On guillotine separability of squares and rectangles

Arindam Khan
Indian Institute of Science, Bengaluru, India
arindamkhan@iisc.ac.in

Madhusudhan Reddy Pittu [1]
Indian Institute of Technology, Kharagpur, India
pittumadhusudhan@iitkgp.ac.in

Abstract

Guillotine separability of rectangles has recently gained prominence in combinatorial optimization, computational geometry, and combinatorics. Consider a given large stock unit (say glass or wood) and we need to cut out a set of required rectangles from it. Many cutting technologies allow only end-to-end cuts called *guillotine cuts*. Guillotine cuts occur in *stages*. Each stage consists of either only vertical cuts or only horizontal cuts. In k-stage packing, the number of cuts to obtain each rectangle from the initial packing is at most k (plus an additional trimming step to separate the rectangle itself from a waste area). Pach and Tardos [20] studied the following question: Given a set of n axis-parallel rectangles (in the weighted case, each rectangle has an associated weight), cut out as many rectangles (resp. weight) as possible using a sequence of guillotine cuts. They provide a guillotine cutting sequence that recovers $1/(2\log n)$-fraction of rectangles (resp. weights). Abed et al. [1] claimed that a guillotine cutting sequence can recover a constant fraction for axis-parallel squares. They also conjectured that for any set of rectangles, there exists a sequence of axis-parallel guillotine cuts that recovers a constant fraction of rectangles. This conjecture, if true, would yield a combinatorial $O(1)$-approximation for *Maximum Independent Set of Rectangles (MISR)*, a long-standing open problem. We show the conjecture is not true, if we only allow $o(\log\log n)$ stages (resp. $o(\log n/\log\log n)$-stages for the weighted case). On the positive side, we show a simple $O(n\log n)$-time 2-stage cut sequence that recovers $1/(1+\log n)$-fraction of rectangles. We improve the extraction of squares by showing that 1/40-fraction (resp. 1/160 in the weighted case) of squares can be recovered using guillotine cuts. We also show $O(1)$-fraction of rectangles, even in the weighted case, can be recovered for many special cases of rectangles, e.g. fat (bounded width/height), δ-large (large in one of the dimensions), etc. We show that this implies $O(1)$-factor approximation for Maximum Weighted Independent Set of Rectangles, the weighted version of MISR, for these classes of rectangles.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Guillotine cuts, Rectangles, Squares, Packing, k-stage packing

1 Introduction

Cutting stock problem is a well-studied problem in combinatorial optimization, starting from the seminal work of Gilmore and Gomory [13]. Specially, the 2-D variant has received a lot of attention due to its application in practice [14, 27]. In these problems, we need to cut out some required geometric objects under some constraints, from a large source material such as glass, rubber, metal, wood or cloth. One special constraint, guillotine cut (end-to-end cut) emerges naturally from the design of cutting machines. Starting from the initial source material (*piece*), in each step such a cutting sequence takes one of the available pieces and finds an end-to-end cut along a straight line to divide it into two smaller pieces. Ultimately, each of the required objects corresponds to one of the final pieces. These cuts are simple to program due to column generation techniques [10, 26]. Due to lower cost and simple usability,

[1] A part of this work was done when the author was a Narendra undergraduate summer intern at IISc.

we find many real-world applications of guillotine cuts, such as in crepe-rubber mill [23], glass industry [22], and paper cutting [19].

The problem has also been studied extensively from theoretical viewpoint. Urrutia [25] asked the following question: Given a family of pairwise disjoint compact convex sets on a sheet of glass, is it true that one can always separate out a constant fraction of them using a guillotine cutting sequence? Pach and Tardos [20] answered the question in negative for line segments. They provided a family of n straight line segments where at most $O(n^{\log_3 2})$ line segments can be separated using guillotine cuts. However, they showed that one can always cut out a constant fraction of the input objects, if all objects have roughly the same size (contains a circle of radius r_1 and contained within a circle of radius r_2, for $r_2 \geq r_1 > 0$). For any family of n pairwise disjoint compact convex sets (resp. line segments) in the plane, one can always separate $\Omega(n^{1/3})$ (resp. $\Omega(n^{1/2})$) of members using guillotine cuts. For any family of n pairwise disjoint compact convex sets (resp. line segments) in the plane, one can always separate $\Omega(n^{1/3})$ (resp. $\Omega(n^{1/2})$) of members.

In this paper, we focus on guillotine separability of rectangles and squares. Given a set of n pairwise disjoint axis-parallel open rectangles $\mathcal{R} := \{R_1, R_2, \ldots, R_n\}$ embedded on a square $[0, N] \times [0, N]$, our goal is to separate as many rectangles as possible by a sequence of axis-parallel guillotine cuts. Pach and Tardos [20] showed that $\Omega(n/\log n)$ rectangles can be separated using guillotine cuts. Abed et al. [1] studied the problem for squares. They claimed [2] a guillotine cutting sequence that recovers 1/81-fraction of any set of axis-parallel squares and made the following conjecture:

▶ **Conjecture 1** ([1]). *For any set of n non-overlapping axis-parallel rectangles there is a guillotine cutting sequence with only axis-parallel cuts separating $\Omega(n)$ of them.*

Furthermore, they extend the problem to the weighted case in which each rectangle R_i has an associated weight $p_i \in \mathbb{Q}$ and the goal is to separate a subset of rectangles using guillotine cuts such that the total profit of separated rectangles is maximized. For this weighted version for squares, they claimed [2] recovery of 4/729-fraction of squares, and a $1/2^{O(d)}$-fraction in the weighted case in d-dimensions (where objects are hypercubes). They also showed that a proof of Conjecture 1 will imply an $O(1)$-approximation for maximum independent set of rectangles (MISR), a notoriously difficult NP-hard problem [2,8]. In maximum weighted independent set of rectangles (MWISR), we are given a set of possibly overlapping axis-parallel rectangles (with associated profit) and the goal to compute a non-overlapping subset of maximum profit. MISR is the cardinality variant, i.e., when all rectangles have equal profit. This connection between MISR and guillotine separability has made the guillotine separability to rise into prominence in recent years [18].

Guillotine cutting for rectangles can also be viewed as a packing problem where all rectangles are packed in such a way that all of them can be cut out using a guillotine cutting sequence. Gilmore and Gomory [13] initiated systematic study of guillotine packing by *k-stage packing*, where each stage consists of either horizontal or vertical guillotine cuts (but not both). Geometric packing problems are a classical well-studied area in approximation algorithms [9,15] [3] . In 2-D geometric knapsack problem (2GK) [16], we are given a set of rectangular items (with associated profit) and unit square knapsack, and the goal is to pack a subset of items into the knapsack maximizing the total profit. This problem is strongly NP-hard [17], even for squares with unit profit. The present-best approximation ratio is

[2] There are bugs in the claim. See Section C.
[3] See Section A for definitions related to approximation algorithms.

1.89 [12]. In 2-D strip packing problem (2SP) [11], we are given a set of rectangular items and fixed-width unbounded-height strip, and the goal is to pack all the items into the strip such that the height of the strip is minimized. Kenyon and Rémila gave an APTAS for the problem [14] using a 3-stage packing. In 2-D bin packing problem (2BP), we are given a set of rectangular items and unit square bins, and the goal is to pack all the items into minimum number of bins. The problem is APX-hard [3] and the present best approximation ratio is 1.405 [4].

All these problems have been studied under k-stage packing [21]. Abed et al. [1] have studied 2GK under guillotine cuts and have given a QPTAS for the cardinality case with quasi-polynomially bounded input. Caprara [6] obtained a 2-stage $T_\infty(\approx 1.691)$-approximation for 2BP, and conjectured that the worst-case asymptotic ratio between the optimal 2-stage 2BP and optimal general 2BP is 3/2. Later, Caprara et al. [7] gave an APTAS for 2-stage 2BP and 2-stage 2SP. Afterwards, Bansal et al. [5] showed an APTAS for guillotine 2BP. Seiden and Woeginger [24] gave an APTAS for guillotine 2SP. Both the APTAS for guillotine 2BP and guillotine 2SP are based on the fact that general guillotine 2BP or guillotine 2SP can be approximated arbitrary well by $O(1)$-stage packing, and such $O(1)$-stage packing can be found efficiently. Bansal et al. [4] conjectured the worst-case asymptotic ratio between the best guillotine 2BP and the best general 2BP is 4/3. This conjecture, if true, along with APTAS for guillotine packing [5], will imply a $(4/3+\varepsilon)$-approximation for 2BP.

1.1 Our contributions

We obtain improved guillotine separability for many classes of rectangles. We show a simple $O(n \log n)$-time algorithm that recovers $1/(\log n + 1)$-fraction of rectangles. The recursive algorithm of [20] can recover $1/(2\log n)$-fraction of rectangles, but can also take $\Omega(\log n)$-stages in the worst case, whereas our algorithm takes only 2-stages. We define multi-level lines called poles to partition the rectangles into guillotine separable classes and used this technique to recover a constant fraction of rectangles for many classes (see Section 5). Using ternary partitions we show a slightly improved bound of $n/\log_3(n+2)$. We then show that unlike other packing problems (such as 2BP or 2SP), in our problem any guillotine packing can not be approximated arbitrary well by $O(1)$-stages. In particular, we show:

▶ **Theorem 2.** *Any guillotine cutting algorithm with a constant number of stages can recover at most $O(\frac{\log\log n}{\log n})$ fraction of total weight. In order to recover a constant fraction of weight we require $\Omega(\frac{\log n}{\log\log n})$ stages.*

▶ **Theorem 3.** *Any guillotine cutting algorithm with a constant number of stages can extract at most $O(\frac{1}{\log\log n})$ fraction of total rectangles. In order to extract a constant fraction of rectangles, we require $\Omega(\log\log n)$ number of stages.*

For the case of squares, we found bugs in [1]. We could fix them by loosing additional multiplicative factor(see Section C) . Then using a more involved sampling and exploiting structural properties of guillotine cuts, we obtain further improvement. These structural properties might find usage in related guillotine packing problems.

▶ **Theorem 4.** *For axis-parallel squares, always there exists a guillotine cutting sequence that recovers $1/40$ (resp. $1/160$ in the weighted case)-fraction.*

We also show that $O(1)$-fraction of rectangles can be recovered for many special classes of rectangles, such as (a) *Fat*: when for each rectangle the ratio of width and height is in $[1/\beta, \beta]$, for some constant β. Fat objects generalize squares and are well-studied [20], (b) *δ-large*: when each rectangle has either width $\geq \delta N$ or height $\geq \delta N$. These rectangles are well-studied in the context of MISR [2], (c) *part-similar*, (d) *anti-laminar* (see Section 5). We also show that if $1/c$-fraction of weight can be recovered for a class of rectangles, then we obtain an $O(n^5)$-time c-approximation for MWISR. Thus obtaining $O(1)$-approximation for MWISR for the above classes of rectangles.

1.2 Organization of the paper

Section 2 describes some building blocks and known results used in this article. Section 3 gives improved guillotine separability of squares. Section 4 describes algorithms and hardness for rectangles. Section 5 studies several special classes of rectangles and obtains constant factor extraction for them.

2 Preliminaries

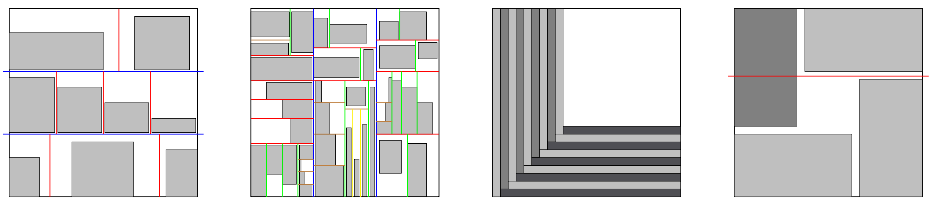
Let $\mathbb{Z}^+$ (resp. $\mathbb{Z}^{\geq 0}$) be the set of positive (resp. nonnegative) integers. Let us define $[n] := \{1, 2, \ldots, n\}$ and $[n \cup 0] := \{0, 1, 2, \ldots, n\}$, for $n \in \mathbb{Z}^+$. We are given a set of axis-parallel nonoverlapping rectangles $\mathcal{R} := \{R_1, R_2, \ldots, R_n\}$ embedded in a box $K := [0, N] \times [0, N]$. Each rectangle $R_i \in \mathcal{R}$ has width w_i, height h_i, and bottom-left corner at (x_i, y_i). Thus rectangle R_i is defined by the region $R(i) := [x_i, x_i + w_i] \times [y_i, y_i + h_i]$, where $x_i \geq 0, y_i \geq 0, x_i + w_i \leq N, y_i + h_i \leq N$. For any two input rectangles $R_i, R_j (j \neq i)$, we have $R(i) \cap R(j) = \emptyset$. Rectangle R_i has an associated weight p_i. In the unweighted case, all rectangles have the same (unit) weight.

2.1 Guillotine separability

▶ **Definition 1** (Piece). *A* piece *is an axis-aligned rectangular region with axis-parallel rectangles $\mathcal{R}$ embedded on it.*

▶ **Definition 2** (Guillotine cut). *A* guillotine cut *for a piece P is an end-to-end axis-parallel cut along a straight line ℓ dividing the piece into two (rectangular) subpieces P_1 and P_2.*

We define a *stage* of guillotine cuts as a set of end to end, equal, axis-parallel cuts separating piece P into further sub-pieces. Cuts in alternate stages alternate between vertical (parallel to y-axis) and horizontal (parallel to x-axis) cuts.



(a) 2-stage **(b)** 5-stage **(c)** many stage **(d)** non-guillotine

Figure 1 Different types of packing: (a), (b), (c) are guillotine packing, while (d) is not.

A guillotine cutting strategy is represented by two types of trees.

▶ **Definition 3** (Guillotine binary tree). A guillotine binary tree *for a set of rectangles $\mathcal{R}$ is a binary tree $\mathcal{T}_B$ where each non-leaf node $v \in V(\mathcal{T}_B)$ is equipped with a piece P_v and the straight line ℓ_v corresponding to the cut such that cutting P_v along the straight line ℓ_v gives us* *subpieces P_{v_1} and P_{v_2}, where v_1, v_2 are the children of v. Thus the piece P_r corresponding to the root node r contains $\mathcal{R}$ and for each leaf node v, the piece P_v contains only one rectangle. For each non-leaf node v, let $cutrec(v)$ (resp. $cutwt(v)$) denote the number (resp. total weight) of rectangles cut by line ℓ_v.*

▶ **Definition 4** (Guillotine stage tree). *A guillotine stage cutting strategy for a set of rectangles $\mathcal{R}$ is represented by a* guillotine stage tree *$\mathcal{T}_S$ where each non-leaf node $v \in V(\mathcal{T}_S)$ is equipped with a piece P_v and the set of straight lines $\mathcal{L}_v$ corresponding to the stage of the cut such that cutting P_v along the straight lines in $\mathcal{L}_v$ gives us subpieces $P_{v_1}, P_{v_2}, \ldots, P_{v_{|\mathcal{L}_v|}}$, where $v_1, v_2, \ldots, v_{|\mathcal{L}_v|}$ are the children of v. For each non-leaf node $v \in V(\mathcal{T}_S)$, let $cutrec(v)$ (resp. $cutwt(v)$) denote the number (resp. weights) of rectangles cut by lines in $\mathcal{L}_v$. A guillotine cutting strategy has k-stages if the tree $\mathcal{T}_S$ has height k.*

We say that $\mathcal{T}_B$ (resp. $\mathcal{T}_S$) separates a set of rectangles $\mathcal{R}$ if $cutrec(v) = 0$ for all $v \in V(\mathcal{T}_B)$ (resp. $\mathcal{T}_S$).

▶ **Definition 5** (k-good cut). *A cut ℓ_v cutting P_v is a k-good cut, if it intersects at most k rectangles and each side contains at least one rectangle completely.*

▶ **Definition 6** ((k, c)-good cut). *For any piece P having at least $2c$ rectangles, a (k, c)-good cut is a cut which cuts at most k rectangle and also has at least c rectangles on both sides.*

▶ **Definition 7** (Extraction factor). *A guillotine cutting strategy is said to have an extraction factor f if it separates f-fraction of rectangles (resp. weight).*

2.2 Grid sampling for squares

Here we briefly discuss grid sampling from [1] (but modify slightly to fix the bugs) when all input rectangles in $\mathcal{R}$ are squares. Let $len(i)$ be the side length of square R_i. Abed et al. [1] introduced a collection of (multi-level) grid lines to do a four step sampling to show that a constant fraction of squares are guillotine separable. First, we define these gridlines and related notions as they constitute an important component in our proofs too. A square $R_i \in \mathcal{R}$ is said to be at level-i if $len(i) \in \left(\frac{N}{2^{i+1}}, \frac{N}{2^i}\right]$. We independently pick random numbers $\tilde{x}, \tilde{y} \in [0, N)$ to define a random shift for drawing the grid. The vertical grid lines at level-i are drawn at $\tilde{x}, \tilde{x} + \frac{N}{2^i}, \tilde{x} + \frac{2N}{2^i}, \cdots$. Similarly, the horizontal grid lines at level-i are drawn at $\tilde{y}, \tilde{y} + \frac{N}{2^i}, \tilde{y} + \frac{2N}{2^i}, \cdots$ (Note that the coordinates mentioned for the vertical and horizontal grid lines are taken *modulo* N). Grid cells circumscribed by successive grid lines at level-$(i-1)$ are said to be at level-i. Thus the level-i grid cells are square regions of size $\frac{2N}{2^i} \times \frac{2N}{2^i}$. Hence, if a level-$j$ square is completely contained within a level-i grid cell, then $i \leq j + 1$. Note that higher the level, more fine grained the grid is, and smaller the grid cells and squares are.

In the *first* step of sampling, we pick squares from either even levels or odd levels, randomly. Let $\mathcal{R}_0$ be the set of squares remaining afterwards. Then $\mathbb{P}[R \in \mathcal{R}_0] = 1/2$.

In the *second* step of sampling, a square $R \in \mathcal{R}_0$ is removed if it intersects with grid lines in the level below it, i.e., any level-i ($i \geq 1$) square R is removed if it intersects a gridline of level $0, \ldots, i-1$. Let $\mathcal{R}_1$ be the set of squares that remained after this step.

▶ **Lemma 5.** *[1] A level-i square $R \in \mathcal{R}_0$ of side length $len(R) \in \left(\frac{N}{2^{i+1}}, \frac{N}{2^i}\right]$ remains in $\mathcal{R}_1$ with probability $(1 - \mu_R)^2 \geq 1/4$, where $\mu_R = len(R) \cdot \frac{2^{i-1}}{N}$*

In the *third* sampling step, the squares in $\mathcal{R}_1$ are sampled to obtain $\mathcal{R}_2$ so that each level-i grid cell contains at most one square of level-i. Let $\mathcal{R}_1^C$ denote the subset of level-i

squares contained within a grid cell C at level-i. We sample each $R \in \mathcal{R}_1^C$ with probability $\{(1-\mu_R)^2 \cdot M_C)^{-1}$ for $M_C = \sum_{S \in \mathcal{R}_1^C} (1-\mu_S)^{-2}$. Let $\mathcal{R}_2$ be the set of squares remaining after this process. Then the probability that a level-i square R remains in $\mathcal{R}_2$ is: $\mathbb{P}[R \in \mathcal{R}_0] \cdot \mathbb{P}[R \in \mathcal{R}_1 | R \in \mathcal{R}_0] \cdot \mathbb{P}[R \in \mathcal{R}_2 | R \in \mathcal{R}_1] = \frac{1}{2} \cdot (1-\mu_R)^2 \cdot \left(\frac{1}{(1-\mu_R)^2 \cdot M_C}\right) = 1/(2 \cdot M_C)$

In the *fourth* step, squares in $\mathcal{R}_2$ are sampled further to obtain a guillotine separable set.

▶ **Definition 8** (*ε-guillotine sampling*). *An ε-*guillotine sampling *for objects $\mathcal{O}$ is a distribution $\mathcal{D} : 2^{\mathcal{O}} \to [0,1]$ such that any object $r \in \mathcal{O}$ is sampled by $\mathcal{D}$ with probability at least ε and each subset in the support of $\mathcal{D}$ is guillotine separable.*

▶ **Lemma 6.** *[1] For any set of objects $\mathcal{O}$, the following two statements are equivalent:*
(i) there is an ε-guillotine sampling for $\mathcal{O}$,
(ii) for any $w : \mathcal{O} \to \mathbb{Z}_{\geq 0}$, there is a guillotine seperable subset $\mathcal{O}' \subseteq \mathcal{O}$ with $w(\mathcal{O}') \geq \varepsilon \cdot w(\mathcal{O})$.

Let *level*(R) be the level of square R. For a level-i square R, let *cell*(R) be the level-i gridcell containing R. We say that two squares R and S are *conflicting* if either R overlaps the boundary of *cell*(S) or S overlaps the boundary of *cell*(R). Note that if R overlaps the boundary of *cell*(S), then *level*$(R) <$ *level*(S). A conflict graph H encodes the conflict structures between squares, where the vertex set $V(H)$ corresponds to the squares in $\mathcal{R}$, and there is an edge between squares R and S if and only if R and S are conflicting.

▶ **Lemma 7.** *[1] For an independent set $I \subseteq V(H)$, $\{R\}_{R \in I}$ are guillotine separable.*

Let H be the conflict graph defined by the squares in $\mathcal{R}_2$. If H is χ-colorable, then we obtain a guillotine separable set of size at least $|\mathcal{R}_2|/\chi$. Abed et al. [1] showed that H is 9-colorable and showed $M_C \leq 81/4$. This shows $\frac{1}{9 \cdot (2M_C)} \geq \frac{2}{729}$-fraction of squares are guillotine separable. For the unweighted case, they claimed an improved bound of 1/81, by exploiting the structure of the tree representing the binary cutting strategy. However, there are bugs in both the proofs of weighted and unweighted cases. We present the bugs and the fixes in Section C.

3 Improved guillotine separability of squares

In this subsection, we prove extraction for unweighted and weighted squares to be $\frac{1}{40}$ and $\frac{1}{160}$, respectively. First we prove the following structural property to showing 5-colorability of conflict graph H. We say that a set of squares cover the edges of a level-l grid cell if the level-l cell has atleast one level-l square inside it and every square in the set intersects atleast one of the edges of the grid cell.

▶ **Lemma 8.** *The edges of any level-l grid cell can be covered by at most 4 squares after the second sampling.*

Proof. Other than squares covering the corners, there can not be any squares on the edges because any square on the edge has to be of level $\leq l-1$ because it intersects a level $l-1$ grid line. But since by first sampling we picked either odd or even parity levels the square on the edge has to be of level $\leq l-2$ which implies that the side of this square exceeds the side length of the level-l grid cell. ◀

▶ **Lemma 9.** *The conflict graph H is 5-colorable.*

Proof. Using similar proof as in [1], we prove this by induction on the number of vertices (squares). The base case (when there is only one square) is obvious. When there are at least two squares, consider the smallest square and the squares adjacent to it in the graph H. By Lemma 8, the vertex corresponding to smallest square has degree at most 4. Now assuming the graph without this vertex is 5-colorable, we can add this vertex and color it with one of the available 5-colors because the degree of this vertex is at most four. So inductively it is proven that the graph H is 5-colorable. ◀

From Lemma 7, the squares representing any independent set of graph H are guillotine separable. The first three (modified) steps of sampling selects every square with probability at least $\varepsilon = 2/81$. Now by sampling the five independent sets from H, uniformly at random, we obtain a set of squares which is ε-guillotine samplable for $\varepsilon = \frac{2}{81} \cdot \frac{1}{5} = 2/405$. From Lemma 6, we obtain extraction factor of 2/405.

3.1 Further improvement in the unweighted case

We can see that the initial set of squares were sampled thrice after picking a parity of levels and drawing a random grid so that no square of level-l was intersected by a grid line of level less than l and every level-l grid cell has at most one square of level-l inside it. We shall do the third sampling a bit differently. After we draw random grid lines and remove all the squares intersected by grid lines of level lower than that of the square, we now allow up to 6 squares inside a grid cell. Let the set of squares that remained after the first sampling be $\mathcal{R}_1$. If the length of a square $R \in \mathcal{R}$ be $len(R) \in (N/2^{i+1}, N/2^i]$, then the probability that this square stays in $\mathcal{R}_1$ is $(1-\mu_R)^2$ where $\mu_R = len(R) \cdot 2^{i-1}/N$. Let $|C|$ denote the number of squares in cell C. Now for the third sampling let us sample each square from a cell C with a probability $\frac{\min(6,|C|)}{(1-\mu_R)^2 \cdot M_C}$, for $M_C = \sum\limits_{S \in C} \frac{1}{(1-\mu_S)^2}$. Let the set of squares that remained after this third sampling be $\mathcal{R'}_2$. Now the probability that a square remains in $\mathcal{R'}_2$ after the third sampling can be written as: $\mathbb{P}[R \in \mathcal{R}_0] \cdot \mathbb{P}[R \in \mathcal{R}_1 | R \in \mathcal{R}_0] \cdot \mathbb{P}[R \in \mathcal{R}_2 | R \in \mathcal{R}_1]$ $= \frac{1}{2} \cdot (1-\mu_R)^2 \cdot \left(\frac{\min(6,|C|)}{(1-\mu_R)^2 \cdot M_C}\right) = \frac{1}{2} \cdot \frac{\min(6,|C|)}{M_C}$.

▷ Claim 9. $M_C / \min(6, |C|) \leq 4$

Proof. Let us find the maximum possible value of $\sum\limits_{r \in C} \frac{1}{(1-\mu_R)^2 \cdot \min(6,|C|)}$ with $\mu_R \in (1/4, 1/2]$.

As $len(R) > len(C)/4$, we have $|C| \leq 9$. As the sum of the areas of squares in C is less than that of the area of C, we have $\sum\limits_{R \in C} {\mu_R}^2 \leq 1$. For the case when $|C| \leq 6$, we have the sum to be at most $\frac{|C|}{(1-1/2)^2 \cdot \min(6,|C|)} = 4$. For $|C| \geq 7$, we use the fact from [1] that the function above is maximized for a given $|C|$ when all μ_R are equal. Thus a simple calculation gives that the maximum occurs when $|C| = 9$ and the value of the summation is equal to $81/24 < 4$. ◀

Therefore, at the end of this sampling, each square is left with probability at least 1/8. Now we consider further properties of guillotine cuts to obtain a better guillotine separable set.

▶ **Definition 10** (T-cut). *A T-cut is constituted by two axis-parallel line segments ℓ_A and ℓ_B such that one of the end points of ℓ_B lies on ℓ_A and $\ell_A \perp \ell_B$.*

A set of rectangles is said to be intersected by a T-cut if each rectangle in the set has a non-empty intersection with the T-cut. The following observations will be helpful in proving the existence of good cuts.

▶ **Observation 1.** *Any set of rectangles intersected by a T-cut is guillotine separable.*

Proof. W.l.o.g. assume in the T-cut ℓ_B is vertical and the bottom endpoint is lying on horizontal line segment ℓ_A. Consider the cut along the bottom edge of the topmost rectangle that is intersected by ℓ_B. If this line is a guillotine cut then we are done, otherwise consider the line aligned with the edge of the rectangle that got intersected first by this line. This has to be a guillotine cut otherwise will contradict our assumption about the top most rectangle. ◀

▶ **Observation 2.** *If there are at most 3 rectangles in a piece, they are guillotine separable.*

Proof. If there are no axis-parallel line intersecting at least two rectangles, the rectangles are guillotine separable by guillotine cuts along the edges of rectangles. Otherwise, if there exists such a line ℓ, we can extend a perpendicular line segment to ℓ that intersects the third remaining rectangle. This will form a T-cut and the proof follows from Observation 2. ◀

▶ **Observation 3.** *If there are at most n (≤ 6) rectangles in a piece, then we can separate at least $(n-1)$ squares by a sequence of guillotine cuts.*

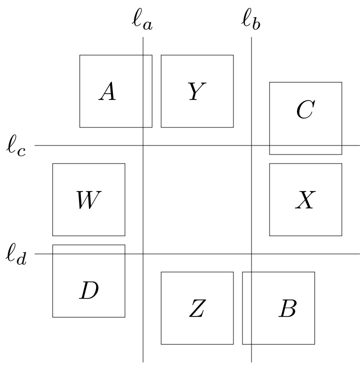
Proof. W.l.o.g. assume that there is no 0-good cut in the piece.
Case 1: There is a 1-good cut ℓ. Let R be the square intersected by ℓ. If there are at most three rectangles on both sides of this cut then we can extract all rectangles, except the one cut by ℓ, due to Observation 2. The other case is when there is one rectangle on one side and four on the other side. If those four rectangles are guillotine separable then we are done. Otherwise, w.l.o.g. assume ℓ is vertical and these four rectangles are on the right side of ℓ and a single rectangles R' is on the left side of ℓ. Let ℓ_1 be the line aligned with the right edge of R'. Then ℓ_1 should intersect R as there are no 0-good cuts. W.l.o.g. assume R lies above R'. Let ℓ_2 be the line aligned with the bottom edge of R and ℓ_3 be the line aligned with the left most edge of the first rectangle (from the left) that ℓ_2 intersects. We can see that ℓ_3 is again a 1-good cut and cut along ℓ_2 after cutting along ℓ_3 is a 0-good cut. These two cuts seperated the piece into three parts with atmost three rectangles in each sub-piece by cutting atmost one rectangle. By Observation 2 we are done.
Case 2: There is no 1-good cut in the piece. Let ℓ_1 be the line aligned with the left most right edge of the rectangles. This line has at least two rectangles on it as there is no 1-good cut. If we have at least four rectangles intersected by ℓ_1 then we are done because we can form a T-cut with all rectangles except possibly one rectangles and by Observation 1 we can extract all of them. Now consider if we have exactly three rectangles on ℓ_1. Then consider the line ℓ_2 that is aligned with the bottom edge of the top most rectangle on ℓ_1. This line should have at least two rectangles on it by our assumption which will again lead to a T containing five rectangles and we are done. Now suppose ℓ_1 has exactly two rectangles on it and ℓ_2 also has exactly two rectangles on it. Let ℓ_3 be the line aligned with the left edge of the first rectangle intersected by ℓ_2. For ℓ_3 to also intersect two rectangles, one of the rectangles have to be the lower rectangle that ℓ_1 intersects. However this implies that the cut along the top edge of the bottom rectangle on ℓ_1 is a 1-good cut which is a contradiction. ◀

▶ **Observation 4.** *If there are at most 10 rectangles, then there exists a $(4,1)$-good cut.*

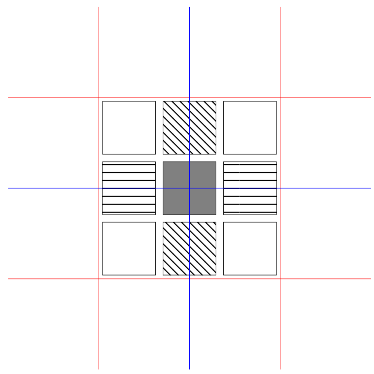
Proof. Consider the median cut that has almost equal number of rectangles on each side and let the line corresponding to the cut be ℓ_1. Suppose if the cut has at least six rectangles on it then any cut separating these rectangles cuts at most four rectangles and hence is a $(4,1)$ good cut. If ℓ_1 cuts at most four then it itself is a $(4,1)$ good cut. The only case is when it has exactly five rectangles on it. Consider every cut separating these five rectangles. If none of these cuts are $(4,1)$-good cuts then every cut has to intersect all other remaining rectangles. This implies that there is one guillotine stage cut separating all the five rectangles on ℓ_1 from the rest. ◀

▶ **Observation 5.** *If there are at most* 12 *squares with level less than* l *in a piece* P *and at least* 7 *squares lie on a grid line of level-*$(l-1)$ *then there exists a* $(6,2)$*-good cut in* P.

Proof. W.l.o.g. assume that the level-$(l-1)$ gridline ℓ_h that is intersecting at least 7 squares, is horizontal. Consider the leftmost level-$(l-1)$ vertical line ℓ_v with at least 2 squares on its left. We now prove that ℓ_v is a $(6,2)$-good cut. Note that this vertical line can cut at most 6 squares because there are a total of at most 12 squares. Let ℓ'_v be the vertical line of level-$(l-1)$ that is immediately to the left of ℓ_v. Then ℓ'_v can have at most 1 square to the left of it by definition of line ℓ_v. We know that there can be at most 1 square of level $< l$ in the space between ℓ_v and ℓ'_v. Thus there are at most 3 squares to the left of line ℓ_v. Therefore, we obtain ℓ_v to be a $(6,2)$-good cut. ◀



(a) Different classes of squares after sampling in the unweighted case.



(b) Different third sampling for weighted case (squares with diagonal, horizontal, full, and no shading belong to classes X, Y, Z, W, respectively)

Figure 2

▶ **Lemma 10.** *Using the new sampling scheme, every sampled instance in the piece (and the following subpieces) admits either a* $(4,1)$*-good cut or a* $(6,2)$*-good cut.*

Proof. For contradiction, assume that there are no $(4,1)$ and $(6,2)$-good cuts. Let P' be the considered piece. Now there are two cases:
Case 1: P' *has at least* 13 *squares.* Label the largest 13 squares as $R_1, R_2, \ldots, R_{12}, R_{13}$ in non-increasing order of size. Consider the grid cell C_{13} of square R_{13} (say, of level l). Let the left, right, top and bottom grid lines forming the edges of C_{13} be $\ell_a, \ell_b, \ell_c, \ell_d$, respectively (see Figure 2a). Let us also label the set of squares that are intersected by $\ell_a, \ell_b, \ell_c, \ell_d$ as A, B, C, D, respectively. Also let us label the set of squares lying completely to the left of line ℓ_a as W, right of line ℓ_b as X, above line ℓ_c as Y, and below line ℓ_d as Z. We observe that at least one of the sets W, X, Y, Z are non empty because there can be at most six squares inside C_{13} after the sampling, and at most four squares are in the set $(A \cup B \cup C \cup D) \setminus (W \cup X \cup Y \cup Z)$ (squares intersecting the edges of the cell). W.l.o.g. let the largest non-empty set be W.

First, we consider the case when $|W| = 1$. As we don't have $(4,1)$-good cuts, ℓ_a intersects at least 5 squares. So, $|Y \cup Z| \geq 2$. However, $|Y|, |Z| \leq |W|$. Hence, $|Y| = 1$ and $|Z| = 1$. Then both the lines ℓ_c and ℓ_d should intersect 5 squares each. Which implies $|X| \geq 2$ which is a contradiction to our assumption that $|W| = 1$ is the largest among $|W|, |X|, |Y|, |Z|$. Now we consider the case when $|W| \geq 2$. If there is any square on the right of ℓ_a other than R_{13} then consider ℓ_a can not intersect ≤ 6 squares, as then we obtain a $(6,2)$-good cut. On the other hand if it intersects at least 7 squares, then from Observation 5 we have a $(6,2)$ good cut which is a contradiction. So let us consider the case when there are no squares on the right of ℓ_a except R_{13}. As there are no $(4,1)$-good cuts, ℓ_a should intersect at least 5 squares. Since we have at least 5 squares on ℓ_a, $|Y \cup Z| \geq 2$. W.l.o.g let us assume $|Y| \geq 1$. Then it implies again that there have to be at least 5 squares on line ℓ_c and all these squares are on or to the left of line ℓ_a. Using similar arguments as in proof of Observation 5, let ℓ_p be the left most level-$(l-1)$ vertical grid line that has at least two squares on its left. Note that $\ell_p \neq \ell_a$ and thus has at least two squares on its right. Then ℓ_p must intersect ≥ 7 squares as there are no $(6,2)$-good cuts. However, then by observation 5, we will again have a $(6,2)$-good cut, giving a contradiction.
Case 2: P' has at most 12 squares. Consider any two squares and draw a line ℓ_1 that separates both these squares. The number of squares lying on this line have to be at least 5 by our assumption that there is no $(4,1)$-good cut. Now draw a line ℓ_2 perpendicular to ℓ_1 which has at least two squares on both its sides out of the 5 squares that were lying on ℓ_1. The line ℓ_2 should have at least 7 squares on it to avoid being the $(6,2)$-good cut. Now consider the line ℓ_3 perpendicular to ℓ_2 that has at least 2 squares on each side of ℓ_3 out of the 7 squares lying on l_2. Line ℓ_3 also should have at least 7 squares lying on it by the previous argument. Two perpendicular lines each having 7 squares lying on them guarantee a total of at least 13 squares in total. This is a contradiction. ◀

Using these observations and several other properties of guillotine cuts, we show existence of good cuts in the sample instance.

▶ **Theorem 11.** *Given a set of n squares obtained after the sampling, we can find a subset of at least $n/40$ squares that are guillotine separable.*

Proof. Using Lemma 10, we define a guillotine cutting sequence on a piece using only $(4,1)$ and $(6,2)$-good cuts until each subpiece has 6 or fewer rectangles. Then if the subpiece has 4, 5 or 6 rectangles and is not guillotine separable, we use Observation 3 to separate them, cutting at most one rectangle. This whole cutting strategy can be represented by a binary tree with internal nodes storing the number of squares that were killed by the cut. Each leaf node contains guillotine separable squares. Let f_1, f_2 be the number of leaf nodes containing one square and greater than one square, respectively. By the property of binary tree we know that the number of internal nodes is $f_1 + f_2 - 1$. Let v_i be a leaf node and v_j be its parent node. Now if v_i has one square in it and v_j corresponds to a $(4,1)$-good cut, then v_j has at most 4 squares in it. If v_i was obtained using Observation 3 then its parent node v_j again has only one square in it. Now assume that there is an internal node v_a which has two leaf children v_b, v_c, and both v_b and v_c have exactly one square in each of them. Then as v_a can have at most four squares, the three nodes v_a, v_b, v_c have at most six squares in total. By our assumption then we would have used the strategy as defined in Observation 3. This gives a contradiction. Thus every internal node can have at most one leaf child which has 1 square in it. This observation tells us that there are at least f_1 internal nodes each of which correspond to at most 4 squares. The maximum number of rectangles killed is at most

$4f_1 + 6(f_2 - 1)$ and at least $f_1 + 2f_2$ squares are extracted. So the fraction of squares that are saved is at least $\frac{f_1+2f_2}{5f_1+8f_2-6} \geq 1/5$. This implies an overall extraction factor of $\frac{1}{8} \cdot \frac{1}{5} = \frac{1}{40}$. ◀

3.2 Further improvement in the weighted case

From Claim 9, The probability for a square to survive after the second sampling is at least 1/8. Now we will divide these squares into four groups and select a group uniformly at random. Every level-l grid cell contains at most six squares of level-l due to the property of second sampling. Now, consider a particular cell C, then assume ℓ_v, ℓ_h be the vertical line and the horizontal line, respectively, that bisect the cell (see Figure 2b). Note that ℓ_v (resp. ℓ_h) belongs to the vertical (resp. horizontal) gridlines of level-(l). Now we define four sets. Let W_C be the set of squares in C that does not intersect either of ℓ_v, ℓ_h; X_C be the set of squares in C that intersects only ℓ_v, but not ℓ_h; Y_C be the set of squares in C that intersects only ℓ_h, but not ℓ_v; and Z_C be the set of squares in C that intersects both ℓ_h and ℓ_v. Let W (resp. X, Y, Z) be the set of all squares of type W_C for all cells C in the grid decomposition. We select one of these sets W, X, Y, Z uniformly at random. Let $\mathcal{R}_3$ be the squares that survive. Then each square will survive with probability $\geq \frac{1}{8} \times \frac{1}{4} = \frac{1}{32}$.

Now we will look at the conflict graph H of $\mathcal{R}_3$ and prove that the independent sets of H are actually guillotine separable.

▶ **Lemma 12.** *The independent set of squares obtained from the conflict graph H as defined above, is guillotine separable*

Proof. Let P be a piece obtained from such sampling as defined above. We will show the existence of a guillotine cut that does not cut any of the squares in P. Iteratively, this will show guillotine separability of all rectangles embedded on P. If we only have one square in our piece, then we are done. So let us assume that we have at least two squares. Now consider the two squares with the lowest levels. Let the squares be R_1 and R_2 and the levels to which they belong are l_1 and l_2, respectively. There are two cases:

Case 1: $l_1 = l_2$. Then we have two subcases.

In *subcase (a)*, they belong to *different* grid cells. Then we can separate them by cutting along one of the grid lines coinciding with one of the edges of grid cells containing one of the two considered squares. This line does not cut any of the other squares because the level of this line is lesser than every square in the piece (by definition of R_1 and R_2). Now let us apply this cutting procedure as long as we have a set of squares having the lowest level and are in different grid cells. At the end, we should obtain a piece in which the lowest level squares are either of different levels or they belong to the same level and same cell.

In *subcase (b)*, they belong to the *same* grid cell. As sampling S gives at most one square per grid cell, R_1, R_2 belong to one of the groups among W, X or Y. But we can then separate any two squares from the same group along one of the level-l grid lines. This level-l line does not intersect any of the squares in this piece as we do not have any level-l squares outside this cell. By the first sampling, this line can not intersect any higher level squares.

Case 2: $l_1 \neq l_2$. Then we can cut along the grid line coinciding with the edges of grid cell of the second largest square. This does not cut any of the smaller squares by definition of the first sampling. Also this line does not intersect the largest square by the definition of independent set in conflict graph. ◀

Now we are ready to prove the final theorem of this section.

▶ **Theorem 13.** *Given a set of axis-parallel weighted squares embedded in a plane, there is always a guillotine cutting sequence that recovers $\frac{1}{160}$-fraction of weights.*

Proof. As the smallest square can have at most four neighbors after the first sampling step, inductively we can show the conflict graph to be 5-colorable. Hence, from Lemma 12 we can conclude that any set of weighted squares is ε-guillotine sampleable for $\varepsilon = 1/32 \times 1/5$. ◀

4 Extraction of rectangles

Using standard techniques from [1], we can assume that all rectangles are embedded in a $2n \times 2n$ rectangular box with all corners of rectangles having integral coordinates in $[2n \cup 0] \times [2n \cup 0]$. When we refer to the width or height of these rectangles we refer to the width or height of those rectangles in this embedding. W.l.o.g. assume that $\log n \in \mathbb{Z}$. Let us define some horizontal lines called *poles* and give an attribute to each rectangle called *level*. A set of *poles at level-i* is defined as equally spaced horizontal lines with y-coordinates $\{\frac{(2k+1)\times n}{2^{i-1}} \mid k \in [(2^{i-1}-1) \cup 0]\}$. A level-0 line has y-coordinate either 0 or $2n$. The *level of a rectangle* is defined as the smallest level i such that some pole at level i intersects the rectangle. The union of all poles from levels 1 to i divides the plane into 2^i equal partitions which we will call as the *grid-partition of level i*. Let $\mathcal{R}_i$ be the set of rectangles present in i'th level for $i \in [\log n \cup 0]$. Further, assume that $\alpha_i = |\mathcal{R}_i|$.

Now we will use poles to show that we can partition all the input rectangles into $\log n + 1$ groups such that the embedding of rectangles in each group is guillotine separable.

▶ **Theorem 14.** *Given a set of rectangles (possibly weighted) $\mathcal{R}$ embedded in a square, $\frac{1}{\log n+1}$ fraction of total rectangles (resp., weight) can be extracted using 2-stage cuts.*

Proof. Rectangles in $\mathcal{R}_i$ are 2-stage separable, for any $i \in [\log n + 1]$. The first stage consists of cuts along the poles of level $\leq i-1$. These cuts divide the plain into 2^i equal partitions without cutting rectangles in $\mathcal{R}_i$. In each of the partitions, all rectangles are intersected by a pole of level-i and thus no vertical line within a partition can intersect two rectangles. Hence, the second stage can separate all rectangles by vertical cuts. As $\mathcal{R}_i$'s partition $\mathcal{R}$, taking $\mathcal{R}_i$ with the maximum cardinality (resp. weight) gives extraction factor of $\frac{1}{\log n+1}$. ◀

Using a k-ary partition, we can gain even further.

▶ **Lemma 15.** *We can extract $n/\log_3(n+2)$ rectangles by a series of guillotine cuts.*

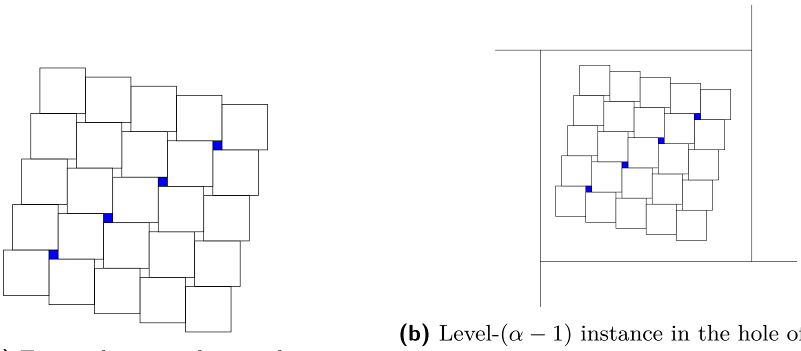
Proof. We prove this by induction on n. The base case is trivial. Consider a T-cut partitioning the plane into three parts with almost equal number of rectangles on each part. Suppose the number of rectangles on the T is greater than $\frac{n}{\log_3(n+2)}$ then we can just extract all of it using Observation 1. Otherwise, by induction on each of the small parts we can extract $\frac{n-\frac{n}{\log_3(n+2)}}{\log_3(n+2-\frac{n}{\log_3(n+2)})-1} \geq \frac{n-\frac{n}{\log_3(n+2)}}{\log_3(n+2)-1} = n/\log_3(n+2)$. ◀

For general rectangles, we will show that we can not obtain much better extraction factor using $O(1)$-stage cuts. However, in Section 5, we use poling arguments to obtain $O(1)$-extraction factor for many special cases of rectangles.

4.1 Hardness for k-stage algorithms in weighted case

In this section, we will prove Theorem 2.

▶ **Definition 11** ([1]). *A $k \times k$ unit square skew-grid is defined as a set of k^2 unit squares arraged in k rows and k columns numbered from bottom to top and left to right, respectively. With the bottom left coordinates of the square belonging to i^{th} row and j^{th} column as*



(a) External square skew-grid

(b) Level-$(\alpha - 1)$ instance in the hole of Level-α instance.

Figure 3 Level-α square skew-grid

$(j + i \cdot \varepsilon, i - j \cdot \varepsilon)$ *for* $i, j \in [(k-1) \cup 0], \varepsilon \leq \frac{1}{k}$. *A square in* i^{th} *row and* j^{th} *column is said to have* location *(i,j). The gaps formed by four adjacent squares are called holes. Those holes that are formed by the four squares with location* $(i,i), (i+1,i+1), (i+1,i), (i,i+1)$ *for* $i \in [0 \leq [(k-2) \cup 0]$ *are called diagonal holes and they are indexed by* i.

▶ **Observation 6.** *Let* $\mathcal{S}$ *be a* $k \times k$ *square skew-grid. Consider a set of* $p \in [(k-1) \cup 0]$ *vertical lines passing through different diagonal holes of* $\mathcal{S}$ *partitioning the plane into* $p+1$ *sub-pieces and in each of these pieces consider some horizontal lines through different diagonal holes (each hole has at most one line passing through it) with total number of horizontal lines* $q \in [(k-1) \cup 0]$. *This set of* $p+q$ *lines intersect a total of at least* $k \cdot p + q$ *squares.*

Proof. We can see that every vertical line through j^{th} diagonal hole intersects all squares with location $(i+1, j)$, $j \leq i$ and $(i, j+1)$, $i \leq j$. So all the p vertical lines intersect a total of $p \cdot k$ squares and none of these squares have location (i,i). Every horizontal line through j^{th} diagonal hole intersects the square with location (j,j), which gives a total of atleast q diagonal squares that get intersected by the horizontal lines. ◀

▶ **Definition 12.** *A level-*α $k \times k$ *square skew-grid (see Figure 3) is defined as a* $k \times k$ *square skew-grid with each of the* $k-1$ *diagonal holes having a level-*$(\alpha-1)$ $k \times k$ *skew-grid in it scaled appropriately to fit inside it. A level-*0 *square skew-grid is defined to be empty. We also say that a square belongs to level-*i *if it is among the largest squares in the level-*i $k \times k$ *square skew-grid that it is contained in.*

Now we show that for level-α $k \times k$ square skew-grid extraction factor is bounded based on the number of stages used. We assign weights uniformly to squares of same level so that the total weight of each level is 1. This implies the overall weight of this instance is α.

▶ **Theorem 16.** *Let* $f_c(\alpha)$ *be the best extraction factor for any* c*-stage extraction algorithm for level-*α $k \times k$ *square skew-grid instance. Then,* $f_c(\alpha) \leq \frac{\alpha}{k} + c - 1$.

Proof. We prove it by induction on $c + \alpha$. As we can extract at most one square out of k in each column by slicing vertically, we have $f_1(\alpha) \leq \alpha/k$ and $f_c(1) \leq 1$. Thus for $c = 1$ and ($c \geq 2$ and $\alpha = 1$), the claim is true. This proves the base case: $c + \alpha = 2$. Let us prove the inductive step from $c + \alpha$ to $c + \alpha + 1$. Let us assume that for every c, α with $c + \alpha \leq s$, the bound on $f_c(\alpha)$ is true. Now we will prove this for every $(c, \alpha+1)$ with $c + \alpha = s$.

Consider the best c-stage maximum weight extraction cutting sequence for a level-$(\alpha+1)$ instance. On every diagonal hole of it, a cutting sequence of stage $\leq c$ is induced. So considering an $(\alpha+1)$-stage configuration, suppose that the number of holes on which an i-stage cutting ($i \le c$)) is induced is h_i. One observation is that the function $f_c(\alpha)$ is monotonic for a fixed α (extraction factor should be non decreasing by increasing stages). Thus using Observation 6,

$$f_c(\alpha+1) \le \frac{k^2 - k \times h_c - h_{c-1}}{k^2} + \sum_{i=1}^{c} \frac{h_i}{k-1} \times f_i(\alpha) \tag{1}$$

$$\le \frac{k^2 - k \times h_c - h_{c-1}}{k^2} + \sum_{i=1}^{c} \frac{h_i}{k-1} \times \left(\frac{\alpha}{k} + i - 1\right) \tag{2}$$

$$\le 1 + h_c \times \left(\frac{\frac{\alpha}{k} + c - 1}{k-1} - \frac{1}{k}\right) + \sum_{i=1}^{c-1} h_i \times \left(\frac{\frac{\alpha}{k} + c - 2}{k-1} - \frac{1}{k^2}\right) \tag{3}$$

$$\le 1 + h_c \times \left(\frac{\frac{\alpha}{k} + c - 1}{k-1} - \frac{1}{k}\right) + (k - 1 - h_c) \times \left(\frac{\frac{\alpha}{k} + c - 2}{k-1} - \frac{1}{k^2}\right) \tag{4}$$

$$\le 1 + h_c \times \left(\frac{1}{k(k-1)} + \frac{1}{k^2}\right) + \frac{\alpha}{k} + c - 2 - \frac{k-1}{k^2} \tag{5}$$

$$\le \frac{\alpha+1}{k} + c - 1 \tag{6}$$

The first term in RHS of (1) is the maximum total weight of level-α squares that can be extracted, following from Observation 6. The second summation term in (1) is the total weight extracted through the i^{th} stage cutting sequence induced on the diagonal holes over all $i \in [c]$. In (2) we can substitute $\frac{\alpha}{k} + i - 1$ in place of $f_i(\alpha)$ by inductive assumption. We get (3) from (2) by rearranging terms and replacing i for all $i \in [c-1]$ by $c-1$. Since $\sum_{i=1}^{c} h_i \le k-1$ and also $\left(\frac{\frac{\alpha}{k}+c-2}{k-1} - \frac{1}{k^2}\right) \ge 0$, we can replace $\sum_{i=1}^{c-1} h_i$ by $k-1-h_c$ in (3). ◀

▶ **Observation 7.** *The total number of squares n in a level-α $k \times k$ square skew-grid is $k^2 \cdot \sum_{i=0}^{\alpha-1}(k-1)^i$. It follows that $(k-1)^{\alpha+1} \le n \le k^{\alpha+1}$.*

So it follows from the result that the maximum extraction fraction using any c-stage algorithm is $f_c(\alpha)/\alpha \le 1/k + (c-1)/\alpha$ which can be bounded by $n^{-\frac{1}{\alpha+1}} + \frac{c-1}{\alpha}$, using Observation 7. Taking $\alpha = \frac{\log n}{\log\log n} - 1$, we get the total extraction factor as $\frac{1}{\log n} + \frac{(c-1)\cdot\log\log n}{\log n - \log\log n}$.

This concludes the proof of Theorem 2. Note that this does not disprove Conjecture 1 for $\Omega(n)$ stages as this instance already admits $O(1)$-extraction factor for $\Omega(n)$ stages.

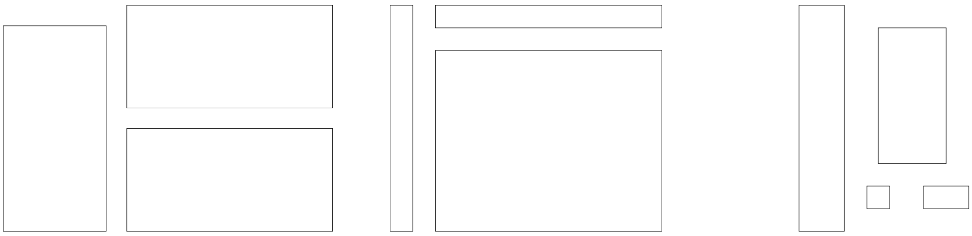
For the unweighted case (Theorem 3), a more involved analysis shows that any guillotine cutting algorithm with a constant number of stages can extract at most $O(\frac{1}{\log\log n})$ fraction of total rectangles. See Section D.1 for the proof of Theorem 3. We also obtain following hardness for d-dimensions, where cuts are axis-parallel hyperplanes. See Section D.2 for more details.

We also consider guillotine separability of d-dimensional axis parallel disjoint hyper cubes. The proof of the following theorem is deferred to the full version.

▶ **Theorem 17.** *There exist a family of d-dimensional axis parallel disjoint hyper cubes for which the guillotine extraction factor is asymptotically upper bounded by $\frac{1}{2\cdot(d-1)}$.*

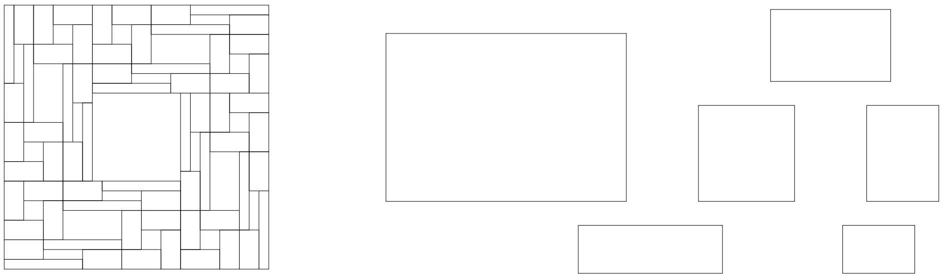
5 Constant extraction-factor for special classes

First, let us show a connection between the weighted case of guillotine separability of rectangles and MWISR (see Section E.1 for details).



(a) fat rectangles ($\beta = 2$) **(b)** δ-large rectangles **(c)** part-simillar instance ($\gamma = 2$)

Figure 4 special cases



(a) balanced instance (every sub piece has a balanced $\sqrt{k}$ cut) **(b)** anti-laminar instance (no rectangles is x-contained in another)

Figure 5 special cases

▶ **Theorem 18.** *If there is a guillotine extraction algorithm which guarantees at least* $1/\alpha$ *fraction of the total weight, then we have a* $O(\alpha)$ *approximation algorithm for the MWISR problem that runs with running time* $O(n^5)$*.*

Now we show $\Omega(1)$-extraction factor for special classes of rectangles. Using Theorem 18, we obtain $O(1)$-approximation for MISR (also MWISR, except for the balanced instance) for these classes. For omitted proofs of this section, see Appendix E.

5.1 Fat rectangles

A rectangle i is fat if $\max_i(\frac{w_i}{h_i}, \frac{h_i}{w_i}) \leq \beta$, where β is a constant (see Figure 4a). We divide the rectangles into two sets: (i) $w \geq h$ and (ii) $w \leq h$, and pick the one with maximum weight. W.l.o.g assume this to be the former set. We generalize the grid sampling techniques for squares and randomly pick a size class modulo $(\log \beta + 2)$ among widths and modulo 2 among heights. Then we exploit properties of this sampling to extract a constant fraction.

▶ **Lemma 19.** *If a piece has* n *rectangles with* $\max_i(\frac{w_i}{h_i}, \frac{h_i}{w_i}) \leq \beta, \beta \geq 1$*, then there is a guillotine cutting strategy with extraction factor* $\frac{1}{1584 \cdot (\log \beta + 2)}$.

5.2 δ-large rectangles

A rectangle i is δ-large if either $w_i \geq \delta N$ or $h_i \geq \delta N$, where $0 < \delta < 1$ (see Figure 4b). By using the poles that we have defined in Section 4, we obtain $O(1)$ extraction factor.

▶ **Lemma 20.** *Given an embedding of a set of* δ*-large rectangles in an* $N \times N$ *square, we can extract* $1/(\log(1/\delta) + 1)$ *fraction of rectangles (resp. weights) using 2-stage cuts.*

5.3 Part-similar instance

An instance is *part-similar* if $\min(\frac{\max_i h_i}{\min_i h_i}, \frac{\max_i w_i}{\min_i w_i}) = \gamma$, where γ is a constant, i.e., rectangles have either similar width or similar height (see Figure 4c). Using random position for poles, we find an interesting relation between the level of pole a rectangle belongs to and the size class of rectangle along the dimension perpendicular to the pole. This bounds the range of poles that any rectangle can belong to and we extract from one of the levels in that range.

▶ **Lemma 21.** *Given a set of rectangles with* $\min(\frac{\max_i h_i}{\min_i h_i}, \frac{\max_i w_i}{\min_i w_i}) = \gamma$*, there exists a* 2*-stage guillotine cutting sequence with extraction factor* $\frac{1}{2\times(\lfloor\log\gamma\rfloor+3)}$.

5.4 Balanced instance

Balanced instances are instances where in any sub-piece with k rectangles, there exists a cut that cuts $c(k)$ rectangles ($c(k)$ should be either $\Omega(k)$ or $O(k^{1-\varepsilon})$, where $0 < \varepsilon < 1$), dividing the remaining rectangles in a balanced way such that the ratio of number of rectangles on both sides is at most a constant $r \geq 1$. We show a constant factor $O(1)$-extraction factor for these instances. This instance includes many probable candidates for worst-case examples, including the skew-grid (see Figure 5a).

▶ **Lemma 22.** *For a fixed* $\alpha = 1 + \frac{1}{r}, r \geq 1, 0 \leq \varepsilon < 1, 0 < f$*, if a configuration has the property that for every large sub-piece of* k *rectangles, there is a cut that cuts either at least* $k \cdot f_1$ *rectangles or at most* $k^{1-\varepsilon} \cdot f$ *rectangles and also divide rectangles into two sets such that the ratio of number of rectangles on the two sides is at most* r*, then we can extract* $n \cdot f_1$ *fraction of rectangles for* $f_1 = e^{-\frac{f\cdot\alpha^{-\varepsilon}}{(1-f\cdot\alpha^{-\varepsilon})(1-\alpha^{-\varepsilon})}}, f \leq \alpha^{\varepsilon}$.

5.5 Anti-laminar instance

A rectangle i is said to be x-contained (resp. y-contained) in another rectangle j if $x_j < x_i < x_i + w_i < x_j + w_j$ (resp. $y_j < y_i < y_i + h_i < y_j + h_j$). A set of rectangles is said to be x-containment free (resp. y-containment free) if no rectangle is x-contained (resp. y-contained) in other. A set of rectangles is *anti-laminar* if it is either x-containment free or y-containment free (see Figure 5b).

▶ **Lemma 23.** *For an anti-laminar instance, we always have an extraction factor* $1/2$.

6 Conclusion

We have made progress towards understanding guillotine separability of rectangles. We showed that Pach-Tardos conjecture is not true, even for squares, if we use $o(\log\log n)$-stages. However, if we allow $\Omega(n)$ stages, even with general rectangles we were unable to find any instance where we can not recover more than $n/2$ rectangles. The balanced instance (see Figure 5a) or its variants are probable candidates for such hard instances. However, we showed that we can still separate a constant fraction of rectangles from these instances. It is interesting to obtain guillotine separability of even $(\log\log n/\log n)$-fraction of rectangles. This will give an $O(n^5)$-time algorithms for MWISR, matching the present best approximation guarantee. Apart from the existential questions, an interesting problem is to find a polynomial-time $O(1)$-approximation algorithm to recover rectangles through a sequence of guillotine cuts.

Acknowledgements. We thank Arnab Maiti and Amatya Sharma for helpful discussions.

References

1 Fidaa Abed, Parinya Chalermsook, José R Correa, Andreas Karrenbauer, Pablo Pérez-Lantero, José A Soto, and Andreas Wiese. On guillotine cutting sequences. In *APPROX*, pages 1–19, 2015.

2 Anna Adamaszek and Andreas Wiese. Approximation schemes for maximum weight independent set of rectangles. In *FOCS*, pages 400–409, 2013.

3 Nikhil Bansal, Jose R Correa, Claire Kenyon, and Maxim Sviridenko. Bin packing in multiple dimensions: inapproximability results and approximation schemes. *Mathematics of Operations Research*, 31:31–49, 2006.

4 Nikhil Bansal and Arindam Khan. Improved approximation algorithm for two-dimensional bin packing. In *SODA*, pages 13–25, 2014.

5 Nikhil Bansal, Andrea Lodi, and Maxim Sviridenko. A tale of two dimensional bin packing. In *FOCS*, pages 657–666, 2005.

6 Alberto Caprara. Packing 2-dimensional bins in harmony. In *FOCS*, pages 490–499, 2002.

7 Alberto Caprara, Andrea Lodi, and Michele Monaci. Fast approximation schemes for two-stage, two-dimensional bin packing. *Mathematics of Operations Research*, 30(1):150–172, 2005.

8 Parinya Chalermsook. Coloring and maximum independent set of rectangles. In *APPROX*, pages 123–134. Springer, 2011.

9 Henrik I Christensen, Arindam Khan, Sebastian Pokutta, and Prasad Tetali. Approximation and online algorithms for multidimensional bin packing: A survey. *Computer Science Review*, 24:63–79, 2017.

10 JM Valério de Carvalho and AJ Guimaraes Rodrigues. An lp-based approach to a two-stage cutting stock problem. *European Journal of Operational Research*, 84(3):580–589, 1995.

11 Waldo Gálvez, Fabrizio Grandoni, Afrouz Jabal Ameli, Klaus Jansen, Arindam Khan, and Malin Rau. A tight $(3/2+\varepsilon)$ approximation for skewed strip packing. *To appear in APPROX*, 2020.

12 Waldo Gálvez, Fabrizio Grandoni, Sandy Heydrich, Salvatore Ingala, Arindam Khan, and Andreas Wiese. Approximating geometric knapsack via l-packings. In *FOCS*, pages 260–271, 2017.

13 PC Gilmore and Ralph E Gomory. Multistage cutting stock problems of two and more dimensions. *Operations research*, 13(1):94–120, 1965.

14 Claire Kenyon and Eric Rémila. A near-optimal solution to a two-dimensional cutting stock problem. *Mathematics of Operations Research*, 25(4):645–656, 2000.

15 Arindam Khan. *Approximation algorithms for multidimensional bin packing*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2016.

16 Arindam Khan, Arnab Maiti, and Amatya Sharma. On guillotine separable packing of two-dimensional knapsack. *Manuscript*, 2020.

17 Joseph YT Leung, Tommy W Tam, CS Wong, Gilbert H Young, and Francis YL Chin. Packing squares into a square. *Journal of Parallel and Distributed Computing*, 10(3):271–275, 1990.

18 Sahil Mahajan. Guillotine cuts. Master's thesis, IIIT Delhi, 2016.

19 Michael L McHale and Roshan P Shah. Cutting the guillotine down to size. *PC AI*, 13:24–26, 1999.

20 János Pach and Gábor Tardos. Cutting glass. In *Proceedings of the 16th Annual Symposium on Computational Geometry*, pages 360–369. ACM, 2000.

21 Enrico Pietrobuoni. *Two-dimensional bin packing problem with guillotine restrictions*. PhD thesis, University of Bologna, Italy, 2015.

22 Jakob Puchinger, Günther R Raidl, and Gabriele Koller. Solving a real-world glass cutting problem. In *European Conference on Evolutionary Computation in Combinatorial Optimization*, pages 165–176. Springer, 2004.

23 W Schneider. Trim-loss minimization in a crepe-rubber mill; optimal solution versus heuristic in the 2 (3)-dimensional case. *European Journal of Operational Research*, 34(3):273–281, 1988.

24 Steven S Seiden and Gerhard J Woeginger. The two-dimensional cutting stock problem revisited. *Mathematical Programming*, 102(3):519–530, 2005.
25 Jorge Urrutia. Problem presented at accota'96: Combinatorial and computational aspects of optimization. *Topology, and Algebra, Taxco, Mexico*, 1996.
26 François Vanderbeck. A nested decomposition approach to a three-stage, two-dimensional cutting-stock problem. *Management Science*, 47(6):864–879, 2001.
27 PY Wang. Two algorithms for constrained two-dimensional cutting stock problems. *Operations Research*, 31(3):573–586, 1983.

A Approximation Algorithms

In this subsection, we define notions related to approximation algorithms.

▶ **Definition 13** (Approximation Guarantee). *For a minimization problem* Π*, an algorithm* $\mathcal{A}$ *has approximation guarantee of* α $(\alpha > 1)$*, if* $\mathcal{A}(I) \leq \alpha\ OPT(I)$ *for all input instance* I *of* Π*. For a maximization problem* Π'*, an algorithm* $\mathcal{A}$ *has approximation guarantee of* α $(\alpha > 1)$*, if* $OPT(I) \leq \alpha\ \mathcal{A}(I)$ *for all input instance* I *of* Π'*.*

▶ **Definition 14** (Asymptotic Approximation Guarantee). *For a minimization problem* Π*, an algorithm* $\mathcal{A}$ *has asymptotic approximation guarantee of* α $(\alpha > 1)$*, if* $\mathcal{A}(I) \leq \alpha\ OPT(I) + o(OPT(I))$ *for all input instance* I *of* Π*.*

▶ **Definition 15** (Polynomial Time Approximation Scheme (PTAS)). *A minimization problem* Π *admits PTAS if for every constant* $\varepsilon > 0$*, there exists a* $(1+\varepsilon)$*-approximation algorithm with running time* $O(n^{f(1/\varepsilon)})$*, for any function* f *that depends only on* ε*.*

If the running time of a PTAS is $O(f(1/\varepsilon)\ n^c)$ for some function f and a constant c that is independent of $1/\varepsilon$, we call it Efficient PTAS (EPTAS), If the running time of a PTAS is polynomial in both n and $1/\varepsilon$, we call it Fully PTAS (FPTAS). Quasi-polynomial time approximation scheme (QPTAS) and pseudo-polynomial time approximation scheme (PPTAS) are defined analogously as PTAS, however, their running times are quasi-polynomial (i.e., $n^{(\log n)^c}$ for some constant $c > 1$) and pseudo-polynomial time, respectively. Asymptotic analogue of PTAS, EPTAS, FPTAS are known as APTAS, AEPTAS, AFPTAS, respectively. We refer the reader to [9] for more on these approximation schemes and their connections with hardness assumptions.

B Omitted parts from Section 2

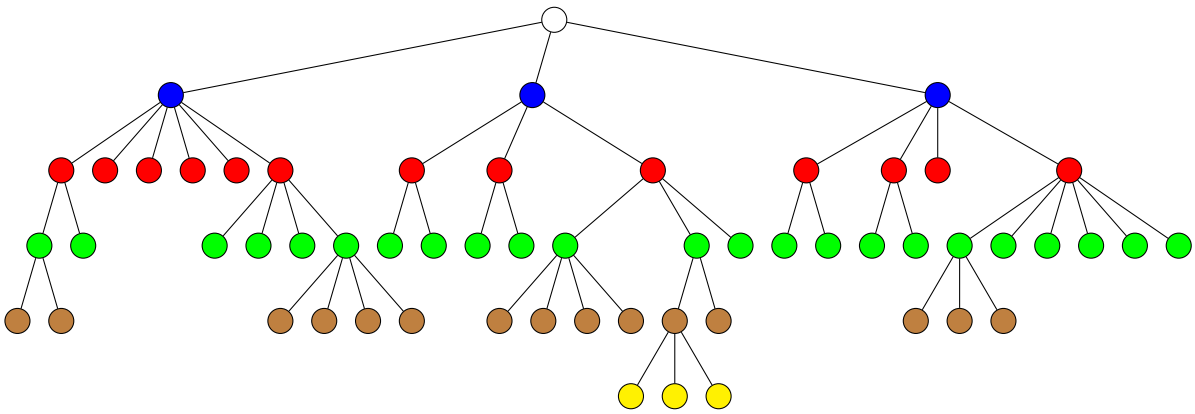
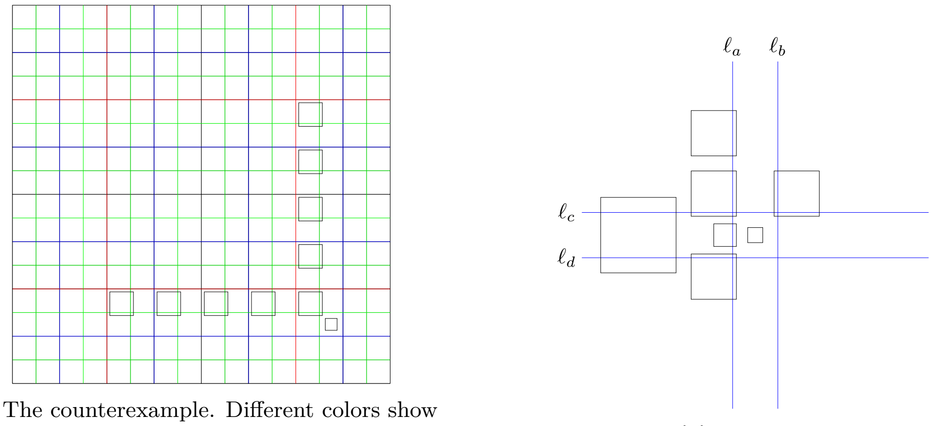


Figure 6 Guillotine stage tree for the 5-stage packing in Figure 1.

C Bugs in [1] and its fix

C.1 First bug in [1]

Abed et al. [1] claimed the following lemma:



(a) The counterexample. Different colors show different levels of gridlines.

(b) The Fix

Figure 7 Second bug and fix

▶ **Lemma 24** (Restatement of Lemma 12 in [1]). *Let $I \subseteq V(H)$ be an independent set. The squares $\{R\}_{R\in I}$ are guillotine separable.*

An important observation is that after the first and second sampling, as defined in [1], a level-$(i-1)$ square can still be inside a level-i grid cell which causes problems in the proof of Lemma 12 of [1] which is essential in proving the constant extraction-factor for the weighted case of squares. The problem is that if a level-i grid cell contains both a level-$(i-1)$ square and a level-i square then these squares may not get separated by the series of cuts used in Lemma 12 of [1].

C.2 A fix for this bug

The problem can be resolved simply by picking only odd parity levels or even parity levels as a first sampling procedure. This sampling guarantees that no level-$(i-1)$ square is present along with a level-i square at the cost of loosing an additional half factor in extraction.

C.3 Second bug in [1]

▶ **Lemma 25** (Restatement of Lemma 7 in [1]). *Let $\mathcal{R}_2$ be the set of rectangles in the piece P at the end of second step of sampling. Then there exists a 3-good cut.*

One of the subcases in the proof considers the case where there are at least 10 squares in a piece. They consider the squares as a sequence $S_1, S_2, \ldots$ ordered by non-increasing side lengths. Let i be the level of S_{10}. It is claimed that at least one of the grid lines forming the sides of the grid cell C of the tenth largest square intersects at most 3 squares and each of the created subpieces contain at least one square completely, proving the lemma. This claim and the lemma was central in the proof of the extraction factor 1/81 in the unweighted case. However, we show a counterexample where none of the gridlines intersect at most 3 squares so that each of the created subpieces contain at least one square. Note that this instance is still guillotine separable, however the gridlines separating S_{13} from other rectangles intersect five rectangles, thus are not 3-good cuts.

C.4 A fix for this bug

We present an alternative to fix the bug. For the case when there are less than ten squares in the piece, Lemma 7 of [1] already proved that a 3-good cut exists. So let us consider the case when there are more than six squares. Consider the seventh-largest square and the grid cell C it is present in. Let the left, right, top and bottom grid lines forming the edges of C_{13} be $\ell_a, \ell_b, \ell_c, \ell_d$, respectively (see Figure 2a). Let us label the set of squares belonging to the region left of grid line marked ℓ_a and not intersected by ℓ_a as W. Similarly we define X, Y, Z, respectively (right of ℓ_b, above ℓ_c, below ℓ_d, respectively). We also define the set of squares intersected by grid line ℓ_a as A, ℓ_b as B, ℓ_c as C, ℓ_d as D, respectively. we know that the 6 squares that are larger than the square in the grid cell have to belong to one or more of these sets. From Lemma 8, we have that the number of squares that can cover the edges of any grid cell is at most four, i.e., the maximum number of squares that can be in sets A, B, C, D but not in W, X, Y, Z are four. But having a total of six squares forces atleast one of the squares in, say w.l.o.g., set W. Now if the set A has three or fewer elements then the cut along ℓ_a is a 3-good cut. If set A has four or more elements then one of the squares in A will belong to Y or Z because the set $A \setminus (Y \cup Z)$ can accommodate at most two squares. So in the case when Y (or Z) has a square then ℓ_c or ℓ_d will be a 3-good cut because they separate one square and cut at most three squares (at most one from A, at most two not from A because A has at least four of the six squares). So for any number of squares in a given piece after second sampling, we must have a 3-good cut. Hence, guaranteeing a guillotine extraction factor of $1/162$.

D Omitted proofs from Section 4

D.1 Hardness for small stage algorithms in unweighted case

Here we prove Theorem 3.

▶ **Lemma 26** (Restatement of lemma 18 from [1]). *The number of squares of a $n \times n$ square skew-grid that can be seperated by a guillotine cutting strategy is at most $\lceil \frac{n^2+2n-2}{2} \rceil \leq \frac{n^2}{2} + n$*

In this subsection we show another example which is constructed almost similarly as in the weighted case such that the extraction factor is bounded based on the number of stages used. In the construction so that the number of squares in all levels is almost same.

▶ **Definition 16.** *For $k \geq 2$, A k-modified level-α square skew-grid is defined as a $\left(2^{2^{\alpha+k-1}}+1\right) \times \left(2^{2^{\alpha+k-1}}+1\right)$ square skew-grid with each of the $2^{2^{\alpha+k-1}}$ diagonal holes having a k-modified level-$(\alpha-1)$ square skew-grid in it scaled appropriately to fit inside it. A k-modified level-0 square skew-grid is defined to be empty. We also say that a square belongs to level-i if it is among the largest squares in the k-modified level-i square skew-grid that it is contained in.*

▶ **Lemma 27.** *Let $E_c(\alpha)$ be the maximum total number of squares that can be extracted using a c-stage cutting sequence on a level α configuration. Let $f_c(\alpha) = E_c(\alpha) \cdot 2^{-2^{k+\alpha}}$. Then $f_c(\alpha) \leq c - 1 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}})$.*

Proof. We prove it by induction on $c+\alpha$. Let us prove some boundary cases first and use inductive step to cover all the cases. For the case when $c = 1$, it is clear that

$$E_1(\alpha) = (2^{2^{\alpha+k-1}}+1) + 2^{2^{\alpha+k-1}} \times (2^{2^{\alpha+k-2}}+1) + 2^{2^{\alpha+k-1}} \times 2^{2^{\alpha+k-2}} \times (2^{2^{\alpha+k-3}}+1) \ldots (7)$$

which gives $f_1(\alpha) = \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}})$ which satisfies the bound we assumed. Now for the case when $c \geq 2$,$\alpha = 1$ we have

$$f_c(1) = E_c(1) \times 2^{-2^{1+k}} \leq 2^{-2^{1+k}} \times \left(\frac{(2^{2^k}+1)^2}{2} + 2^{2^k} + 1\right) \leq 1 \tag{8}$$

By using the upper bound given in Lemma 26 and the fact that $k \geq 2$, we get the above bound which also satisfies the bound for $f_c(\alpha)$ we assumed. We have proved the base case when $c + \alpha = 2$. Let us assume for every c, α with $c + \alpha \leq s$ the bound on $f_c(\alpha)$ is true. Let us prove that for every $(c, \alpha + 1)$ with $c + \alpha = s$ the bound on $f_c(\alpha + 1)$ is true (h_i is defined similarly as in the weighted case).

$$E_c(\alpha+1) \leq \left((2^{2^{\alpha+k}}+1)^2 - h_c \times (2^{2^{\alpha+k}}+1) - h_{c-1}\right) + \sum_{i=1}^{c} h_i \times E_i(\alpha) \tag{9}$$

$$\begin{aligned} f_c(\alpha+1) &\leq (2^{-2^{\alpha+k}}+1)^2 - (h_c \cdot 2^{-2^{\alpha+k}}) \times (2^{-2^{\alpha+k}}+1) - (h_{c-1} \cdot 2^{-2^{\alpha+k+1}}) \\ &\quad + \sum_{i=1}^{c} \left(h_i \cdot 2^{-2^{\alpha+k}}\right) \times f_i(\alpha) && (10) \\ &\leq (2^{-2^{\alpha+k}}+1)^2 + h'_c \times (f_c(\alpha) - 1 - 2^{-2^{\alpha+k}}) + h'_{c-1} \times (f_{c-1}(\alpha) - 2^{-2^{\alpha+k}}) \\ &\quad + \sum_{i=1}^{c-2} h'_i \cdot f_i(\alpha) && (11) \\ &\leq 1 + 2 \cdot 2^{-2^{\alpha+k}} + 2^{-2^{\alpha+k+1}} + h'_c \times (c - 1 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 1 - 2^{-2^{\alpha+k}}) \\ &\quad + h'_{c-1} \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) \\ &\quad + \sum_{i=1}^{c-2} h'_i \cdot (i - 1 + \sum_{j=1}^{\alpha}(2^{-2^{\alpha+k-j}} + 2^{-2^{\alpha+k+1-j}})) && (12) \\ &\leq 1 + 2 \cdot 2^{-2^{\alpha+k}} + 2^{-2^{\alpha+k+1}} + h'_c \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) \\ &\quad + h'_{c-1} \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) \\ &\quad + \sum_{i=1}^{c-2} h'_i \cdot ((c - 1 - 2^{-2^{\alpha+k}}) - 1 + \sum_{j=1}^{\alpha}(2^{-2^{\alpha+k-j}} + 2^{-2^{\alpha+k+1-j}})) && (13) \\ &\leq 1 + 2 \cdot 2^{-2^{\alpha+k}} + 2^{-2^{\alpha+k+1}} + h'_c \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) \\ &\quad + \sum_{i=1}^{c-1} h'_i \cdot ((c - 2 - 2^{-2^{\alpha+k}}) + \sum_{j=1}^{\alpha}(2^{-2^{\alpha+k-j}} + 2^{-2^{\alpha+k+1-j}})) && (14) \\ &\leq 1 + 2 \cdot 2^{-2^{\alpha+k}} + 2^{-2^{\alpha+k+1}} + h'_c \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) \\ &\quad + (1 - h'_c) \times (c - 2 + \sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) - 2^{-2^{\alpha+k}}) && (15) \end{aligned}$$

$$\Rightarrow f_c(\alpha+1) \leq c - 1 + \sum_{i=1}^{\alpha+1}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}}) \tag{16}$$

The inequality (9) is obtained similarly as in the weighted case. (10) is obtained from (9) by multiplying both sides by $2^{-2^{\alpha+k+1}}$ and then replacing every $E_i(\alpha) \cdot 2^{-2^{\alpha+k}}$ by $f_i(\alpha)$ for $1 \leq i \leq c$. (11) is obtained by replacing every term $h_i \cdot 2^{-2^{\alpha+k}}$ by h'_i for $1 \leq i \leq c$ and rearranging some terms. (12) is obtained from (11) by replacing every term $f_i(\alpha)$ by $i - 1 + \sum_{j=1}^{\alpha}(2^{-2^{\alpha+k-j}} + 2^{-2^{\alpha+k+1-j}})$. (13) is obtained from (12) by replacing every $1 \leq i \leq c-2$ by $c - 1 - 2^{-2^{\alpha+k}}$. (14) is obtained by just rearranging terms from Inequality 13. We also have that $\sum_{i=1}^{\alpha} h_i \leq 2^{2^{\alpha+k}}$ which implies $\sum_{i=1}^{\alpha} h'_i \leq 1$ and since the term $(c - 2 + \sum_{j=1}^{\alpha}(2^{-2^{\alpha+k-j}} + 2^{-2^{\alpha+k+1-j}}) - 2^{-2^{\alpha+k}}) \geq 0$, we can replace the term $\sum_{i=1}^{c-1} h'_i$ by $1 - h'_c$. Rearranging and cancelling out terms will complete the inductive step. ◀

Now we also have that the total number of squares $n = 2^{2^{\alpha+k}} \times \sum_{i=1}^{\alpha}(1 + 2^{-2^{k+i-1}})^2$. This gives the extraction fraction $E_c(\alpha)/n = f_c(\alpha)/\sum_{i=1}^{\alpha}(1 + 2^{-2^{k+i-1}})^2 \leq c-1+\sum_{i=1}^{\alpha}(2^{-2^{\alpha+k-i}} + 2^{-2^{\alpha+k+1-i}})/\sum_{i=1}^{\alpha}(1 + 2^{-2^{k+i-1}})^2 \leq c/\alpha$ (because the numerator is $< c$ and denominator $> \alpha$). Now in order to get a good upper bound we need to minimize c/α for fixed n, c which means we need to maximize α for fixed n. which happens when $k = 2$. So $n \approx \alpha \cdot 2^{2^{\alpha}}$. Thus α is in the order of $\log\log n$ and the upper bound c/α in the order of $c/\log\log n$. Which says we need at least order $\log\log n$ number of stages for any algorithm to guarantee a constant extraction factor even for squares case. Strangely the results we got say that using constant stages we cannot guarantee better than $\frac{\log n}{\log\log n}$ and $\log\log n$ factors for the weighted and unweighted cases, respectively, which matches the best-known approximation ratios for MISR.

D.2 Hardness results in d-dimension

▶ **Definition 17.** *A corner of a d-dimension hypercube is defined as the vertex of a hypercube with the least sum of all the d coordinates.*

▶ **Definition 18** (k^d hyper skew cube)**.** *For a given skew setting $\boldsymbol{A}$ with $\boldsymbol{A}_{i,j} = \pm 1$ when $i \neq j$ and $\boldsymbol{A}_{i,j} = -\boldsymbol{A}_{j,i}$, a k^d* hyper skew cube *is a set of k^d unit hyper cubes arranged so that every square indexed $(i_1, i_2, i_3, \ldots, i_d)$, $0 \leq i_1 \leq k-1, 0 \leq i_2 \leq k-1, \ldots, 0 \leq i_d \leq k-1$ has the coordinates of its corner as $\boldsymbol{p} = \boldsymbol{i} + \varepsilon \cdot \boldsymbol{A} \cdot \boldsymbol{i}$,$0 < \varepsilon \leq \frac{1}{k}$ ($\boldsymbol{i}$ and $\boldsymbol{p}$ are the vector representations). Also the corresponding k^d* hyper regular cube *is when every square indexed $(i_1, i_2, i_3 \ldots i_d), 0 \leq i_1 \leq k-1, 0 \leq i_2 \leq k-1, \ldots, 0 \leq i_d \leq k-1$ has the coordinates of its corner as $\boldsymbol{p} = \boldsymbol{i}$. A* sub k^d hyper skew cube *is a subset of hyper cubes from the k^d hyper skew cube and the corresponding sub k^d hyper regular cube is defined similarly as above.*

It is to be noted that the perimeter of a d-dimensional hypercube is the sum of lengths of all the edges which will be $d \cdot 2^{d-1}$. The hypercube is bounded by $2d$ hyperplanes of dimension $d-1$ which is just a hypercube of dimension $d-1$. So it is natural that we define the perimeter of a face as $\frac{d \cdot 2^{d-1}}{2 \cdot d} = 2^{d-2}$.

▶ **Definition 19.** *For a given sub k^d hyper skew cube instance q, we define $U(q)$ as the number of cubes in it and $P(q)$ as the sum of perimeters of all the faces of the corresponding sub k^d hyper regular cube that belong to a single hyper cube (only those faces that are exposed). Let $S(q)$ be the maximum number of hypercubes that are guillotine separable.*

▶ **Lemma 28.** *For a given sub k^d hyper skew cube instance q, $S(q) \leq \left\lceil \frac{2^{d-1} \cdot U(q) + P(q) - d \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil$.*

Proof. Let us prove this by induction on $U(q)$. The base step $U(q) = 1$ is clearly true. Suppose $U(q) \geq 2$ and consider the first optimal cut that separates q into q_1, q_2 and cuts r

items. We have $U(q_1) + U(q_2) + r = U(q)$ and $P(q) \leq P(q_1) + P(q_2) - 2 \cdot (2^{d-2} \cdot (r+1)) \leq$ $P(q) \leq P(q_1) + P(q_2) - 2 \cdot (2^{d-2} \cdot r)$.

$$
\begin{aligned}
S(q) &= S(q_1) + S(q_2) && (17)\\
&\leq \left\lceil \frac{2^{d-1} \cdot U(q_1) + P(q_1) - d \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil + \left\lceil \frac{2^{d-1} \cdot U(q_2) + P(q_2) - d \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil && (18)\\
&\leq \left\lceil \frac{2^{d-1} \cdot (U(q_1) + U(q_2)) + (P(q_1) + P(q_2)) - d \cdot 2^d + (d-1) \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil && (19)\\
&\leq \left\lceil \frac{2^{d-1} \cdot (U(q) - r) + (P(q) + (r+1) \cdot 2^{d-1}) - d \cdot 2^d + (d-1) \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil && (20)\\
&\leq \left\lceil \frac{2^{d-1} \cdot U(q) + P(q) - d \cdot 2^{d-1}}{(d-1) \cdot 2^d} \right\rceil && (21)
\end{aligned}
$$

Inequality (19) follows from (18) using the fact that $\lceil a/2 \rceil + \lceil b/2 \rceil \leq \lceil (a+b+1)/2 \rceil$. ◀

▶ **Theorem 29.** *There exist a family of d-dimensional axis parallel disjoint hyper cubes for which the guillotine extraction factor is asymptotically upper bounded by $\frac{1}{2 \cdot (d-1)}$.*

Proof. Follows directly from Lemma 28. ◀

It is to be noted that results in Section 4.1 can be extended to d-dimension easily using the k^d skew hyper cube. By careful analysis we get extraction factor upper bounded by $\frac{1}{\log n} + \frac{(c-1) \cdot \log \log n}{\log n - (d-1) \cdot \log \log n}$.

E Omitted proofs from Section 5

E.1 Connection between MWISR and weighted guillotine problem

In this subsection we prove Theorem 18. Note that Abed et al. [1] already showed a similar connection between MISR and the unweighted guillotine problem. We build on their approach to build a dynamic program to handle the weighted case.

We are given a set of n axis-parallel rectangles. We can assume that the corners of rectangles have integer coordinates in the range $\{0, \ldots, 2n\}$ and w.l.o.g no two rectangles are exactly coinciding (if such a case exists just consider the rectangle with the largest weight). Consider a piece P out of $O(n^4)$ such pieces possible in integer plane $[0, 2n] \times [0, 2n]$.
If P has no rectangle completely lying inside, we take the solution to be an empty set.
If $P = R$ (plane exactly coincides rectangle R), we take the maximum of the below two cases:
Case 1: Consider solution for P to be only R and discard all other rectangles inside R.
Case 2: Discard R and consider all rectangles inside R. Try all possibilities of dividing P into two smaller pieces using a horizontal or vertical guillotine cut such that the horizontal/vertical coordinates of this cut is an integer. Consider one such cut and let $P_1 \neq \emptyset \neq P_2$ denote the resulting pieces. The DP looks up the solutions for the cells representing P_1 and P_2 and combines them to a solution for P. It selects the cut yielding the optimal total profit from the resulting two subproblems. Let us define two tables $DPG[i][j][k][l]$ and $RECT[k_1][l_1][k_2][l_2]$. Where $DPG[i][j][k][l]$ stores the maximum weight guillotine separable independent set of rectangles in the piece having bottom left coordinates(k, l) and top right coordinates $(k+i, l+j)$ and $RECT[k_1][l_1][k_2][l_2]$ stores the weight of the rectangle in the input having bottom left and top right coordinates (k_1, l_1) and (k_2, l_2) respectively if such a

rectangle is present else it stores zero otherwise.

$$DPG[i][j][k][l] = \max\Big(\max_{1\le s\le j-1}(DPG[i][s][k][l] + DPG[i][j-s][k+s][l]),$$
$$\max_{1\le s\le i-1}\big(DPG[s][j][k][l] + DPG[i-s][j][k+s][l]\big), RECT[k][l][k+i][l+j]\Big).$$

With constraints $1 \le i \le 2n$, $1 \le j \le 2n$, $0 \le k \le 2n-i$, $0 \le l \le 2n-j$.
This DP gives an $O(n^5)$ time algorithm and returns the maximum weight guillotine separable independent set of rectangles.

E.2 Fat rectangles

We give proof of Lemma 19. Let us consider objects that have bounded $\frac{w}{h} \le \beta$, $\frac{h}{w} \le \beta$, $\beta \ge 1$ for any rectangle in $\mathcal{R}$. Previously, a square of side length $\ell \in (\frac{N}{2^{i+1}}, \frac{N}{2^i}]$ is said to belong to level i. Now similarly, for a rectangle with width w and height h, we say that its width w belongs to level i if $w \in (\frac{N}{2^{i+1}}, \frac{N}{2^i}]$ and height h belongs to level j if $h \in (\frac{N}{2^{j+1}}, \frac{N}{2^j}]$. Let us only consider that class of rectangles for which $h \ge w$ and we can assume w.l.o.g. that they are at least half in weight (resp. number). Now it follows that for every rectangle $j \le i$, since $\beta \ge \frac{h}{w} > \frac{\frac{N}{2^{j+1}}}{\frac{N}{2^i}} = 2^{i-j-1}$, we have $j \le i < j+1+\log\beta$. Now consider all the width classes and select all the classes that are congruent to some random $r \in [0, \log\beta+1]$ modulo $(2+\log\beta)$. Also randomly select either even or odd classes from the height class. The expected number(or weight) of rectangles that remain until now are $\frac{1}{4\cdot(2+\log\beta)}$. To this set of rectangles let us apply the sampling procedures that we applied for squares. In the first sampling, we remove all those rectangles that are intersected by vertical lines of level less than that of its width or horizontal lines of level less than that of its height. Which guarantees an expected number of $\frac{1}{4}$ number of rectangles that remained after our first two sampling procedures. In the next sampling procedure, we define a cell for a rectangle of level (i, j) (let us say for now informally that the level of a rectangle is (width class, height class) as the rectangle formed by the two pair of vertical and horizontal level $i-1$ and level $j-1$ lines that contain this rectangle. We want every cell of a rectangle to have at most one rectangle with level (i, j). It is an important observation that any rectangle with $i' < i$ or $j' < j$ cannot stay inside a level (i, j) cell, because $i' < i$ implies $i' \le i - (2+\log\beta) \le i-2$, which implies the width of this rectangle exceeds that of the cell. And $j' < j$ implies that $j' \le j-2$, which exceeds the height of the cell. A bug arises in the second sampling mentioned in [1] because a cell of level-i can have a level $i-1$ square inside it which causes problems in all the results that follow. Since there can be at most 9 rectangles of level (i, j) inside a level (i, j) cell. We select a square randomly. The total expected weight (resp. number) of squares that remain after the previous sampling is $\frac{1}{9}$. After these samplings we have an important property that we can use.

▶ **Observation 8.** *If a rectangle of level-(i', j') intersects one of the grid lines of a level (i, j) rectangle then $i' + j' < i + j$.*

Proof. Suppose the rectangle intersects the horizontal level $j-1$ line then we have $j' \le j-1 \le i-1$ and $i' < j' + \log\beta + 1 \le i + \log\beta$, which implies $i' \le i$, which then implies $i' + j' < i + j$. Suppose the rectangles intersects the vertical $i-1$ level lines then we have $i' \le i-1$, which implies $i' \le i - 2 - \log\beta < j-1$ and $j' \le i' \le i - 2 - \log\beta < j-1$, which then implies $i' + j' < i + j$. We can also see that the total number of rectangles that can intersect the boundary of this cell can be at most 10 (4 in corners, 0 on vertical edges because $j' < j - 1$, 3 each on horizontal edges). ◀

We can define the conflict graph H same as defined for squares and it is easy to see that H is 11 colorable. And also if we think $i + j$ as the level of a rectangle now it also follows similarly that independent set of H is guillotine separable. So combining all of this gives a $\frac{1}{1584\cdot(\log\beta+2)}$ extraction algorithm. Even when we have bounded $\frac{w_{max}}{w_{min}}$ or $\frac{h_{max}}{h_{min}}$ we can extract $O(\frac{1}{\log\beta})$ where β is the bound using the same grid sampling techniques. However, we will solve that problem with a different technique in the later subsection.

E.3 δ-large rectangles

Now we prove Lemma 20.

Proof. All rectangles will be intersected by some poles of level $\leq \lceil\log(1/\delta)\rceil$. Similar to proof of Theorem 14, we can select the level with the maximum number of rectangles (resp. weights) to extract $n/(\log(1/\delta) + 1)$ rectangles (resp. weights) using 2-stages of cuts. ◀

E.4 Part-similar instance

Here we prove Lemma 21.

Proof. Let us define the size classes for a rectangles based on their height. A rectangle belongs to size class i if its height is in the range $(\frac{2n}{2^{i+1}}, \frac{2n}{2^i}]$ for $i \in [\log 2n \cup 0]$.

Now let us choose a random $y \in [2n \cup 0]$ uniformly and shift all the horizontal poles by y wrapping up appropriately. Let us delete all those rectangles that are intersected by a pole of level less than its size class. The probability that a rectangle stays after this sampling is at least 1/2. Which implies that there exists a y such that more than half of the rectangles are not intersected by a line with level lesser than their size classes. Now if an object has height $> \frac{N}{2^{i+1}}$ then it has to belong to level $\leq i + 1$. This implies that after the random sampling, every rectangle from size class i belongs to level i or $i + 1$. Suppose we have a set of rectangles with $\frac{h_{max}}{h_{min}} = \gamma$. Let all the rectangles after the sampling be distributed over levels $[\alpha, \alpha + c - 1]$. The minimum size difference between any two rectangles in levels α and $\alpha + c - 1$ is $> 2^{c-3}$. So $\gamma > 2^{c-3}$ gives $c \leq \lfloor\log\gamma\rfloor + 3$. This guarantees a level with at least $\frac{n}{2\times(\lfloor\log\gamma\rfloor+3)}$ rectangles. ◀

E.5 Balanced instance

▶ **Lemma 30.** *For an instance of n rectangles, if in any sub-piece with k rectangles we have a cut that cuts at most $c(k)$ rectangles dividing the remaining rectangles in a balanced way such that the number of rectangles on both sides has ratio at most $r \geq 1$, then we have an extraction factor $f(n) \geq n \cdot e^{-\frac{\alpha}{\alpha - c(\alpha)}\cdot\sum_{i=1}^{\log_\alpha n}\frac{c(\alpha^i)}{\alpha^i}}$, where $\alpha = 1 + \frac{1}{r}$ (note that in this result $c(k)$ can be any function).*

Let us start with a simpler case where any sub-configuration obtained after few cuts has a $O(\sqrt{p})$ cut that divides the number of rectangles on each side almost equally, where p is the number of rectangles in that sub-configuration. Let $f(n) = n/g(n)$ be the number of rectangles that can be extracted by simply cutting using the available $O(\sqrt{n})$ cut. We have $f(n) \geq 2 \cdot f((n - \sqrt{n})/2)$. This implies $g(n) \leq (\frac{n}{n-\sqrt{n}}) \cdot g(\frac{n-\sqrt{n}}{2}) \leq (\frac{n}{n-\sqrt{n}}) \cdot g(\frac{n}{2})$ (since $g(n)$ is a non decreasing function). We have base cases $g(p) = 1, p \leq 3$. Using the above inequality recursively, we obtain the upper bound for $g(2^k)$ as $\prod_{i=1}^{k}(1-2^{-i/2})^{-1} = e^{-\sum_{i=1}^{k}\log(1-2^{-i/2})}$. Using $1-1/x \le \log(x)$ for $x \ge 0$, we obtain $g(2^k) \le e^{\sum_{i=1}^{k}\frac{2^{-i/2}}{1-2^{-i/2}}} \le e^{\sum_{i=1}^{k}\frac{2^{-i/2}}{1-2^{-1/2}}} \le e^{3\sqrt{2}+4}$. So $f(n) \ge n/e^{3\sqrt{2}+4}$.

Now we state the proof of Lemma 30, a more general case.

Proof. In balanced instance, in any sub-configuration with k rectangles we always have a cut which cuts at most $c(k)$ and divides the remaining rectangles in a balanced way such the number of rectangles on both sides are in some $r:1$ ratio with $r \ge 1$. Then we have $f(n) \ge f(\frac{r}{r+1}\cdot(n-c(n))) + f(\frac{1}{r+1}\cdot(c-c(n)))$. This after replacing $f(n)$ with $n/g(n)$ gives $\frac{n}{g(n)} \ge \frac{(\frac{r}{r+1})\cdot(n-c(n))}{g(\frac{r}{r+1}\cdot(n-c(n)))} + \frac{\frac{1}{r+1}\cdot(n-c(n))}{g(\frac{1}{r+1}\cdot(n-c(n)))}$. Now using the fact that the function $g(n)$ is non decreasing, we obtain $g(n) \le (1-\frac{c(n)}{n})^{-1}\cdot g(\frac{n}{1+\frac{1}{r}})$. Taking $\alpha = 1+1/r$ and applying the inequality recursively, we get that $g(n) \le e^{-\frac{\alpha}{\alpha-c(\alpha)}\cdot\sum_{i=1}^{\log_\alpha n}\frac{c(\alpha^i)}{\alpha^i}}$. Whenever $c(n) = O(n^{1-\varepsilon})$ and $\alpha = 1+\frac{1}{r}$ is constant, the value of $g(n)$ is bounded by a constant. For $c(n) = n^{1-\varepsilon}$ we get $f(n) \ge n\cdot e^{-\frac{\alpha^\varepsilon}{(\alpha^\varepsilon-1)^2}}$. This implies we can always extract a constant fraction of the number of rectangles. ◀

The proof of Lemma 22 follows from the proof of Lemma 30. Using Lemma 30, we can extract a constant fraction of items if $c(k)$ is either $\Omega(k)$ or $O(k^{1-\varepsilon})$. The idea is that if $c(k)$ is $\Omega(k)$ then we can extract all the rectangles that are cut and this is already a constant fraction. If $c(k)$ is $O(k^{1-\varepsilon})$ then we can cut and recurse into the subpieces.

E.6 Anti-laminar instance

Intuitively, a set of rectangles is *anti-laminar*, if along one of the axes, the intervals formed by projecting the rectangles on that axis have the property that no interval is completely inside another. First we show the proof of Lemma 31.

▶ **Lemma 31.** *Let I be a set of n intervals on the x-axis such that no two intervals are completely contained inside each other. Let $\{p_1,\cdots,p_{2n}\}$ be the endpoints of intervals in I. Then we can distribute weight n to the intervals (p_i,p_{i+1}) for $i\in[2n-1]$ such that each interval in I gets weight 1.*

Proof. Let us prove this by induction on the number of intervals. This is obviously true when there is only one interval. Let us sort the intervals by increasing value of the left end point and label them $1,\ldots,k+1$. Suppose the claim is true until $n=k$. Then for $n=k+1$ we can assume that the first interval intersects the second interval w.l.o.g (otherwise we can assign weights for intervals other than the first interval and assign unit weight to the first interval which completes the inductive step) and consider the left end point of second interval. Consider the k intervals to the right of this point and assign weights so that all this k intervals have the same weights. Now since the second interval is not contained in the first interval, the right end point of the second interval is to the right of the right end point of the first interval. We can just assign the weight contained in the region from the right end point of first interval to the right end point of the second interval in the region from the left end point of the first interval to the left endpoint of the second interval. Which makes the weight of first interval equal to that of the second and hence this gives equal weight to all the intervals. We can also observe that this is impossible when an interval is contained in another interval. ◀

▶ **Lemma 32** (Restatement of Lemma 13 in [1]). *For a set of rectangles with equal width (or height) there exists a guillotine cutting strategy that separates at least* $1/2$ *of the weight.*

Now we are ready to prove of Lemma 23.

Proof. Without loss of generality let us suppose that the projection of rectangles on x-axis satisfies the no containment property. By Lemma 31, we can assign weights to intervals on x-axis such that the total weight inside each interval is same. Now if we scale each strip associated with the intervals with a scaling factor equal to its weight then we get another set of rectangles where each rectangle has the same width. Now by lemma 32 we can separate $1/2$ the total weight of the given set of rectangles. ◀