiese, A. [2017]. A QPTAS for the general scheduling problem with identical release dates, *ICALP*, Vol. 80, pp. 31:1–31:14.

Baker, B. S., Brown, D. J. and Katseff, H. P. [1981]. A 5/4 algorithm for two-dimensional packing, *J. Algorithm* **2**(4): 348–368.

Baker, B. S., Jr., E. G. C. and Rivest, R. L. [1980]. Orthogonal packings in two dimensions, *SIAM J. Comput.* **9**(4): 846–855.

Bansal, N., Caprara, A., Jansen, K., Prädel, L. and Sviridenko, M. [2009]. A structural lemma in 2-dimensional packing, and its implications on approximability, *ISAAC*, pp. 77–86.

Buchsbaum, A. L., Karloff, H., Kenyon, C., Reingold, N. and Thorup, M. [2004]. OPT versus LOAD in dynamic storage allocation, *SIAM J. Comput.* **33**: 632–646.

Chekuri, C. and Khanna, S. [2005]. A polynomial time approximation scheme for the multiple knapsack problem, *SIAM J. Comput.* **35**(3): 713–728.

Coffman Jr., E. G. and Bruno, J. L. [1976]. *Computer and job-shop scheduling theory*, John Wiley & Sons.

Coffman Jr., E. G., Csirik, J., Galambos, G., Martello, S. and Vigo, D. [2013]. Bin packing approximation algorithms: survey and classification, *Handbook of Combinatorial Optimization*, Springer, pp. 455–531.

Coffman Jr., E. G., Garey, M. R., Johnson, D. S. and Tarjan, R. E. [1980]. Performance bounds for level-oriented two-dimensional packing algorithms, *SIAM J. Comput.* **9**(4): 808–826.

Correa, J. R. and Kenyon, C. [2004]. Approximation schemes for multidimensional packing, *SODA*, pp. 186–195.

Cygan, M., Fomin, F. V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M. and Saurabh, S. [2015]. *Parameterized Algorithms*, 1st edn, Springer Publishing Company, Incorporated.

Epstein, L. and van Stee, R. [2006]. This side up!, *ACM Transactions on Algorithms (TALG)* **2**(2): 228–243.

Feige, U. and Vondrak, J. [2006]. Approximation algorithms for allocation problems: Improving the factor of 1-1/e, *FOCS*, pp. 667–676.

Fishkin, A. V., Gerber, O. and Jansen, K. [2005]. On efficient weighted rectangle packing with large resources, *ISAAC*, pp. 1039–1050.

Fishkin, A. V., Gerber, O., Jansen, K. and Solis-Oba, R. [2005]. Packing weighted rectangles into a square, *MFCS*, pp. 352–363.

Fleischer, L., Goemans, M. X., Mirrokni, V. S. and Sviridenko, M. [2011]. Tight approximation algorithms for maximum separable assignment problems, *Math. Oper. Res.* **36**(3): 416–431.

Gálvez, W., Grandoni, F., Amelli, A. J., Jansen, K., Khan, A. and Rau, M. [2019]. Improved approximation algorithms for restricted instances of strip packing. Unpublished manuscript.

Gálvez, W., Grandoni, F., Heydrich, S., Ingala, S., Khan, A. and Wiese, A. [2017]. Approximating geometric knapsack via L-packings, *FOCS*, pp. 260–271.

Gálvez, W., Grandoni, F., Ingala, S. and Khan, A. [2016]. Improved pseudo-polynomial-time approximation for strip packing, *FSTTCS*, pp. 9:1–9:14.

Garey, M. and Johnson, D. [1979]. *Computers and Intractability: A Guide to the theory of NP-completeness*, Freeman NY.

Garey, M. R. and Johnson, D. S. [1978]. "Strong" NP-completeness results: Motivation, examples, and implications, *JACM* **25**(3): 499–508.

Golan, I. [1981]. Performance bounds for orthogonal oriented two-dimensional packing algorithms, *SIAM J. Comput.* **10**(3): 571–582.

Grandoni, F., Ingala, S. and Uniyal, S. [2015]. Improved approximation algorithms for unsplittable flow on a path with time windows, *WAOA*, pp. 13–24.

Grandoni, F., Kratsch, S. and Wiese, A. [2019]. Parameterized approximation schemes for independent set of rectangles and geometric knapsack, *ESA*. To appear.

Grandoni, F., Mömke, T., Wiese, A. and Zhou, H. [2017]. To augment or not to augment: Solving unsplittable flow on a path by creating slack, *SODA*, SIAM, pp. 2411–2422.

Grötschel, M., Lovasz, L. and Schrijver, A. [2012]. *Geometric Algorithms and Combinatorial Optimization*, Algorithms and Combinatorics, Springer Berlin Heidelberg.

Hadlock, F. [1975]. Finding a maximum cut of a planar graph in polynomial time., *SIAM J. Comput.* **4**(3): 221–225.

Harren, R. [2006]. Approximating the orthogonal knapsack problem for hypercubes, *ICALP*, pp. 238–249.

Harren, R., Jansen, K., Prädel, L. and van Stee, R. [2014]. A $(5/3+\varepsilon)$-approximation for strip packing, *Computational Geometry* **47**(2): 248–267.

Harren, R. and van Stee, R. [2009]. Improved absolute approximation ratios for two-dimensional packing problems, *APPROX-RANDOM*, pp. 177–189.

Henning, S., Jansen, K., Rau, M. and Schmarje, L. [2018]. Complexity and inapproximability results for parallel task scheduling and strip packing, *CSR*, pp. 169–180.

Heydrich, S. and Wiese, A. [2017]. Faster approximation schemes for the two-dimensional knapsack problem, *SODA*, pp. 79–98.

Hochbaum, D. (ed.) [1997]. *Approximation Algorithms for NP-hard Problems*, PWS Publishing Company.

Jansen, K. and Prädel, L. [2014]. A new asymptotic approximation algorithm for 3-dimensional strip packing, *SOFSEM*, pp. 327–338.

Jansen, K. and Rau, M. [2017]. Improved approximation for two dimensional strip packing with polynomial bounded width, *WALCOM*, pp. 409–420.

Jansen, K. and Rau, M. [2019]. Closing the gap for pseudo-polynomial strip packing, *ESA*. To appear.

Jansen, K. and Solis-Oba, R. [2008]. A polynomial time approximation scheme for the square packing problem, *IPCO*, pp. 184–198.

Jansen, K. and Solis-Oba, R. [2009]. Rectangle packing with one-dimensional resource augmentation, *Discrete Optimization* **6**(3): 310–323.

Jansen, K. and Thöle, R. [2010]. Approximation algorithms for scheduling parallel jobs, *SIAM J. Comput.* **39**(8): 3571–3615.

Jansen, K. and van Stee, R. [2005]. On strip packing with rotations, *STOC*, pp. 755–761.

Jansen, K. and Zhang, G. [2004a]. Maximizing the number of packed rectangles, *SWAT*, pp. 362–371.

Jansen, K. and Zhang, G. [2004b]. On rectangle packing: maximizing benefits, *SODA*, pp. 204–213.

Karbasioun, M. M., Shaikhet, G., Kranakis, E. and Lambadaris, I. [2013]. Power strip packing of malleable demands in smart grid, *IEEE ICC*, pp. 4261–4265.

Kenyon, C. and Rémila, E. [2000]. A near-optimal solution to a two-dimensional cutting stock problem, *Math. Oper. Res.* **25**(4): 645–656.

Lau, L.-C., Ravi, R. and Singh, M. [2011]. *Iterative Methods in Combinatorial Optimization*, 1st edn, Cambridge University Press.

Leung, J. Y., Tam, T. W., Wong, C., Young, G. H. and Chin, F. Y. [1990]. Packing squares into a square, *J. Parallel Distr. Comp.* **10**(3): 271–275.

Miyazawa, F. K. and Wakabayashi, Y. [2004]. Packing problems with orthogonal rotations, *LATIN*, pp. 359–368.

Mömke, T. and Wiese, A. [2015]. A $(2+\varepsilon)$-approximation algorithm for the storage allocation problem, *ICALP*.

Nadiradze, G. and Wiese, A. [2016]. On approximating strip packing with a better ratio than 3/2, *SODA*, pp. 1491–1510.

Pilipczuk, M., van Leeuwen, E. J. and Wiese, A. [2017]. Approximation and parameterized algorithms for geometric independent set with shrinking, *MFCS*, Vol. 83, pp. 42:1–42:13.

Ranjan, A., Khargonekar, P. and Sahni, S. [2015]. Offline first fit scheduling in smart grids, *IEEE SCC*, pp. 758–763.

Schiermeyer, I. [1994]. Reverse-fit: A 2-optimal algorithm for packing rectangles, *ESA*, pp. 290–299.

Shmoys, D. B. and Tardos, É. [1993]. An approximation algorithm for the generalized assignment problem, *Mathematical Programming* **62**(1-3): 461–474.

Sleator, D. [1980]. A 2.5 times optimal algorithm for packing in two dimensions, *Information Processing Letters* **10**(1): 37–40.

Steinberg, A. [1997]. A strip-packing algorithm with absolute performance bound 2, *SIAM J. Comput* **26**(2): 401–409.

Tang, S., Huang, Q., Li, X.-Y. and Wu, D. [2013]. Smoothing the energy consumption: Peak demand reduction in smart grid, *INFOCOM*, pp. 1133–1141.

Vazirani, V. V. [2001]. *Approximation algorithms*, Springer-Verlag.

Wiese, A. [2017]. A (1+epsilon)-approximation for unsplittable flow on a path in fixed-parameter running time, *ICALP*, Vol. 80, pp. 67:1–67:13.

Wiese, A. [2018]. Independent set of convex polygons: From n^ϵ to $1+\epsilon$ via shrinking, *Algorithmica* **80**(3): 918–934.

Williamson, D. P. and Shmoys, D. B. [2011]. *The Design of Approximation Algorithms*, 1st edn, Cambridge University Press.

Zukerman, M., Jia, L., Neame, T. and Woeginger, G. J. [2001]. A polynomially solvable special case of the unbounded knapsack problem, *Oper. Res. Lett.* **29**(1): 13–16.