

Improved Bounds on the Treedepth of Grid Graphs Using Continuous Cutting

Bachelor Thesis of

Wolf Heinrich Luttermann

At the Department of Informatics
Institute of Theoretical Informatics

Reviewers: Dr. rer. nat. Torsten Ueckerdt
TT-Prof. Dr. Thomas Bläsius
Advisor: Kolja Kühn

Time Period: 9th May 2026 – 9th September 2026

Statement of Authorship

Ich versichere wahrheitsgemäß, die Arbeit selbstständig verfasst, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde sowie die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Wolf Luttermann

Karlsruhe, September 9, 2026

Abstract

The treedepth of a graph is a well-studied parameter and closely related to the parameter treewidth. For the k -by- k grid it is known that the treewidth is exactly k but for the treedepth only a lower bound of $1.667k$ and an upper bound of $2.519k + o(k)$ are known. This thesis aims to improve the upper bound by introducing a new mathematical problem called the *continuous treedepth*. It is the geometric analog of treedepth for shapes in the plane and closely related to the treedepth of the k -by- k grid and its subsets. We prove that an upper bound on the continuous treedepth of a shape yields an upper bound on the treedepth of the corresponding grid graph. Using this, we improve the upper bound on the treedepth of the k -by- k grid from $2.519k + o(k)$ to $2.45k + \mathcal{O}(\log k)$. We believe that future research into the continuous treedepth may lead to interesting discoveries and has the potential to show even stronger bounds on the treedepth of grid graphs.

Deutsche Zusammenfassung

Die Baumtiefe eines Graphen ist ein viel untersuchter und gut verstandener Parameter, welcher eng mit der Baumweite verwandt ist. Für das k -mal- k -Gitter ist bekannt, dass die Baumweite bei genau k liegt, während für die Baumtiefe des Gitters nur eine untere Schranke von $1.667k$ und eine obere Schranke von $2.519k + o(k)$ bewiesen worden ist. Diese Arbeit widmet sich dem Problem, die obere Schranke zu verbessern. Dazu wird ein neues mathematisches Problem, die *kontinuierliche Baumtiefe*, formalisiert, welche den Parameter Baumtiefe auf Formen in der Ebene überträgt. Es wird gezeigt, dass eine obere Schranke der kontinuierlichen Baumtiefe einer Form für einen entsprechenden Gitter-Graphen eine obere Schranke seiner Baumtiefe impliziert. Auf diese Weise wird die obere Schranke des k -mal- k -Gitters von $2.519k + o(k)$ zu $2.45k + \mathcal{O}(\log k)$ verbessert. Die kontinuierliche Baumtiefe kann als eigenständiges Problem Teil zukünftiger Untersuchungen sein und ebenfalls der noch genaueren Bestimmung der Baumtiefe von Gitter-Graphen dienen.

Contents

1	Introduction	1
1.1	Related Work	2
1.2	Contributions	3
1.3	Outline	4
2	Preliminaries	5
2.1	Treedepth	6
2.2	Grid Graphs	6
2.3	Continuous Objects	6
3	Discrete Decomposition Trees	9
3.1	Definition	9
3.2	Correspondence of Elimination Forests and Decomposition Trees	10
4	Continuous Treedepth	13
4.1	Definition	13
4.2	Example and Existence	16
5	Connecting Continuous and Discrete Treedepth	19
5.1	The Lattice-Vertex Deletion Set of a Curve	19
5.2	Dividing Grid Graphs along Cuts	27
5.3	Transforming Continuous to Discrete Decomposition Trees	28
6	Upper Bounds	31
6.1	Shape and Grid Graph Definitions	31
6.2	Overview of the Best Upper Bounds	32
6.3	The Rotated Rectangle	32
6.4	The Wedge	35
6.5	The Triangle	39
6.6	The Square	41
6.7	Discrete Approach for the Rotated Rectangle	44
7	Conclusion	49
	Bibliography	51

1. Introduction

The *treedepth* of a graph G is a measure of how close the graph G is to a star-graph and it is closely related to the parameters *treewidth* and *pathwidth*. Treedepth has been studied under many different names, such as *minimum elimination tree height*, *vertex ranking number*, *ordered chromatic number* or *minimum number of colors in a centered coloring*. Each one gives a slightly different perspective on the parameter. From a theoretical perspective, treedepth plays a central role in structural graph theory, for example, in the study of graph sparsity [NdM12]. But it is also an important parameter in algorithmic research. For example, exploiting treedepth allows speeding up a number of graph algorithms [IOO18]. To get an intuition for treedepth we describe the elimination game, which is closely related to its characterization using elimination trees:

A graph is given, and we play a game in rounds. In each round the player is allowed to delete one vertex of each connected component of the current graph. The goal is to delete the entire graph in as few rounds as possible. The treedepth of the graph is the minimum number of rounds in which this game can be finished.

For example, consider the 4-by-4 grid in Figure 1.1. We present a possible sequence of deletions: The grid is connected, which means in the first round only a single vertex can be deleted. The sequence starts with the vertex labeled 1. This does not disconnect the graph, so the sequence continues deleting only one vertex per round: the vertex number 2 in round two, then 3 and finally 4 in the fourth round. The resulting graph is disconnected, making it possible to delete two vertices in each round. The sequence continues with the vertices labeled with number 5 in the fifth round and the two vertices with label 6 in the sixth round. Now, four connected components that each consist of two vertices are left. In two more rounds these components are deleted. This sequence proves that the treedepth of the 4-by-4 grid is at most 8.

This is not an optimal sequence. In fact there exists one that uses only 7 rounds, showing that the treedepth of the 4-by-4 grid is at most 7. For further elimination sequences of specific grids of small dimension, check out <https://howl-lovat.vercel.app/?access=howl-guest>. This website allows to play the elimination game on an m -by- n grid to show upper bounds on the treedepth of the graph for fixed m and n .

One might wonder how good or even optimal deletion sequences are constructed. Naturally, three things seem quite intuitive for a good decomposition:

- The vertices that are chosen in a connected component form a small separator.

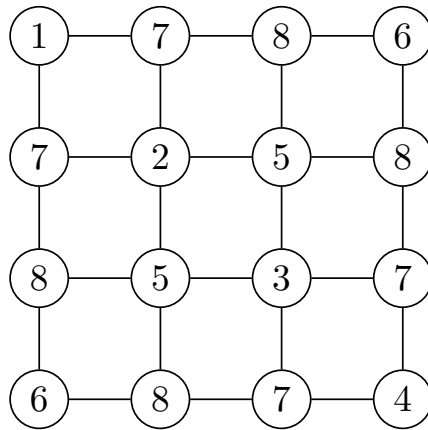


Figure 1.1: Elimination strategy for the 4-by-4 grid.

- These separators tend to be balanced.
- The leftover components after a separation can be cheaply separated again.

The first is plausible because as soon as a separator is deleted, more vertices can be deleted per round. So a small separator ensures that the graph can be deleted quicker. At the same time, the leftover components should all be small, so they themselves can be deleted quickly. This is why the separators tend to be balanced, so the separation does not lead to one little but another very big component. Finally, it is clear that after deleting a separator, the decomposition continues with the leftover components. So it is better if the leftover components are again easy to separate. We will come back to this intuition later, when providing a characterization of the treedepth using decomposition trees.

The type of graph that this thesis focuses on are grid graphs which are subsets of the k -by- k grid. This grid is particularly interesting in light of the following result:

Theorem 1.1. [KR18] *There exists a constant c such that every graph G of treedepth at least k^c contains one of the following minors:*

- the k -by- k grid
- the complete binary tree of height k
- the path on 2^k vertices

All of the three graphs have treedepth in $\Theta(k)$ and therefore the treedepth of G is polynomial in the treedepth of the minors. The treedepth of the path on 2^k vertices is $k + 1$ and for the complete binary tree of height k it is exactly k . This motivates the question, what the exact treedepth of the k -by- k grid is. Just like paths and binary trees, the k -by- k grid has a very regular structure. Thus, one might hope that it is not hard to find the treedepth of the grid. Furthermore, if we again consider the intuitive properties of a good construction that decomposes a graph, they seem to find such a construction for a grid graph because one can quickly find small and balanced separators. This makes it quite surprising, that the exact treedepth value of the k -by- k grid is still unknown. Only linear upper and lower bounds have been proven. In this thesis we aim to improve the upper bound and develop a better understanding of the nature of this problem.

1.1 Related Work

The problem of finding the treedepth of graphs was introduced under varying names such as the rank number using minimal k -rankings [GLP96] [GLP99] and minimum elimination

tree height [Pot88]. The problem is also shown to be NP-complete [Pot88]. There is an interest in determining the treedepth of specific graph classes because for many NP-hard problems there are efficient parameterized algorithms for graphs of small treedepth that often use dynamic programming [NPSW23]. The treedepth of a graph G is related to the parameters *pathwidth* and *treewidth* and the following equation holds:

$$\text{treewidth}(G) \leq \text{pathwidth}(G) \leq \text{treedepth}(G) - 1 \quad [\text{BGHK95}]$$

As stated earlier, it is shown that the graph minors enforcing high treedepth are the k -by- k grid, the complete binary tree of height k and the path on 2^k vertices [KR18]. The minor enforcing high treewidth is as well the k -by- k grid and it has treewidth exactly k [CC14]. The exact treedepth of the grid remains an open question but there has been progress in multiple directions. For some specific grid types the treedepth is determined, for other more broader classes only upper and lower bounds have been found. Considering the m -by- n grid the treedepth has been determined for fixed $m \in \{1, 2, 3, 4\}$ and for general m and n , upper and lower bounds have been published. For $m = 1$, the graph is a path on n vertices and the treedepth or rank number is $\lfloor \log_2 n \rfloor + 1$ [BDJ⁺98]. For $m = 2$, the 2-by- n grid is a ladder and its rank number is determined [NON09]. For $m = 3$, Alpert determined the exact value, proved a general upper bound of the m -by- n grid of $m \cdot (\lfloor \log_2 n \rfloor + 1)$ and provided an upper bound for the k -by- k grid of $3k - \text{bitcount}(k) - 2\lfloor \log_2 k \rfloor$ [Alp10]. This upper bound was improved to roughly $2.519k \pm o(k)$ by Bar-Noy et al. and a lower bound of $\frac{3}{2}k$ is provided as well [BNCL⁺12]. Chen improved the lower bound to $\frac{5}{3}k - \frac{25}{9}$ and determined the exact rank number of the 4-by- n grid [Che13].

1.2 Contributions

As described, the problem of finding a good upper bound can be reduced to finding good separators and recursively cutting a grid graph into smaller and smaller pieces. To find an upper bound of the general k -by- k grid, because k can be any integer, one would intuitively picture it as a square area that is cut into pieces by drawing continuous lines on it. Especially the upper bound

$$2.519k \pm o(k)$$

is argued in that way, though without a formalized explanation as to why this suffices to prove upper bounds up to sublinear terms [BNCL⁺12].

In this work, we aim to close this gap by introducing the continuous treedepth of shapes in the plane. In this new problem, a bounded area in the plane is given with the goal to cut it into ever decreasing pieces. The pieces that remain after a cut are arranged in a rooted tree, similar to how an elimination tree is formed. The height of this tree is calculated considering the length of the cuts and the goal is to keep the height as small as possible. This new problem is then connected to the treedepth of grid graphs. We show that an upper bound of the continuous treedepth of a shape can be translated to an upper bound of a corresponding grid graph. This allows us to improve the treedepth of a few grid graphs including the k -by- k grid. For the continuous *rotated rectangle* $\mathcal{R}_{a,b}$, the *wedge* $\mathcal{W}_a$, *triangle* $\mathcal{T}_a$ and *square* $\mathcal{S}_a$ as well as for their corresponding grid graphs $R_{m,n}$, W_k , T_k and G_k the upper bounds proven in this thesis using the continuous treedepth are stated in Table 1.1.

Additionally, the upper bound for the rotated rectangle graph $R_{n,m}$ is improved to $2m + n - 2 - \text{bc}(m - 1) - \text{bc}(n - 1)$ using an entirely discrete approach. Here $\text{bc}(k)$ is the number of ones in the binary representation of k . This gives an idea about how close the continuous approach gets us to the upper bound of a grid graph.

Shape	Upper Bound	Grid Graph	Upper Bound
$\mathcal{R}_{a,b}$	$2a + b$	$R_{m,n}$	$2m + n + \mathcal{O}(\log n)$
$\mathcal{W}_a$	$\frac{1+\sqrt{57}}{8}a \approx 1.069 a$	W_k	$1.069 k + \mathcal{O}(\log k)$
$\mathcal{T}_a$	$\frac{5+\sqrt{57}}{8}a \approx 1.569 a$	T_k	$1.569 k + \mathcal{O}(\log k)$
$\mathcal{S}_a$	$\frac{87+7\sqrt{57}}{42+2\sqrt{57}}a \approx 2.450 a$	G_k	$2.450 k + \mathcal{O}(\log k)$

Table 1.1: Upper bounds on the treedepth of shapes and grid graphs.

1.3 Outline

We start by fixing some basic definitions and by introducing objects that are relevant throughout the whole thesis in chapter 2.

Next, in chapter 3, we introduce decomposition trees and prove that they too characterize treedepth. We do this by showing that they correspond to elimination forests.

We then introduce the continuous treedepth of shapes in chapter 4. We provide an upper bound for the square shape and with it prove the existence of the continuous treedepth for all shapes.

In chapter 5 the problem of finding the continuous treedepth of a shape is connected to the treedepth of grid graphs. We provide Theorem 5.11 which translates a continuous cutting strategy to an upper bound of the treedepth.

This result is then used in chapter 6 to provide upper bounds on the treedepth of different shapes and grid graphs. In this chapter we also improve the upper bound of the treedepth of the rotated rectangle grid using an entirely discrete approach.

Finally, in chapter 7, we summarize the results and give an outlook on possible progress on the problems in this thesis.

2. Preliminaries

If not mentioned otherwise, all graphs in this thesis are considered to be finite, undirected, and simple, which means there are no loops or double edges between vertices. We first fix some basic definitions and notations.

For a graph G with vertex set $V = V(G)$ and edge set $E = E(G)$, we define:

- For $M \subseteq V$ the induced graph on vertex set M , denoted by $G[M]$ is the graph on vertex set M and an edge between $u, v \in M$ if and only if $uv \in E(G)$.
- For a graph G and any vertex set $M \subseteq V(G)$, the induced graph $G[M]$ is a component of G if there are no edges between M and $V(G) \setminus M$ in G . Note that a component is not necessarily connected.
- For graphs G and H , we define the union $G \cup H$ as the graph on vertex set $V(G) \cup V(H)$ and edge set $E(G) \cup E(H)$.

Definition 2.1 (Graph Isomorphism). *An isomorphism from a graph G to a graph H is a bijective function $f : V(G) \rightarrow V(H)$ such that for all $u, v \in V(G)$ it hold that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$. If an isomorphism from G to H exists, we call them isomorphic.*

Definition 2.2 (Rooted Tree). *A rooted tree is a connected, cycle-free graph T with a designated root-vertex r . For any vertex $v \in V(T)$ let $(r = x_0, x_1, \dots, x_n = v)$ be the unique path from root r to v in T . The depth of v is the number of vertices on this path:*

$$\text{depth}(v) := n + 1$$

The height of the tree T is defined:

$$h(T) := \max_{v \in V(T)} \text{depth}(v)$$

Definition 2.3 (Rooted Forest). *A rooted forest is a disjoint union of rooted trees. The height of a rooted forest is the maximum height of the rooted trees in it. We define that a vertex u is a descendant of a vertex v in a rooted forest F if v is on the unique path from a root to u .*

2.1 Treedepth

There are many ways to define the treedepth. Earlier work on the treedepth of the k -by- k grid has studied it under the names and definitions of *ordered coloring* and *rank number*. In this thesis, we work with the treedepth defined using *elimination forests* because it simplifies the translation to decomposition trees later in chapter 3.

Definition 2.4 (Elimination Forest). *A rooted forest F is an elimination forest of a graph G if and only if $V(F) = V(G)$ and for every edge $uv \in E(G)$ it holds that one vertex of u and v is a descendant of the other in F .*

Definition 2.5 (Treedepth). *The treedepth of a graph G denoted by $\text{td}(G)$ is the minimum height of an elimination forest of G .*

2.2 Grid Graphs

We define general *grid graphs* to be subsets of the k -by- k grid and introduce how we embed a grid graph into the plane.

Definition 2.6 (k -by- k -Grid). *For $k \in \mathbb{N}$, the k -by- k grid, denoted by G_k , is the graph on the vertex set $V(G_k) = \{0, \dots, k-1\}^2$ and an edge between vertices (x, y) and (x', y') if and only if $x = x'$ and $|y' - y| = 1$ or $y = y'$ and $|x' - x| = 1$.*

Definition 2.7 (Grid Graph). *A graph G is a grid graph if it is a subgraph of the k -by- k grid G_k for some $k \in \mathbb{N}$. That implies that its vertex set $V(G)$ is a subset of $\mathbb{N}^2$.*

Definition 2.8 (Induced Grid Graph). *The induced grid graph on a finite set $V \subset \mathbb{Z}^2$ is the grid graph on the vertex set V and an edge between vertices (x, y) and (x', y') if and only if $x = x'$ and $|y' - y| = 1$ or $y = y'$ and $|x' - x| = 1$.*

Definition 2.9 (Straight Line). *A straight line between two points $p = (x_p, y_p)$ and $q = (x_q, y_q)$ denoted by $\overline{pq}$ is the image of the function $\gamma : [0, 1] \rightarrow \mathbb{R}^2, t \mapsto (x_p + t \cdot (x_q - x_p), y_p + t \cdot (y_q - y_p))$.*

Definition 2.10 (Canonical Embedding). *For a grid graph G with vertex set V and edge set E , we define the canonical embedding:*

$$\text{Emb} : V \cup E \rightarrow \mathcal{P}(\mathbb{R}^2) \text{ with } \begin{array}{ll} \text{Emb}(v) := \{v\} & \text{for } v \in V \\ \text{Emb}(uv) := \overline{uv} & \text{for } uv \in E \end{array}$$

We additionally define:

$$\text{Emb}(G) := \text{Emb}(V) \cup \text{Emb}(E) \subseteq \mathbb{R}^2$$

2.3 Continuous Objects

For the later defined continuous treedepth, we define some continuous objects.

First, we define *shapes* in the plane. They correspond to the grid graphs in the discrete setting as the objects that get decomposed.

Definition 2.11 (Shape). *A subset $S \subset \mathbb{R}^2$ is a shape if S is the closure of an open, connected and bounded set $S_0 \subset \mathbb{R}^2$. We denote the closure $S = \overline{S_0}$. The diameter S for some norm $\|\cdot\|$ is defined as:*

$$\text{diam}_{\|\cdot\|}(S) = \sup_{p,q \in S} \|p - q\|$$

Definition 2.12 (Shape Subset). *For a given norm $\|\cdot\|$, we say that a Shape S' is a subset of a shape S up to $\|\cdot\|_\infty$ -isometry if and only if there is a $\|\cdot\|$ -isometry $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ such that $\phi(S') \subseteq S$. We denote this property by $S' \subseteq_{\|\cdot\|} S$*

We define a *curve*, which is a continuous line in the plane. It later serves as the base for defining a cut that divides a shape into smaller shapes.

Definition 2.13 (Curve). *A curve in the plane is the image $I(\gamma)$ of a continuous function $\gamma : [a, b] \rightarrow \mathbb{R}^2$. For simplicity, we also refer to γ as a curve.*

Definition 2.14 (Length of a Curve). *For a given norm $\|\cdot\|$, the length $\text{len}_{\|\cdot\|}(\gamma)$ of a curve $\gamma : [a, b] \rightarrow \mathbb{R}^2$ is defined in the following way:*

For $n \in \mathbb{N}$, we call $a = x_0 < x_1 < \dots < x_n = b$ a partition of $[a, b]$. Let $\mathcal{P}$ be the set of all possible partitions of $[a, b]$. Define:

$$\text{len}_{\|\cdot\|}(\gamma) := \sup_{P \in \mathcal{P}} \sum_{i=1}^n \|\gamma(x_i) - \gamma(x_{i-1})\|$$

We call the curve γ $\|\cdot\|$ -rectifiable if and only if $\text{len}_{\|\cdot\|}(\gamma)$ is finite.

Definition 2.15 (p -Norm). *For $p \in \mathbb{N}$ we define the p -norm $\|\cdot\|_p$ on the plane $\mathbb{R}^2$:*

$$\|(x, y)\|_p := \sqrt[p]{|x|^p + |y|^p} \text{ for all } (x, y) \in \mathbb{R}^2$$

We also define the Chebyshev norm denoted by $\|\cdot\|_\infty$:

$$\|(x, y)\|_\infty := \lim_{p \rightarrow \infty} \|(x, y)\|_p = \max\{|x|, |y|\}$$

3. Discrete Decomposition Trees

In this chapter, we present a more complex characterization of the treedepth. In section 3.1 we define decomposition trees which are more similar to the continuous treedepth we define in chapter 4. In section 3.2 we show that the minimum height of a decomposition tree characterizes the treedepth by linking it to elimination forests.

3.1 Definition

Our definition of the treedepth via elimination forests encapsulates the idea of the elimination game stated in the introduction where only a single vertex is deleted per connected component at a time. It is known in the literature that the treedepth can also be characterized by deletion of small separators:

Theorem 3.1. [DKKM93] *Let G be a graph which is not a clique. Then*

$$\text{td}(G) = \min_S \max_C (|S| + \text{td}(C))$$

Where S is a minimal vertex separator in G and C is a connected component of $G[V(G) \setminus S]$.

This motivates the definition of decomposition trees that allow to wrap multiple vertices into a *deletion set* and represent a strategy to decompose a graph.

Definition 3.2 (Decomposition Tree of a Graph). *For a graph G a decomposition tree D is a non-empty rooted tree where the nodes are graphs, G being the root and for every non-leaf H with children $H_1, \dots, H_m$ there is a deletion set $M \subseteq V(H)$ denoted by $\text{dset}(H)$ such that $H - M$ consists of the pairwise disjoint (not necessarily connected) components $H_1, \dots, H_m$, meaning $G - M = \bigcup_{i=1}^m H_i$.*

For any graph $H \in V(D)$ let $(G = X_0, X_1, \dots, X_n = H)$ be the unique path from root G to H in D . Define the depth of H :

$$\text{depth}(H) := \sum_{i=0}^{n-1} |\text{dset}(X_i)|$$

The height of the tree D is defined:

$$h(D) := \max_{H \in V(D)} \text{depth}(H)$$

Now, a decomposition tree encapsulates a strategy to divide a graph into smaller pieces by assigning deletion sets. Note that the deletion sets can also be empty. Also the graphs can be empty or disconnected. For a decomposition tree to give a strategy on how to delete the graph completely, we need to define the following property:

Definition 3.3 (Complete Decomposition Tree). *A decomposition tree of a graph G is called complete if and only if all leaf graphs have zero vertices.*

3.2 Correspondence of Elimination Forests and Decomposition Trees

We now show how complete decomposition trees also characterize treedepth by showing how they correspond to elimination forests. We first show how any elimination forest can be transformed into a decomposition tree of at most the same height in Lemma 3.4. Then we prove that any decomposition tree can be transformed into an elimination forest of at most the same height in Lemma 3.5. In Corollary 3.6, we bring both together to show that the minimum height of a decomposition tree of a graph G characterizes the treedepth of G .

Lemma 3.4. *Let G be a graph and F an elimination forest of G . Then there exists a complete decomposition tree D of G such that $h(D) \leq h(F)$.*

Proof. We prove this by induction on the number of vertices n of G .

Base case ($n \in \{0, 1\}$): If G has no vertex, then F has no vertex and we can construct a decomposition tree with one node that is the graph on zero vertices. It has height zero and is complete. If G has one vertex v , then F is also just the one vertex graph and therefore has height 1. We construct the decomposition tree D with root G , $\text{dset}(G) = \{v\}$ and the graph on zero vertices as the child of G . This makes D a complete decomposition tree of G of height 1.

Inductive step ($n \geq 2$): Assume that for all graphs H on fewer than n vertices and any elimination forest F_H of H , we can construct a complete decomposition tree of height at most $h(F_H)$.

Let G be a graph on n vertices and F an elimination forest of G . If F contains multiple trees, call induction on each one to get a set P of decomposition trees. Join all decomposition trees together by inserting the root G with an empty deletion set as the parent of the roots of all decomposition trees in P . The resulting complete decomposition tree of G has height at most $h(F)$.

If F only contains one elimination tree T , remove the root r to be left with elimination trees $T_1, \dots, T_m$. Now, every T_i is an elimination tree of the graph $G[V(T_i)]$ and has height of at most $h(F) - 1$. The graph $G[V(T_i)]$ has at most $n - 1$ vertices. By induction, for every $i \in \{1, \dots, m\}$ there exists a complete decomposition tree D_i of height at most $h(F) - 1$. Because T is an elimination tree, there are no edges in G between $G[V(T_i)]$ and $G[V(T_j)]$ for $i \neq j$. This allows us to choose $G[T]$ as the root of a new decomposition tree D with deletion set $\{r\}$ and attach $D_1, \dots, D_m$ below it. Now D is complete and has height at most $h(F)$. This finishes the induction and the proof. $\square$

Lemma 3.5. *Let G be a graph and D a complete decomposition tree of G . Then there exists an elimination forest F of G such that $h(F) \leq h(D)$.*

Proof. We prove this by induction on the number of vertices n of G .

Base case ($n \in \{0, 1\}$): If G has no vertex then we can take an empty tree. If G has one vertex v then D has height at least 1 and we can take $T = G$ as the one vertex tree.

Inductive step ($n \geq 2$): Assume that for all graphs H on fewer than n vertices and any complete decomposition tree D_H of H we can construct an elimination forest of height at most $h(D_H)$.

Let G be a graph on n vertices and D a complete decomposition tree of G . If the root of D has an empty deletion set, we delete it to get one or multiple decomposition trees. We recursively repeat this process with any resulting tree that has a root with an empty deletion set, until we are left with a set P of decomposition trees with roots that have non-empty deletion sets.

If there are multiple trees in P , then each is a decomposition tree of a component of G , so there are no edges in between the root graphs of two different decomposition trees in P . In this case we can apply induction to every tree in P to get elimination trees that, combined, form an elimination forest of G of height at most $h(D)$.

If there is only one decomposition tree $\tilde{D}$ in P , then G is the root of $\tilde{D}$. We define $k := |\text{dset}(G)|$. We know $k \geq 1$ because $\text{dset}(G)$ is not empty. Remove G to get a set of decomposition trees $D_1, \dots, D_m$ with respective root graphs $G_1, \dots, G_m$. The decomposition trees all have height at most $h(D) - k$. Every G_i has at most $n - k$ vertices and because $k \geq 1$, by induction there exist elimination forests $F_1, \dots, F_m$ of $G_1, \dots, G_m$ respectively that all have height at most $h(D) - k$. Join these elimination forest to one elimination forest F of height at most $h(D) - k$. We now order the vertices of $\text{dset}(G) = \{x_1, \dots, x_k\}$, make them form a path $x_1, \dots, x_k$ and insert x_k as the parent of all roots of the trees in F . This forms a new tree T with x_1 being the root. It has height at most $h(D)$ because the added path increases the height by exactly k . Also, $V(T) = V(G)$ and in G there are no edges between two G_i, G_j with $i \neq j$ because in the decomposition tree $\tilde{D}$ they are children of G . That means, there are no edges between two trees in F and we can conclude that T is an elimination tree of G . This finishes the induction. $\square$

Corollary 3.6. *The treedepth of a graph G is equal to the minimum height of a complete decomposition tree of G .*

Proof. This result follows immediately from Lemma 3.4 and 3.5. $\square$

4. Continuous Treedepth

In this chapter the continuous treedepth of shapes in the plane is introduced. We first state the necessary definitions in section 4.1. Then we give an example and prove the existence of the continuous treedepth in section 4.2.

4.1 Definition

The idea is similar to treedepth on graphs: One can cut the shape into connected components and then recursively continues to cut the leftover shapes into smaller pieces. Similar to the treedepth of graphs, this creates a decomposition tree of the shape. Because the cuts don't consist of vertices like the deletion sets, the height of the decomposition tree is calculated by using the length of the cuts. We define a cut and the decomposition tree of shapes:

Definition 4.1 (Cut). *A cut γ in a shape S is a finite union of (the image of) $\|\cdot\|$ -rectifiable curves (γ_i) :*

$$\gamma = \bigcup_{i=1}^n \mathbf{I}(\gamma_i) \subseteq S$$

such that $S \setminus \gamma$ consists of $m \geq 2$ two connected components.

The length of cut γ for norm $\|\cdot\|$ is defined as $\text{len}_{\|\cdot\|}(\gamma) := \sum_{i=1}^n \text{len}_{\|\cdot\|}(\gamma_i)$.

Define the curve count of γ : $\text{cc}(\gamma) := n$. This is the number of curves in γ .

For $i \in \{1, \dots, m\}$, name the remaining connected components C_γ^i . The remaining shapes are $\overline{C_\gamma^i}$. We denote the remaining shapes $S_\gamma^i := \overline{C_\gamma^i}$.

For a shape S and a cut γ in S , observe that $S_\gamma^1, \dots, S_\gamma^m$ are proper shapes because $S = \overline{S_0}$ for some open, connected and bounded set S_0 , and therefore $S_\gamma^i = \overline{C_\gamma^i} = \overline{C_\gamma^i \cap S_0}$ with $C_\gamma^i \cap S_0$ being open, bounded and connected. It also holds that $S = \bigcup_{i=1}^m S_\gamma^i$, so the points of the cut are added back into the remaining shapes by taking the closure.

Definition 4.2 (Decomposition Tree of a Shape). *A decomposition tree D of a shape S is a non-empty rooted tree where the nodes are shapes, S being the root and for every non-leaf $Y \in V(D)$ with children $Y_1, \dots, Y_m$ there is a cut γ in Y , denoted by $\text{cut}(Y)$, that leaves m shapes, such that $Y_\gamma^i = Y_i$ for all $i \in \{1, \dots, m\}$.*

Further define the following: For any shape $X \in V(D)$ let $(S = X_0, X_1, \dots, X_n = X)$ be the unique path from root S to X in D . Let $\|\cdot\|$ be a norm. Define the depth and the curve count of X :

$$\text{depth}_{\|\cdot\|}(X) := \sum_{i=0}^{n-1} \text{len}_{\|\cdot\|}(\text{cut}(X_i))$$

$$\text{cc}(X) := \sum_{i=0}^{n-1} \text{cc}(\text{cut}(X_i))$$

Now the height and the curve count of the tree D is defined:

$$h_{\|\cdot\|}(D) := \max_{X \in V(D)} \text{depth}_{\|\cdot\|}(X)$$

$$\text{cc}(D) = \max_{X \in V(D)} \text{cc}(X)$$

We write $h(D)$ instead of $h_{\|\cdot\|}(D)$ if the considered norm is clear from the context.

Definition 4.3 (ε -Decomposition Tree). *A decomposition tree D is defined to be an ε -decomposition tree for some $\varepsilon > 0$ and some norm $\|\cdot\|$ if all leaves in D have a $\|\cdot\|$ -diameter less than ε .*

To later construct decomposition trees easier, we introduce a notation for them. Let S be a shape and D a decomposition tree of S . For every shape $X \in V(D)$ let $D(X)$ be the sub decomposition tree of D that starts with X as the root. If X is a leaf, we simply denote $D(X)$ with X . If X is a non-leaf, let m be the number of children, then we recursively denote $D(X)$:

$$X \xrightarrow[\text{cc}(\text{cut}(X))]{\text{len}_{\|\cdot\|}(\text{cut}(X))} \begin{cases} D(X_{\text{cut}(X)}^1) \\ D(X_{\text{cut}(X)}^2) \\ \dots \\ D(X_{\text{cut}(X)}^m) \end{cases}$$

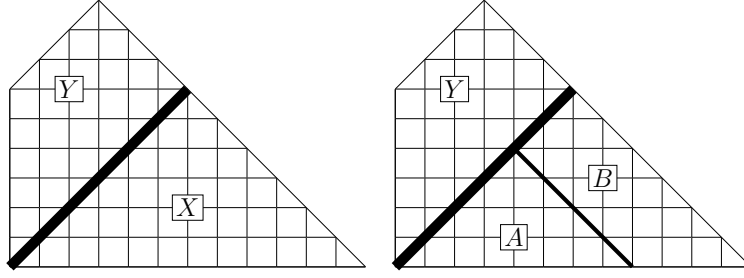
To get a better understanding of the notation, we give the following example. Figure 4.1 shows how a shape S is cut into the shapes X and Y , and how X is then cut into A and B . Both cuts consist of one curve. Let's say the first one has length 6 and the second one length 4. Then the resulting decomposition tree is denoted the following way:

$$S \xrightarrow[1]{6} \begin{cases} X \\ Y \end{cases} \xrightarrow[1]{4} \begin{cases} A \\ B \end{cases}$$

In contrast to treedepth on graphs, it is not possible to entirely delete a shape when we cut it into pieces. For that reason, ε -treedepth is introduced. This parameter only requires all pieces to have a diameter less than ε .

Definition 4.4 (Continuous ε -Treedepth). *Let S be a shape, $\|\cdot\|$ a norm, and $\varepsilon > 0$. Define $\mathcal{D}(S, \varepsilon)$ as the set of all ε -decomposition trees of S for norm $\|\cdot\|$. We define the ε -treedepth of S :*

$$\text{ctd}_{\varepsilon, \|\cdot\|}(S) := \inf_{D \in \mathcal{D}(S, \varepsilon)} h_{\|\cdot\|}(D)$$

Figure 4.1: Decomposition of shape S after one and two cuts.

We observe that for any shape S , norm $\|\cdot\|$ and $\varepsilon > \varepsilon' > 0$ the following holds:

$$\text{ctd}_{\varepsilon', \|\cdot\|}(S) \geq \text{ctd}_{\varepsilon, \|\cdot\|}(S)$$

This is because for any $D \in \mathcal{D}(S, \varepsilon')$ the leaves of D have diameter less than ε' , so less than ε as well. That means $D \in \mathcal{D}(S, \varepsilon)$ holds and therefore $\mathcal{D}(S, \varepsilon') \subseteq \mathcal{D}(S, \varepsilon)$.

Now we define the continuous treedepth of shapes.

Definition 4.5 (Continuous Treedepth). *Based on the ε -treedepth, we define the treedepth of a shape S for a norm $\|\cdot\|$:*

$$\text{ctd}_{\|\cdot\|}(S) := \sup_{\varepsilon > 0} \text{ctd}_{\varepsilon, \|\cdot\|}(S)$$

We write $\text{ctd}_{\varepsilon}(S)$ instead of $\text{ctd}_{\varepsilon, \|\cdot\|}(S)$ and $\text{ctd}(S)$ instead of $\text{ctd}_{\|\cdot\|}(S)$ if the considered norm is clear from the context.

Now that we have defined the continuous treedepth, we argue that a few properties hold that simplify working with the continuous treedepth. The reasoning could be properly formalized, but because this is not the core of the thesis, we argue less formally and more intuitively.

For a cut γ in a shape S and an isometric subshape $S' \subseteq_{\|\cdot\|} S$, we obtain a corresponding cut γ' for S' by applying the isometric function to γ and restricting the result to S' . Here, we won't formally argue the details and just note that the length of γ' is at most the length of γ and the remaining shapes are subsets up to $\|\cdot\|$ -isomorphism of the original remaining shapes. That means we can also transform any decomposition tree D of S into a decomposition tree D' of S' . Its height is at most the height of D and the shapes will be subsets of the original shapes up to $\|\cdot\|$ -isometry. So if D was an ε -decomposition tree of S , then D' is an ε -decomposition tree of S' of at most the same height. We can conclude that $\text{ctd}_{\varepsilon}(S') \leq \text{ctd}_{\varepsilon}(S)$.

We also simplify our notation for decomposition trees. If a cut leaves two shapes X, Y with $Y \subseteq_{\|\cdot\|} X$, then it suffices to state a decomposition tree for X and we assume a corresponding tree to be applied to Y . As an example, consider the following tree again:

$$S \xrightarrow[\frac{1}{1}]{\frac{6}{1}} \left\{ \begin{array}{l} X \\ Y \end{array} \right. \xrightarrow[\frac{1}{1}]{\frac{4}{1}} \left\{ \begin{array}{l} A \\ B \end{array} \right.$$

If now $Y \subseteq_{\|\cdot\|} X$ holds, then we can place a cut in Y similar to the one in X . Figure 4.2 shows how this cut is placed, considering Y is a subset of S if mirrored on the first cut. Without simplification, the tree is denoted in the following way:

$$S \xrightarrow[1]{6} \left\{ \begin{array}{l} X \xrightarrow[1]{4} \begin{Bmatrix} A \\ B \end{Bmatrix} \\ Y \xrightarrow[1]{3} \begin{Bmatrix} A' \\ B' \end{Bmatrix} \end{array} \right.$$

The simpler notation of this tree is:

$$S \xrightarrow[1]{6} X \xrightarrow[1]{4} \begin{Bmatrix} A \\ B \end{Bmatrix}$$

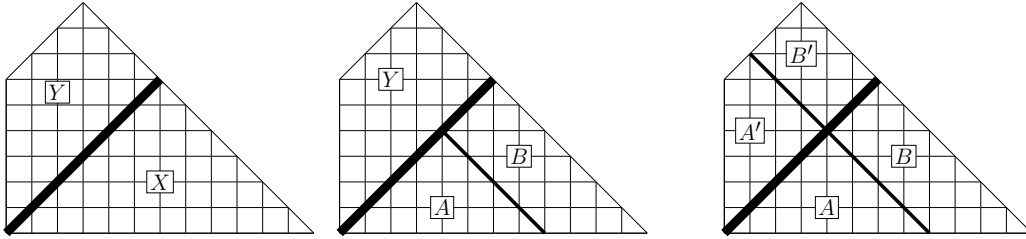


Figure 4.2: Decomposition of shape S after one, two and three cuts.

4.2 Example and Existence

In this section, considering the Euclidean norm, we show an upper bound on the treedepth of the simple a -by- a -square, that is for $a \in \mathbb{R}$ with $a > 0$ the shape $[0, a] \times [0, a]$, denoted with $\mathcal{S}_a$. We do this by constructing a decomposition tree for every $\varepsilon > 0$ and calculating its height. Using this example we conclude that the treedepth exists for every shape.

Lemma 4.6. *For the square $\mathcal{S}_a$ the Euclidean norm $\|\cdot\|_2$ and any $\varepsilon > 0$ it holds that $\text{ctd}_\varepsilon(\mathcal{S}_a) < 3a$.*

Proof. We construct a decomposition tree of $\mathcal{S}_a$. For an arbitrary $\varepsilon > 0$. In the middle of $\mathcal{S}_a$, place a vertical cut of length a to get two new shapes: $X_a := [0, a/2] \times [0, a]$ and $[a/2, a] \times [0, a]$. The two shapes are $\|\cdot\|_2$ -isometric and therefore we get the following decomposition tree:

$$\mathcal{S}_a \xrightarrow[1]{a} X_a$$

Now cutting horizontally in half leaves $\mathcal{S}_{a/2}$ and $[0, a/2] \times [a/2, a]$, which are $\|\cdot\|_2$ -isometric again and we get the following decomposition tree:

$$\mathcal{S}_a \xrightarrow[1]{a} X_a \xrightarrow[1]{\frac{1}{2}a} \mathcal{S}_{\frac{1}{2}a}$$

Define $a_t := (1/2)^t a$ for $t \in \mathbb{N}_0$ and we can generalize the decomposition tree to:

$$\mathcal{S}_{a_t} \xrightarrow[1]{a_t} X_{a_t} \xrightarrow[1]{a_{t+1}} \mathcal{S}_{a_{t+1}}$$

We now recursively define the following decomposition tree $D(\mathcal{S}_{a_t})$:

If $\text{diam}_{\|\cdot\|_2}(\mathcal{S}_{a_t}) < \varepsilon$, we define $D(\mathcal{S}_{a_t}) := \mathcal{S}_{a_t}$, a single leaf. Otherwise define:

$$D(\mathcal{S}_{a_t}) := \mathcal{S}_{a_t} \xrightarrow[1]{a_t} X_{a_t} \xrightarrow[1]{a_{t+1}} D(\mathcal{S}_{a_{t+1}})$$

This is now an ε -decomposition tree, because all leaves have diameter less than ε . To calculate its height, we need $t \in \mathbb{N}_0$ such that $\text{diam}_{\|\cdot\|_2}(\mathcal{S}_{a_t}) < \varepsilon$. The $\|\cdot\|_2$ -diameter of the square $\mathcal{S}_a$ is $\text{diam}_{\|\cdot\|_2}(\mathcal{S}_a) = a\sqrt{2}$. We therefore choose $\tilde{t} \in \mathbb{N}_0$ such that $a_{\tilde{t}} = (1/2)^{\tilde{t}} < \varepsilon/\sqrt{2}$. We prove by backward induction that $h(D(\mathcal{S}_{a_t})) < 3a_t$ for all $t \in \{0, \dots, \tilde{t}\}$:

Base case ($t = \tilde{t}$): With our choice of $\tilde{t}$ we can show:

$$\text{diam}_{\|\cdot\|_2}(\mathcal{S}_{a_{\tilde{t}}}) = a_{\tilde{t}}\sqrt{2} < \frac{\varepsilon}{\sqrt{2}}\sqrt{2} = \varepsilon$$

It follows that $D(\mathcal{S}_{a_{\tilde{t}}}) = \mathcal{S}_{a_{\tilde{t}}}$ and with that $h(D(\mathcal{S}_{a_{\tilde{t}}})) = 0 < 3a_{\tilde{t}}$.

Inductive step ($t \in \{0, \dots, \tilde{t} - 1\}$): Assume $h(D(\mathcal{S}_{a_{t+1}})) < 3a_{t+1}$ holds.

If $\mathcal{S}_{a_t}$ has diameter less than ε , we again have height 0 which is less than $3a_t$. Otherwise it holds that $D(\mathcal{S}_{a_t}) = \mathcal{S}_{a_t} \xrightarrow[1]{a_t} X_{a_t} \xrightarrow[1]{a_{t+1}} D(\mathcal{S}_{a_{t+1}})$.

We calculate:

$$\begin{aligned} h(D(\mathcal{S}_{a_t})) &= h\left(\mathcal{S}_{a_t} \xrightarrow[1]{a_t} X_{a_t} \xrightarrow[1]{a_{t+1}} D(\mathcal{S}_{a_{t+1}})\right) \\ &= a_t + a_{t+1} + h(D(\mathcal{S}_{a_{t+1}})) \\ &< a_t + a_{t+1} + 3a_{t+1} && \text{(induction hypothesis)} \\ &= a_t + \frac{1}{2}a_t + \frac{3}{2}a_t \\ &= 3a_t \end{aligned}$$

This finishes the induction.

With $a = a_0$ we can conclude $\text{ctd}_\varepsilon(\mathcal{S}_a) \leq h(D(\mathcal{S}_a)) < 3a$. □

Corollary 4.7. *For the Euclidean norm it holds that $\text{ctd}(\mathcal{S}_a) \leq 3a$.*

Proof. This follows directly from Lemma 4.6, because for all $\varepsilon > 0$ it holds that $\text{ctd}_\varepsilon(\mathcal{S}_a) < 3a$. We can conclude $\text{ctd}(\mathcal{S}_a) \leq 3a$. □

Lemma 4.8. *For every shape S and the Euclidean norm, the continuous treedepth exists.*

Proof. Let S be an arbitrary shape. Because S is bounded, we can find $a \in \mathbb{R}$ such that $S \subseteq_{\|\cdot\|_2} \mathcal{S}_a$. Now for an arbitrary $\varepsilon > 0$, by Lemma 4.6 we know $\text{ctd}_\varepsilon(\mathcal{S}_a) < 3a$ and it follows $\text{ctd}_\varepsilon(S) \leq \text{ctd}_\varepsilon(\mathcal{S}_a) < 3a$. We can conclude that $\text{ctd}(S) < 3a$ and therefore that the continuous treedepth exists for S . □

We now know that the treedepth exists for every shape when considering the Euclidean norm. It is easy to see, that for all other p -norms, the same upper bound for the square holds because horizontal and vertical distances, and therefore the lengths of the cuts in this example, are the same in all p -norms. This also implies that for all p -norms, the treedepth exists for every shape. For any norm that is not a p -norm, the proof would have to be adjusted a bit and the upper bound on the square could be different, but the existence would follow as well. This is because we exponentially shrink dimensions of the shape and

therefore sum up exponentially shrinking cut lengths. This geometric sum always results in a constant upper bound for a fixed shape no matter how small ε is chosen.

Later on, when treedepth on shapes is connected to treedepth on grid graphs, we use the Chebyshev norm $\|\cdot\|_\infty$. This is because it aligns the best with deleting vertices from a grid, as we see in the next chapter 5.

5. Connecting Continuous and Discrete Treedepth

In this chapter we connect the continuous treedepth to the discrete one. In the continuous setting we apply cuts to shapes to decompose them. The goal is that we can apply these cuts also to grid graphs that fit into these shapes. We first show how any curve can be translated into a set of vertices that “follows” the curve in section 5.1. Then we argue that any cut can be translated to a set of vertices, such that its deletion divides the embedding of a grid graph in section 5.2. Finally, in section 5.3, we are going to prove how a continuous decomposition tree is translated to a discrete one and conclude the main Theorem 5.11 which states that any 1-decomposition tree of a shape implies an upper bound on the treedepth of grid graphs that fit into the shape.

The norm we use to build the connection is the Chebyshev norm $\|\cdot\|_\infty$. This is because it brings the length of a curve close to the size of the separator in a grid graph that divides the graph correspondingly. In Figure 5.1 we see an example of two lines that cross the same number of vertices. As desired, in the Chebyshev norm, the length of the lines is the same.

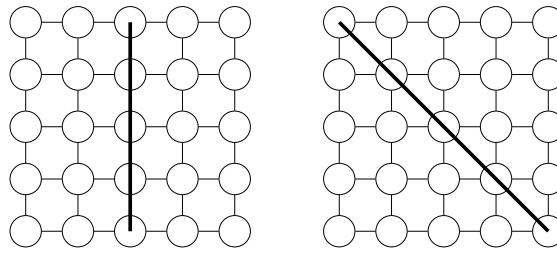


Figure 5.1: Lines touching exactly 5 vertices of the grid of different Euclidean length.

5.1 The Lattice-Vertex Deletion Set of a Curve

We first consider an arbitrary curve in the plane and calculate how many vertices of the infinite square lattice need to be deleted to get rid of all intersections. For that the following definitions are needed:

Definition 5.1 (Square Lattice). *The square lattice L is the infinite graph with vertex set $\mathbb{Z}^2$ and an edge between vertices (x, y) and (x', y') if and only if $x = x'$ and $|y' - y| = 1$ or $y = y'$ and $|x' - x| = 1$.*

We define the canonical embedding of the square lattice and its subsets such that it corresponds to the definition of the canonical embedding of grid graphs.

Definition 5.2 (Canonical Embedding). *Let L' be a (finite or infinite) subgraph of the square lattice L with vertex set V and edge set E , we define the canonical embedding:*

$$\text{Emb} : V \cup E \rightarrow \mathcal{P}(\mathbb{R}^2) \text{ with } \begin{array}{ll} \text{Emb}(v) := \{v\} & \text{for } v \in V \\ \text{Emb}(uv) := \overline{uv} & \text{for } uv \in E \end{array}$$

We additionally define:

$$\text{Emb}(L') := \text{Emb}(V) \cup \text{Emb}(E) \subseteq \mathbb{R}^2$$

Definition 5.3 (Straight Lines Path). *A straight lines path P is a special curve that consists of a finite number of straight lines: $P := \bigcup_{i=1}^{m-1} \overline{p_i p_{i+1}}$. For $m = 1$ the straight lines path is just a single point and has length 0.*

Definition 5.4 (Discrete Option). *Let L be the square lattice and γ a $\|\cdot\|_\infty$ -rectifiable curve. A set $M \subseteq V(L)$ is called a discrete option for γ if and only if $|M| \leq \text{len}_{\|\cdot\|_\infty}(\gamma) + 1$ and $\text{Emb}(L - M) \cap \gamma = \emptyset$. So γ does not touch any vertex or edge of $\text{Emb}(L - M)$.*

The main Proposition that we prove is the following:

Proposition 5.5. *Let L be the square lattice and γ a $\|\cdot\|_\infty$ -rectifiable curve. Then there is a set of vertices $M \subseteq V(L)$ that is a discrete option for γ .*

We prove this Proposition at a later point because it builds on Lemma 5.6 and Lemma 5.7 that are proven first. In the first step, we transform a curve into a straight lines path, such that every vertex and edge of the square lattice that the curve touches, are touched by the straight lines path as well.

Lemma 5.6. *Let L be the square lattice and γ a $\|\cdot\|_\infty$ -rectifiable curve. If γ touches $\text{Emb}(L)$ at least once, then there exists a straight lines path $P = \bigcup_{i=1}^{m-1} \overline{p_i p_{i+1}}$ such that the following properties hold:*

- Every vertex and edge of $\text{Emb}(L)$ that γ touches is also touched by P .
- $\text{len}_{\|\cdot\|_\infty}(P) \leq \text{len}_{\|\cdot\|_\infty}(\gamma)$
- Every p_i lies on a vertex or an edge of $\text{Emb}(L)$.

Proof. Let $\gamma : [a, b] \rightarrow \mathbb{R}^2$. Define $E_\gamma := \{uv \in E(L) \mid \exists x \in [a, b] : \gamma(x) \in \overline{uv}\}$ and $V_\gamma := \{v \in V(L) \mid \exists x \in [a, b] : \gamma(x) = v\}$ the sets of edges and vertices, that γ touches.

Now there exists a function $f : E_\gamma \cup V_\gamma \rightarrow [a, b]$ such that for every $uv \in E_\gamma$ it holds that $\gamma(f(uv)) \in \overline{uv}$ and for every $v \in V_\gamma$ it holds that $\gamma(f(v)) = v$.

The curve γ is $\|\cdot\|_\infty$ -rectifiable, so $\|\gamma(x) - \gamma(a)\| \leq \text{len}_{\|\cdot\|_\infty}(\gamma)$ holds for every $x \in [a, b]$. That means that $\gamma([a, b])$ is bounded and therefore only finitely many vertices and edges of $\text{Emb}(L)$ can be reached by γ . So E_γ and V_γ are both finite. This means $U := f(E_\gamma \cup V_\gamma) \subseteq [a, b]$ is finite as well and therefore of the form $U = \{x_1, \dots, x_m\}$ such that $x_i < x_{i+1}$ for all $i \in \{1, \dots, m-1\}$.

Define $p_i := \gamma(x_i)$. Then the straight lines path is the following:

$$P := \bigcup_{i=1}^{m-1} \overline{p_i p_{i+1}}$$

By construction it touches all edges and vertices that γ touches and all p_i are on vertices or edges of $\text{Emb}(L)$. We define the curve $\gamma' : [x_1, x_m] \rightarrow \mathbb{R}^2, x \mapsto \gamma(x)$. Because $[x_0, x_m] \subseteq [a, b]$, the length of γ' is at most the length of γ . The length of γ' is defined as the supremum over the length of all possible straight line partitions. Because $x_0 < x_1 < \dots < x_m$ is a partition of $[x_0, x_m]$, we can now upper bound the length of P :

$$\begin{aligned} \text{len}_{\|\cdot\|}(P) &= \sum_{i=1}^{m-1} \text{len}_{\|\cdot\|}(\overline{p_i p_{i+1}}) \\ &= \sum_{i=1}^{m-1} \|p_{i+1} - p_i\| = \sum_{i=1}^{m-1} \|\gamma(x_{i+1}) - \gamma(x_i)\| \\ &\leq \sup_{\{\tilde{x}_1, \dots, \tilde{x}_{\tilde{n}}\} \in \mathcal{P}} \sum_{i=1}^{\tilde{n}} \|\gamma(\tilde{x}_{i+1}) - \gamma(\tilde{x}_i)\| \\ &= \text{len}_{\|\cdot\|}(\gamma') \\ &\leq \text{len}_{\|\cdot\|}(\gamma) \end{aligned}$$

With $\mathcal{P}$ being the set of all partitions of $[x_0, x_m]$. □

Now we show that we can subdivide any straight line such that every line segment ends up being in a unit square with integer borders.

Lemma 5.7. *Let $\overline{pq}$ be a straight line, then it can be partitioned into a straight lines path $P := \bigcup_{i=0}^n \overline{p_i p_{i+1}}$ such that for every $i \in \{0, \dots, n\}$ there exists $k, l \in \mathbb{Z}$ with $\overline{p_i, p_{i+1}} \subseteq [k, k+1] \times [l, l+1]$ and $\text{len}_{\|\cdot\|}(P) = \text{len}_{\|\cdot\|}(\overline{pq})$.*

Proof. Define $p = (x_p, y_p)$, $q = (x_q, y_q)$, $x : [0, 1] \rightarrow \mathbb{R}, t \mapsto x_p + t \cdot (x_q - x_p)$ and $y : [0, 1] \rightarrow \mathbb{R}, t \mapsto y_p + t \cdot (y_q - y_p)$.

Now we insert points onto $\overline{pq}$ where it crosses integer values so each segment is within a unit square with integer borders.

Define $T := \{t \in [0, 1] \mid x(t) \in \mathbb{Z} \setminus \{x_p, x_q\} \text{ or } y(t) \in \mathbb{Z} \setminus \{y_p, y_q\}\}$. As the set of all $t \in [0, 1]$ where $\overline{pq}$ crosses an integer value in x- or y-direction. Now, between x_p and x_q as well as between y_p and y_q there are only finitely many integers. Assume for the sake of contradiction that T was not finite. Then at least one coordinate, w.l.o.g. x , would have to hit some integer k multiple times. But that would already imply that the line is horizontal, so $k = x_p$. This value was excluded though, leading to a contradiction.

That means T is finite and can be ordered to $T = \{t_1, \dots, t_n\}$ such that $t_i < t_{i+1}$ for all $i \in \{1, \dots, n-1\}$. Now we can get the points $p_i := (x(t_i), y(t_i))$ for $i \in \{1, \dots, n-1\}$ and we define $p_0 := p$ and $p_{n+1} := q$. The straight lines path is:

$$P := \bigcup_{i=0}^n \overline{p_i p_{i+1}}$$

For any $i \in \{0, \dots, n\}$, consider (t_i, t_{i+1}) . By construction, we know that either for all $s \in (t_i, t_{i+1})$ the values $x(s)$ and $y(s)$ are not integers or there is an $s \in (t_i, t_{i+1})$ with $x(s) \in \{x_p, x_q\}$ or $y(s) \in \{y_p, y_q\}$. In the first case it is clear that $\overline{p_i p_{i+1}} = \{(x(s), y(s)) \mid$

$s \in [t_i, t_{i+1}]$ is inside a unit square. If there is an $s \in (t_i, t_{i+1})$ such $x(s) \in \{x_p, x_q\}$ then $\overline{pq}$ is horizontal. If there is an $s \in (t_i, t_{i+1})$ such that $y(s) \in \{y_p, y_q\}$ then $\overline{pq}$ is vertical. Therefore only one of the two can hold. If $\overline{pq}$ was horizontal. Then it must hold that $\overline{p_i p_{i+1}} = \{x(t_i)\} \times [y(t_i), y(t_{i+1})]$. We know that for all $s \in (t_i, t_{i+1})$ the value $y(s)$ is not an integer, so $\overline{p_i p_{i+1}}$ is in a unit square with integer borders. We can argue the same way in the vertical case. We can conclude that for all $i \in \{0, \dots, n\}$ the straight line $\overline{p_i p_{i+1}}$ is within a unit square with integer borders.

Also, because the p_i only subdivided the line $\overline{pq}$ the sum of their length is equal to the length of $\overline{pq}$. $\square$

Now we can prove the final Proposition of this section that was stated in the beginning. For any curve we construct a vertex set of the square lattice such that deleting it leaves a subset of the square lattice that does not touch the curve in any point.

To avoid repetition in the proof of Proposition 5.5, we first prove the following Auxiliary Lemma.

Lemma 5.8. Auxiliary Lemma

Let $P := \bigcup_{i=1}^{m-2} \overline{p_i p_{i+1}}$ be a straight lines path, L the square lattice, p_m another point and $M \subseteq V(L)$ a discrete option for P . Define $P' := P \cup \overline{p_{m-1} p_m}$. If there is a set $M' \subseteq V(L)$ such that $\text{Emb}(L - M') \cap P' = \emptyset$, $|M'| \leq |M| + 1$ and $\text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1} p_m}) \geq 1$ then M' is already a discrete option for P' .

Proof. We only need to show that the size of M' is at most $\text{len}_{\|\cdot\|_\infty}(P') + 1$:

$$\begin{aligned} |M'| &\leq |M| + 1 \leq \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + 1 + 1 \\ &\leq \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1} p_m}) + 1 \\ &= \text{len}_{\|\cdot\|_\infty}(P_m) + 1 \end{aligned}$$

So M' is already a discrete option for P_m . $\square$

We now prove the main Proposition of this section:

Proposition 5.5. *Let L be the square lattice and γ a $\|\cdot\|_\infty$ -rectifiable curve. Then there is a set of vertices $M \subseteq V(L)$ that is a discrete option for γ .*

Proof. If γ does not touch $\text{Emb}(L)$ at all, we take the empty set as a discrete option for γ . So, assume γ touches $\text{Emb}(L)$ at least once. By Lemma 5.6 there exists a straight lines path $P' \bigcup_{i=1}^{b-1} \overline{r_i r_{i+1}}$ for the curve γ that touches the same vertices and edges of $\text{Emb}(L)$ and its length is at most the length of γ . For every $i \in \{0, \dots, b-1\}$ we consider the straight line $\overline{r_i r_{i+1}}$. By Lemma 5.7 there exists a subdivision of $\overline{r_i r_{i+1}}$ into a straight lines path such that every straight line segment is within a unit square with integer borders. We concatenate all these subdivisions back together to get one straight lines path $P = \bigcup_{i=1}^{n-1} \overline{p_i p_{i+1}}$. Now P has the following properties:

- Every vertex and edge of $\text{Emb}(L)$ that γ touches, is also touched by P (Lemma 5.6).
- Every straight line $\overline{p_i p_{i+1}}$ is within a unit square with integer borders, meaning there are $k_i, l_i \in \mathbb{Z}$ such that $\overline{p_i p_{i+1}} \subseteq [k_i, k_i + 1] \times [l_i, l_i + 1]$ (Lemma 5.7).
- All p_i are on the boundary of their unit square $[k_i, k_i + 1] \times [l_i, l_i + 1]$ because in both Lemmas used, the points are only placed with at least one coordinate in $\mathbb{Z}$. That means that each point p_i is on a vertex or the interior of an edge of $\text{Emb}(L)$.

- $\text{len}_{\|\cdot\|_\infty}(P) \leq \text{len}_{\|\cdot\|_\infty}(\gamma)$

For all $m \in \{2, \dots, n\}$ define $P_m := \bigcup_{i=1}^{m-1} \overline{p_i p_{i+1}}$ and $P_1 := p_1$, so P_m is the path up to point p_m . We will prove that for every P_m there exists a discrete option. To do this we will use induction on m to show a slightly stronger statement:

For every $m \in \{1, \dots, n\}$ there exists a discrete option M for P_m and one of three properties is satisfied:

- **Property A:** The point p_m is on a vertex of $\text{Emb}(L)$.
- **Property B:** The point p_m is on the interior of an edge $\overline{uv}$, M includes u and there is also a discrete option for P_m that includes v .
- **Property C:** The point p_m is on the interior of an edge $\overline{uv}$ and M includes some $w \in \{u, v\}$ such that $\text{len}_{\|\cdot\|_\infty}(P_m) + 1 \geq |M| + \|p_m - w\|_\infty$.

Base case ($m = 1$): In this case $P_m = P_1 = p_1$, so P_1 has length 0.

If $p_1 \in V(L)$ then the discrete option is $M = \{p_1\}$, because it contains p_1 and $|M| = 1 = \text{len}_{\|\cdot\|_\infty}(P_1) + 1$. So property A holds.

Otherwise p_1 lies on the interior of an edge $\overline{uv}$ of L . Then $\{u\}$ and $\{v\}$ are both discrete options for P_1 , because removing either vertex removes uv and both sets have cardinality only 1. So we have proven that property B holds.

Inductive step ($m > 1$):

Now assume one of the three properties holds for P_{m-1} .

Property A: If property A holds for P_{m-1} then p_{m-1} is a vertex and a discrete option M for P_{m-1} exists. Of course $p_{m-1} \in M$ must hold. Now consider the unit square $[k_m, k_m + 1] \times [l_m, l_m + 1]$ that contains $\overline{p_{m-1} p_m}$. Name the corner vertices clockwise $w = p_{m-1}, v_1, v_2, v_3$. We know that the endpoint p_m of $\overline{p_{m-1} p_m}$ must be on the boundary of this unit square. We handle cases depending on the position of p_m . Observe possible positions of p_m in Figure 5.2 and in which cases they are handled.

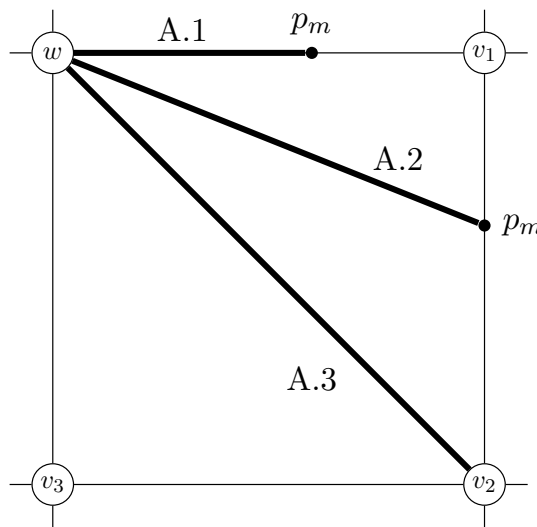


Figure 5.2: Possible placements of $\overline{p_{m-1} p_m}$ in Case A.

Case A.1: Point p_m is on the interior of $\overline{wv_1}$ or $\overline{wv_3}$.

Here M is already a discrete option for P_m because the edges are incident to $w = p_{m-1}$ which is in M . Also the following holds:

$$\text{len}_{\|\cdot\|_\infty}(P_m) + 1 = \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \|p_m - w\|_\infty + 1 \geq |M| + \|p_m - w\|_\infty$$

With that, we have shown that property C holds for P_m .

Case A.2: Point p_m is on the interior of $\overline{v_1 v_2}$ or $\overline{v_2 v_3}$.

W.l.o.g let p_m be on the interior of $\overline{v_1 v_2}$. We can take any $u \in \{v_1, v_2\}$ to get $M' := M \cup \{u\}$. Now M' removes $v_1 v_2$ and it holds that $\text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1} p_m}) \geq 1$ because p_{m-1} is not incident to $v_1 v_2$. With the Auxiliary Lemma 5.8 it follows that M' is a discrete option for P_m . We can conclude that property B holds for P_m , because u can be any of the endpoints of the edge p_m lies on.

Case A.3: Point p_m is on v_1, v_2 or v_3 .

We define $M' = M \cup \{p_m\}$, so p_m gets removed by M' . Again, it holds that $\text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1} p_m}) \geq 1$ because p_{m-1} and p_m are both vertices. So with the Auxiliary Lemma 5.8 we get that M' is a discrete option for P_m and property A holds as well.

Property B: If property B holds for P_{m-1} , then p_{m-1} lies on the interior of an edge $\overline{uv}$ and there is a discrete option M_u with $u \in M_u$ and a discrete option M_v with $v \in M_v$. Take the unit square that $\overline{p_{m-1} p_m}$ lies in and label the vertices on the corners of it in a circle u, v, v' and u' . Observe possible positions of p_m in Figure 5.3 and in which cases they are handled.

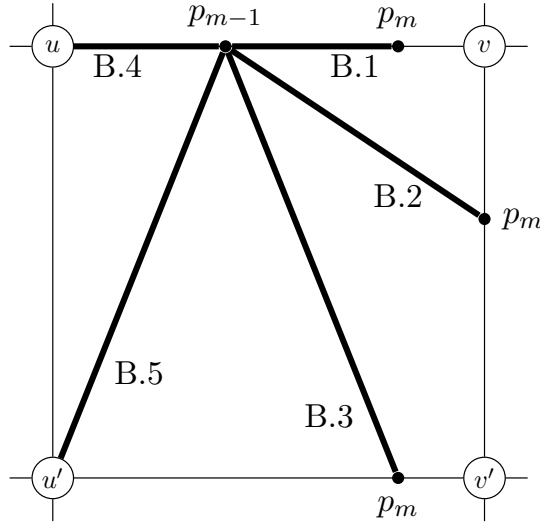


Figure 5.3: Possible placements of $\overline{p_{m-1} p_m}$ in Case B.

Case B.1: Point p_m lies on the interior of $\overline{uv}$.

In this case property B directly follows for P_m as well because no new edge or vertex is touched and therefore M_u and M_v are discrete options for P_m as well.

Case B.2: Point p_m lies on the interior of $\overline{uu'}$ or $\overline{vv'}$.

W.l.o.g. let p_m be on the interior of $\overline{vv'}$. We can take M_v as a discrete option for P_m because it removes vv' . Also the length of $\overline{p_{m-1}p_m}$ is at least $\|p_m - v\|_\infty$ which enables us to show:

$$\begin{aligned} \text{len}_{\|\cdot\|_\infty}(P_m) + 1 &= \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1}p_m}) + 1 \\ &\geq \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \|p_m - v\|_\infty + 1 \\ &\geq |M_u| + \|p_m - v\| \end{aligned}$$

We can conclude that property C holds for P_m .

Case B.3: Point p_m lies on the interior of $\overline{u'v'}$.

The sets $M_u \cup \{u'\}$ and $M_u \cup \{v'\}$ both remove edge $u'v'$. Also the length of $\overline{p_{m-1}p_m}$ is at least 1, because the points are on non-adjacent edges. With the Auxiliary Lemma 5.8 we can conclude for both $M_u \cup \{u'\}$ and $M_u \cup \{v'\}$ that they are discrete options for P_m . We end up with property B being true for P_m .

Case B.4: Point p_m lies on $w \in \{u, v\}$.

Take M_w , it removes the vertex w , so it is a discrete option for P_m and we end up with property A.

Case B.5: Point p_m lies on $w \in \{u', v'\}$.

We define $M' = M \cup \{w\}$ to make sure w gets removed. It holds that $\text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1}p_m}) \geq 1$ because edge uv is not incident to $w = p_m$. By the Auxiliary Lemma 5.8 we get that M' is a discrete option for P_m and property A holds as well.

Property C: If property C holds for P_{m-1} , then p_{m-1} lies on the interior of an edge $\overline{uv}$ and there is a discrete option M for P_{m-1} that contains u or v , w.l.o.g. $v \in M$, such that $\text{len}_{\|\cdot\|_\infty}(P_{m-1}) + 1 \geq |M| + \|p_{m-1} - v\|_\infty$. Take the unit square that $\overline{p_{m-1}p_m}$ lies in and name the vertices on the corners of it in a circle u, v, v' and u' . Observe possible positions of p_m in Figure 5.4 and in which cases they are handled.

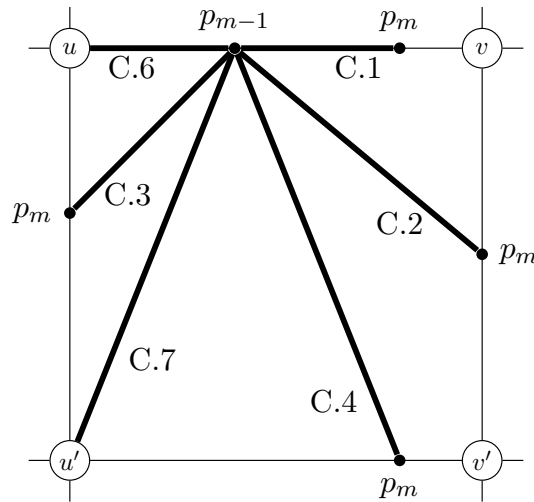


Figure 5.4: Possible placements of $\overline{p_{m-1}p_m}$ in Case C.

Case C.1: Point p_m lies on the interior of $\overline{uv}$.

The set M is a discrete option for P_m because no new vertices or edges compared to P_{m-1} are touched. Also the following holds:

$$\begin{aligned} \text{len}_{\|\cdot\|_\infty}(P_m) + 1 &= \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1}p_m}) + 1 \\ &\geq |M| + \|p_{m-1} - v\|_\infty + \|p_{m-1} - p_m\|_\infty \\ &\geq |M| + \|p_m - v\|_\infty \end{aligned}$$

So all conditions for property C hold for P_m .

Case C.2: Point p_m lies on the interior of $\overline{vv'}$.

We can take M as a discrete option for P_m because it contains v and therefore removes vv' . Also the length of $\overline{p_{m-1}p_m}$ is at least $\|p_m - v\|_\infty$ which enables us to show:

$$\begin{aligned} \text{len}_{\|\cdot\|_\infty}(P_m) + 1 &= \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1}p_m}) + 1 \\ &\geq \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + \|p_m - v\|_\infty + 1 \\ &\geq |M| + \|p_m - v\|_\infty \end{aligned}$$

We can conclude that property C holds for P_m .

Case C.3: Point p_m lies on the interior of $\overline{uu'}$.

In this case we can prove that $M \cup \{u\}$ and $M \cup \{u'\}$ are both discrete options for P_m . Of course they both remove the edge uu' . Also the distance from v to p_m is at least 1 because v is not incident to uu' and we can use the triangle inequality to show:

$$\|p_{m-1} - v\|_\infty + \|p_m - p_{m-1}\|_\infty \geq \|p_m - v\|_\infty \geq 1$$

With this we can prove:

$$\begin{aligned} |M \cup \{u\}| &\leq |M| + 1 \leq \text{len}_{\|\cdot\|_\infty}(P_{m-1}) + 1 - \|p_{m-1} - v\|_\infty + 1 \\ &= \text{len}_{\|\cdot\|_\infty}(P_m) - \text{len}_{\|\cdot\|_\infty}(\overline{p_{m-1}p_m}) - \|p_{m-1} - v\|_\infty + 2 \\ &= \text{len}_{\|\cdot\|_\infty}(P_m) - (\|p_m - p_{m-1}\|_\infty + \|p_{m-1} - v\|_\infty) + 2 \\ &\leq \text{len}_{\|\cdot\|_\infty}(P_m) - 1 + 2 \\ &= \text{len}_{\|\cdot\|_\infty}(P_m) + 1 \end{aligned}$$

The same goes for $M \cup \{u'\}$, so for P_m property B holds.

Case C.4: Point p_m lies on the interior of $\overline{u'v'}$.

The sets $M \cup \{u\}$ and $M \cup \{v\}$ both remove the edge $u'v'$ and the length of $\overline{p_{m-1}p_m}$ is at least 1 because the points are on non-adjacent edges. By the Auxiliary Lemma 5.8 we can conclude that both $M \cup \{u\}$ and $M \cup \{v\}$ are discrete options for P_m and we end up with property B being true for P_m .

Case C.5: Point p_m lies on v .

The set M is already a discrete option for P_m , because $v \in M$ and we end up with property A.

Case C.6: Point p_m lies on u .

Now $p_m = u$ has distance 1 to v and we can argue in the same way as in case 3 that $|M \cup \{u\}| \leq \text{len}_{\|\cdot\|_\infty}(P_m) + 1$. So $M \cup \{u\}$ is a discrete option for P_m and we end up with property A.

Case C.7: Point p_m lies on v' or u' .

Because p_m is not incident to uv that p_{m-1} lies on, we have $\|p_m - p_{m-1}\|_\infty \geq 1$. So $M \cup \{p_m\}$ is a discrete option for P_m by the Auxiliary Lemma 5.8 and property A holds as well.

With that the induction is complete and we can conclude that there exists a discrete option M for P_n . Because P_n touches at all elements of L that γ touches and $\text{len}_{\|\cdot\|_\infty}(P_n) \leq \text{len}_{\|\cdot\|_\infty}(\gamma)$ holds, M is also a discrete option for γ . $\square$

This Proposition is an interesting result in itself. One can easily see that the same statement holds as well for any norm that assigns larger values to vectors of $\mathbb{R}^2$, for example the Euclidean norm or any other p -norm.

5.2 Dividing Grid Graphs along Cuts

Using Proposition 5.5, we show how a cut of a shape can be transformed to a vertex subset of a grid graph that fits into the shape. This vertex subset then divides the embedding of the grid graph in the same way the cut divides the shape.

Lemma 5.9. *Let G be a grid graph and S a shape such that $\text{Emb}(G) \subseteq S$. Let $\gamma = \bigcup_{i=1}^{\text{cc}(\gamma)} I(\gamma_i)$ be a cut in S and the remaining shapes S_γ^i for $i \in \{1, \dots, m\}$. Then there exists a set $M \subseteq V(G)$ of size at most $\text{len}_{\|\cdot\|_\infty}(\gamma) + \text{cc}(\gamma)$ and such that $G - M = \bigcup_{i=1}^m G_i$ consists of the pairwise disjoint components $G_1, \dots, G_m$ and for all $i, j \in \{1, \dots, m\}, i \neq j$: $G_i \cap G_j = \emptyset$ and $\text{Emb}(G_i) \subseteq S_\gamma^i$.*

Proof. For every $i \in \{1, \dots, \text{cc}(\gamma)\}$, consider curve γ_i . By Proposition 5.5 there exists a discrete option M_i with $|M_i| \leq \text{len}_{\|\cdot\|_\infty}(\gamma_i) + 1$. We define $M'_i = M_i \cap V(G)$ and $M = \bigcup_{i=1}^{\text{cc}(\gamma)} M'_i$. Now the following holds:

$$|M| = \left| \bigcup_{i=1}^{\text{cc}(\gamma)} M'_i \right| \leq \sum_{i=1}^{\text{cc}(\gamma)} |M'_i| \leq \sum_{i=1}^{\text{cc}(\gamma)} (\text{len}_{\|\cdot\|_\infty}(\gamma_i) + 1) = \text{len}_{\|\cdot\|_\infty}(\gamma) + \text{cc}(\gamma)$$

We now need to argue that M divides G . Define $G' := G - M$, then it holds that $\text{Emb}(G') \cap \gamma = \emptyset$ because all elements that the curves in γ touched are removed by deleting the vertices in M . Also $\text{Emb}(G') \subseteq S \setminus \gamma$ holds and we know that $S \setminus \gamma$ consists of $m \geq 2$ connected components C_γ^i ($i \in \{1, \dots, m\}$). Remember that the remaining shapes are exactly $S_\gamma^i = \overline{C_\gamma^i}$. For every $i \in \{1, \dots, m\}$ define $V_i := C_\gamma^i \cap V(G')$ and $G_i = G[V_i]$. It must hold that the V_i are pairwise disjoint because the C_γ^i are connected components and therefore also pairwise disjoint. This implies that for all $i, j \in \{1, \dots, m\}, i \neq j$: $G_i \cap G_j = \emptyset$.

We first show $\text{Emb}(G_i) \subseteq S_\gamma^i$ for all $i \in \{1, \dots, m\}$. We know that $V(G_i) = V_i \subseteq C_\gamma^i$ by construction. If there was an edge $e \in E(\text{Emb}(G_i))$ that was not entirely in C_γ^i then it must touch γ because it is within S and its endpoints are in C_γ^i . But then one adjacent vertex would be in M , a contradiction because the vertices in M were removed. So $\text{Emb}(G_i) \subseteq C_\gamma^i \subseteq S_\gamma^i$ holds for all $i \in \{1, \dots, m\}$.

We now need to prove $G' = \bigcup_{i=1}^m G_i$. The direction " $\supseteq$ " holds by construction. Take any vertex $v \in V(\text{Emb}(G'))$ then it does not touch γ so it is in $S \setminus \gamma$, therefore it is in some C_γ^i and with that in G_i . Now take any edge $uv \in E(G')$. If $u, v \in V(G_i)$ for some $i \in \{1, \dots, m\}$, then $uv \in E(G_i)$ because G_i is an induced subgraph of G . If $u \in G_i$ and $v \in G_j$ for $i, j \in \{1, \dots, m\}, i \neq j$, then $u \in C_\gamma^i$ and $v \in C_\gamma^j$ would hold. But the two components C_γ^i and C_γ^j are disconnected by γ and because $\overline{uv} \in S$ we can conclude that $\overline{uv}$ must touch γ . So one endpoint u or v would be in M , a contradiction because they are both in $V(G')$. This finishes the proof. $\square$

5.3 Transforming Continuous to Discrete Decomposition Trees

Now that we know how a cut is applied to a grid graph, it is not hard to transform a decomposition tree of a shape into a decomposition tree of a grid graph that the shape fits into.

Lemma 5.10. *Let G be a connected grid graph and S a shape such that $\text{Emb}(G) \subseteq S$. Let D be a decomposition tree for S , then there exists a decomposition tree D' for G that is isomorphic to D and the isomorphism $f : V(D) \rightarrow V(D')$ also satisfies the following additional properties:*

- $f(S) = G$
- For any shape $X \in V(D)$ it holds that $\text{Emb}(f(X)) \subseteq X$.
- For any non-leaf $X \in V(D)$ it holds that $|\text{dset}(f(X))| \leq \text{len}_{\|\cdot\|_\infty}(\text{cut}(X)) + \text{cc}(\text{cut}(X))$

Proof. We inductively construct D' starting with the root. We set $f(S) := G$ for the base, so $\text{Emb}(f(S)) = \text{Emb}(G) \subseteq S$ already holds. Now for the step from any non-leaf $X \in V(D)$ to the children $X_1, \dots, X_m$, we assume that $H := f(X)$ and $\text{Emb}(H) \subseteq X$ holds. Define $\gamma := \text{cut}(X)$, then by Lemma 5.9 for graph H , shape X and cut γ , we get a set $M \subseteq V(H)$ such that $H - M = \bigcup_{i \in \{1, \dots, m\}} H_i$ with the H_i being pairwise disjoint and $\text{Emb}(H_i) \subseteq X_i$ for all $i \in \{1, \dots, m\}$. Define $\text{dset}(H) := M$ and insert all $H_1, \dots, H_m$ as children of H in D' such that $f(X_i) = H_i$. Then the induction step is complete because $\text{Emb}(f(X_i)) = \text{Emb}(H_i) \subseteq X_i$ for all $i \in \{1, \dots, m\}$. We also know by Lemma 5.9 that $|M| \leq \text{len}_{\|\cdot\|_\infty}(\gamma) + \text{cc}(\gamma)$, so we can conclude:

$$|\text{dset}(f(X))| = |M| \leq \text{len}_{\|\cdot\|_\infty}(\gamma) + \text{cc}(\gamma) = \text{len}_{\|\cdot\|_\infty}(\text{cut}(X)) + \text{cc}(\text{cut}(X))$$

By construction D' is also already a valid decomposition tree of G . □

This proves that any decomposition tree of a shape that a grid graph fits into can be applied to the grid graph as well and the leftover graphs will be at most as big as the leftover shapes of the original tree. This works with only slightly larger cost, precisely at most an extra +1 per curve that is used. If all cuts only consist of one curve, which is going to be the case in most decompositions in this work, then every deletion set is at most the size of the length of its corresponding cut plus 1.

For the remainder of the chapter, we fix $\|\cdot\|_\infty$ to be the norm. We now take an ε -decomposition tree of a shape S which means the leaf shapes of the tree all have a diameter smaller than ε . If we transform it into a decomposition tree of a grid graph G , then any two vertices in the embedding of a fixed leaf graph must be less than ε apart from each other. Choosing $\varepsilon = 1$, this implies that all leaf graphs have at most one vertex because the distance between two vertices in a grid graph is always at least 1. This enables us to show Theorem 5.11 stating that any 1-decomposition tree of a shape provides an upper bound for the treedepth of any grid graph that fits into the shape.

Theorem 5.11. *Let S be a shape, G a grid graph such that $\text{Emb}(G) \subseteq S$ and D a 1-decomposition tree of S considering the norm $\|\cdot\|_\infty$. Then the following holds:*

$$\text{td}(G) \leq h_{\|\cdot\|_\infty}(D) + \text{cc}(D) + 1$$

Proof. By Lemma 5.10 there exists an isomorphic decomposition tree D' of G and the isomorphism $f : V(D) \rightarrow V(D')$. Take a leaf H in D' such that $\text{depth}(H) = h(D')$.

Let $G = H_0, H_1, \dots, H_n = H$ be the unique path from root G to leaf H . Also, for all $i \in \{0, \dots, n\}$ let $X_i \in V(D)$ such that $f(X_i) = H_i$ and $X := X_n$. Now every X_i is the shape that contains the embedding of H_i . The following upper bound for the height of D' holds:

$$\begin{aligned}
 h(D') &= \text{depth}(H) \\
 &= \sum_{i=0}^{n-1} |\text{dset}(H_i)| \\
 &\leq \sum_{i=0}^{n-1} (\text{len}_{\|\cdot\|_\infty}(\text{cut}(X_i)) + \#\text{cut}(X_i)) && \text{Lemma 5.10} \\
 &= \sum_{i=0}^{n-1} \text{len}_{\|\cdot\|_\infty}(\text{cut}(X_i)) + \sum_{i=0}^{n-1} \#\text{cut}(X_i) \\
 &= \text{depth}_{\|\cdot\|_\infty}(X) + \text{cc}(X) \\
 &\leq h_{\|\cdot\|_\infty}(D) + \text{cc}(D)
 \end{aligned}$$

Now for any leaf H in D' , there is a leaf X in D such that $f(X) = H$. We also know that $\text{Emb}(H) \subseteq X$ and $\text{diam}_{\|\cdot\|_\infty}(X) < 1$. That means that H is a graph on at most one vertex. To upper bound $\text{td}(G)$ we need all leaves to be graphs on zero vertices. For that construct a new decomposition tree from D' in the following way. For all leaves H in D' that are graphs on one vertex, attach a new graph H' as the new leaf. Let v be the single vertex in H , we set $\text{dset}(H) = \{v\}$, so the graph H' has zero vertices. Name the resulting decomposition tree D'' . Now all leaves in D'' are graphs on zero vertices, so D'' gives an upper bound on the treedepth of G by Corollary 3.6:

$$\text{td}(G) \leq h(D'')$$

The height of D'' is at most $h(D') + 1$ because any path from root to leaf in D'' needs at most one extra step with a deletion set of size 1. We can conclude:

$$\begin{aligned}
 \text{td}(G) &\leq h(D'') \\
 &\leq h(D') + 1 \\
 &\leq h_{\|\cdot\|_\infty}(D) + \text{cc}(D) + 1
 \end{aligned}$$

□

Finally, this Theorem formalizes that the intuition that an upper bound on the continuous treedepth of a shape implies an upper bounds on the treedepth of grid graphs that fit into this shape. The built connection, though, is not as strong as one might hope. On the one hand, the curve count prevents us from having a clean $\text{td} \leq \text{ctd}$ relation. On the other hand, for a shape S with dimensions in $\mathcal{O}(n)$ and a grid graph G with $\text{Emb}(G) \subseteq S$, we also cannot derive something like $\text{td}(G) \leq \text{ctd}_1(S) + o(n)$ because we have no guarantee that for near optimal decomposition trees of the shape, a sublinear term bounds the curve count. It is likely that it is at most of logarithmic order because a cut usually has a curve count in $\mathcal{O}(1)$ and the dimensions of the shape can be decreased exponentially, which means a logarithmic number of cuts is needed just as in the example of section 4.2. For that reason, the Theorem still allows us to improve the linear term of the upper bound on the treedepth of grid graphs, which we see in the next chapter 6.

6. Upper Bounds

In this chapter we aim to prove upper bounds for specific shapes and grid graphs. For that, we first construct ε -decomposition trees for the shapes and measure their height and curve count. These trees allow us to upper bound the continuous treedepth of the shapes as well as the treedepth of the corresponding grid graphs using Theorem 5.11. We first define the specific shapes and subgraphs in Section 6.1 and then tackle the shapes one by one in the following Sections 6.3 to 6.6. An overview of the best bounds proven is displayed in section 6.2. In this chapter the considered norm is always the Chebyshev norm $\|\cdot\|_\infty$.

6.1 Shape and Grid Graph Definitions

Definition 6.1. *The continuous rotated rectangle, denoted by $\mathcal{R}_{a,b}$ ($a, b \in \mathbb{R}_{>0}$), is the shape:*

$$\mathcal{R}_{a,b} := \{(x, y) \in [0, a+b] \times [0, a+b] \mid \max\{a-x, x-a\} \leq y \leq \min\{a+x, a+2b-x\}\}$$

Definition 6.2. *The discrete rotated rectangle, denoted by $R_{m,n}$ ($m, n \in \mathbb{N} \setminus \{1\}$), is the induced grid graph on the following vertex set:*

$$V(R_{m,n}) := \mathbb{Z}^2 \cap \mathcal{R}_{m-1, n-1}$$

Definition 6.3. *The continuous triangle, denoted by $\mathcal{T}_a$ ($a \in \mathbb{R}_{>0}$), is the shape:*

$$\mathcal{T}_a := \{(x, y) \in [0, a] \times [0, a] \mid 0 \leq y \leq a-x\}$$

Definition 6.4. *The discrete triangle, denoted by T_k ($k \in \mathbb{N} \setminus \{1\}$), is the induced grid graph on the following vertex set:*

$$V(T_{m,n}) := \mathbb{Z}^2 \cap \mathcal{T}_{k-1}$$

Definition 6.5. *The continuous wedge, denoted by $\mathcal{W}_a$ ($a \in \mathbb{R}_{>0}$), is the shape:*

$$\mathcal{W}_a := \{(x, y) \in [0, \frac{a}{2}] \times [0, a] \mid x \leq y \leq a-x\}$$

Shape	Upper Bound	Grid Graph	Upper Bound
$\mathcal{R}_{a,b}$	$2a + b$	$R_{m,n}$	$2m + n + \mathcal{O}(\log n)$
$\mathcal{W}_a$	$\frac{1+\sqrt{57}}{8}a \approx 1.069 a$	W_k	$1.069 k + \mathcal{O}(\log k)$
$\mathcal{T}_a$	$\frac{5+\sqrt{57}}{8}a \approx 1.569 a$	T_k	$1.569 k + \mathcal{O}(\log k)$
$\mathcal{S}_a$	$\frac{87+7\sqrt{57}}{42+2\sqrt{57}}a \approx 2.450 a$	G_k	$2.450 k + \mathcal{O}(\log k)$

Table 6.1: Upper bounds on the treedepth of shapes and grid graphs.

Definition 6.6. The discrete wedge, denoted by W_k ($k \in \mathbb{N} \setminus \{1\}$), is the induced grid graph on the following vertex set:

$$V(W_{m,n}) := \mathbb{Z}^2 \cap \mathcal{W}_{k-1}$$

Definition 6.7. The square, denoted by $\mathcal{S}_a$ ($a \in \mathbb{R}_{>0}$), is the shape:

$$\mathcal{S}_a := [0, a] \times [0, a]$$

Note that the $\mathcal{S}_{k-1}$ corresponds to the square grid G_k .

6.2 Overview of the Best Upper Bounds

In the following sections, the upper bounds are proven step by step. In Table 6.1, all upper bounds proven in this section are collected for a quick overview.

6.3 The Rotated Rectangle

In this section we upper bound the treedepth of the continuous as well as the discrete rotated rectangle. For that we first construct a recursive ε -decomposition tree in the following Lemma 6.8 and then conclude bounds for the treedepth.

Lemma 6.8. For the shape $\mathcal{R}_{a,b}$ with $a \leq b$ and any $\varepsilon > 0$ with $\varepsilon \leq b$ there exists an ε -decomposition tree D that satisfies the following properties:

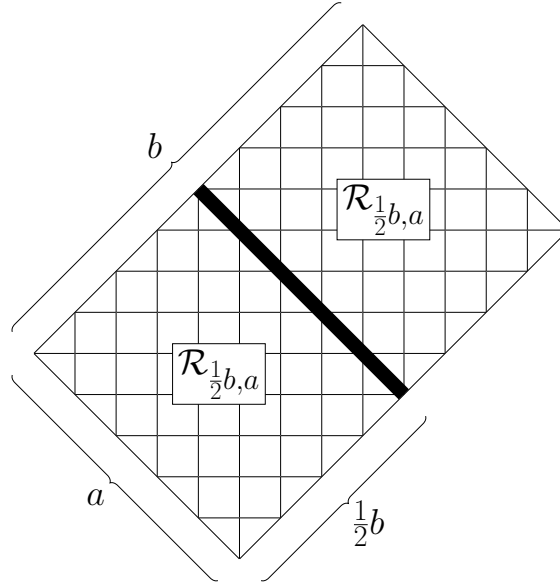
1. $h(D) < 2a + b$
2. $cc(D) = 2\lfloor \log_2(b/\varepsilon) \rfloor + 4$

Proof. Consider the shape $\mathcal{R}_{a,b}$ with $a \leq b$ and any $\varepsilon > 0$ with $\varepsilon \leq b$. Cutting the shape in half according to Figure 6.1 using a cut of length a leaves two shapes that are $\|\cdot\|_\infty$ -isometric to $\mathcal{R}_{\frac{b}{2},a}$. We get the following decomposition tree:

$$\mathcal{R}_{a,b} \xrightarrow[1]{a} \mathcal{R}_{\frac{b}{2},a}$$

We cut $\mathcal{R}_{\frac{b}{2},a}$ in the same way to get:

$$\mathcal{R}_{a,b} \xrightarrow[1]{a} \mathcal{R}_{\frac{b}{2},a} \xrightarrow[1]{\frac{b}{2}} \mathcal{R}_{\frac{a}{2},\frac{b}{2}}$$


 Figure 6.1: Decomposition of the continuous rotated rectangle $\mathcal{R}_{a,b}$.

We now define for $t \in \mathbb{N}_0$: $a_t := (1/2)^t a$ and $b_t := (1/2)^t b$. Still, $a_t \leq b_t$ holds and the decomposition tree for any $t \in \mathbb{N}_0$ becomes:

$$\mathcal{R}_{a_t, b_t} \xrightarrow[1]{a_t} \mathcal{R}_{b_{t+1}, a} \xrightarrow[1]{b_{t+1}} \mathcal{R}_{a_{t+1}, b_{t+1}}$$

We define the following recursive decomposition tree $D_\varepsilon(\mathcal{R}_{a_t, b_t})$ for any $\mathcal{R}_{a_t, b_t}$.

If $a_t < \varepsilon/2$ and $b_t < \varepsilon/2$, then define $D_\varepsilon(\mathcal{R}_{a_t, b_t}) := \mathcal{R}_{a_t, b_t}$, otherwise:

$$D_\varepsilon(\mathcal{R}_{a_t, b_t}) := \mathcal{R}_{a_t, b_t} \xrightarrow[1]{a_t} \mathcal{R}_{b_{t+1}, a} \xrightarrow[1]{b_{t+1}} D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})$$

Now every leaf in $D_\varepsilon(\mathcal{R}_{a_t, b_t})$ has $\|\cdot\|_\infty$ -diameter smaller ε because only the base case is a leaf and if $a_t, b_t < \varepsilon/2$ then $\text{diam}_{\|\cdot\|_\infty}(\mathcal{R}_{a_t, b_t}) = a_t + b_t < \varepsilon$. We set $D := D_\varepsilon(\mathcal{R}_{a, b}) = D_\varepsilon(\mathcal{R}_{a_0, b_0})$ and conclude that D is an ε -decomposition tree of $\mathcal{R}_{a, b}$.

We now calculate the height of the tree. For that we need to determine $\tilde{t} \in \mathbb{N}_0$ such that $a_{\tilde{t}} \leq b_{\tilde{t}} < \frac{\varepsilon}{2} \leq b_{\tilde{t}-1}$. Set $\tilde{t} := \lfloor \log_2(b/\varepsilon) \rfloor + 2$. Because $\varepsilon \leq b$, it follows that $\tilde{t} \geq 2$. Furthermore, the two inequalities hold:

$$b_{\tilde{t}} = 2^{-\tilde{t}} b = 2^{-\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor - 2} b < 2^{-(\log_2(\frac{b}{\varepsilon}) - 1) - 2} b = \frac{\varepsilon}{2b} b = \frac{\varepsilon}{2}$$

$$b_{\tilde{t}-1} = 2^{-(\tilde{t}-1)} b = 2^{-\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor - 2 + 1} b \geq 2^{-\log_2(\frac{b}{\varepsilon}) - 1} b = \frac{\varepsilon}{2b} b = \frac{\varepsilon}{2}$$

By backward induction, we now prove that $h(D_\varepsilon(\mathcal{R}_{a_t, b_t})) < 2a_t + b_t$ for all $t \in \{0, \dots, \tilde{t}\}$.

Base case ($t = \tilde{t}$): We know $a_{\tilde{t}} \leq b_{\tilde{t}} < \frac{\varepsilon}{2}$, so $D_\varepsilon(\mathcal{R}_{a_{\tilde{t}}, b_{\tilde{t}}}) = \mathcal{R}_{a_{\tilde{t}}, b_{\tilde{t}}}$ and therefore:

$$h(D_\varepsilon(\mathcal{R}_{a_{\tilde{t}}, b_{\tilde{t}}})) = 0 < 2a_{\tilde{t}} + b_{\tilde{t}}$$

Inductive step ($t \in \{0, \dots, \tilde{t} - 1\}$): Assume that $h(D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) < 2a_{t+1} + b_{t+1}$ holds.

We know $b_t \geq \varepsilon/2$ holds, so we can calculate the height of $D_\varepsilon(\mathcal{R}_{a_t, b_t})$:

$$\begin{aligned}
 h(D_\varepsilon(\mathcal{R}_{a_t, b_t})) &= h(\mathcal{R}_{a_t, b_t} \xrightarrow[1]{a_t} \mathcal{R}_{b_{t+1}, a} \xrightarrow[1]{b_{t+1}} D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) \\
 &= a_t + b_{t+1} + h(D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) \\
 &< a_t + b_{t+1} + 2a_{t+1} + b_{t+1} && \text{(induction hypothesis)} \\
 &= a_t + \frac{1}{2}b_t + 2\frac{1}{2}a_t + \frac{1}{2}b_t \\
 &= 2a_t + b_t
 \end{aligned}$$

This finishes the induction and with $a = a_0$ and $b = b_0$ we can conclude the first property:

$$h(D) = h(D_\varepsilon(\mathcal{R}_{a, b})) < 2a + b$$

We now calculate the curve count of the tree.

Use backward induction to prove: $\text{cc}(D_\varepsilon(\mathcal{R}_{a_t, b_t})) = 2(\tilde{t} - t)$ for all $t \in \{0, \dots, \tilde{t}\}$

Base case ($t = \tilde{t}$): We know $D_\varepsilon(\mathcal{R}_{a_{\tilde{t}}, b_{\tilde{t}}})$ is a leaf and it follows:

$$\text{cc}(D_\varepsilon(\mathcal{R}_{a_{\tilde{t}}, b_{\tilde{t}}})) = 0 = 2(\tilde{t} - \tilde{t})$$

Inductive step ($t \in \{0, \dots, \tilde{t} - 1\}$): Assume $\text{cc}(D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) = 2(\tilde{t} - (t + 1))$ holds.

We know $b_t \geq \varepsilon/2$ holds, so we can calculate the curve count of $D_\varepsilon(\mathcal{R}_{a_t, b_t})$:

$$\begin{aligned}
 \text{cc}(D_\varepsilon(\mathcal{R}_{a_t, b_t})) &= \text{cc}(\mathcal{R}_{a_t, b_t} \xrightarrow[1]{a_t} \mathcal{R}_{b_{t+1}, a} \xrightarrow[1]{b_{t+1}} D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) \\
 &= 2 + \text{cc}(D_\varepsilon(\mathcal{R}_{a_{t+1}, b_{t+1}})) \\
 &= 2 + 2(\tilde{t} - (t + 1)) && \text{(induction hypothesis)} \\
 &= 2(\tilde{t} - t)
 \end{aligned}$$

This finishes the induction and gives us the curve count of D :

$$\begin{aligned}
 \text{cc}(D) &= \text{cc}(D(\mathcal{R}_{a_0, b_0})) \\
 &= 2(\tilde{t} - 0) \\
 &= 2(\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor + 2) \\
 &= 2\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor + 4
 \end{aligned}$$

With that the second property of the tree is proven. □

Corollary 6.9. $\text{ctd}(\mathcal{R}_{a, b}) \leq 2a + b$

Proof. For any $\varepsilon > 0$ with $\varepsilon \leq b$, the ε -decomposition tree D of Lemma 6.8 provides an upper bound for the ε -treedepth of $\mathcal{R}_{a, b}$:

$$\text{ctd}_\varepsilon(\mathcal{R}_{a, b}) \leq h(D) < 2a + b$$

Because ε can be chosen arbitrarily small, we conclude:

$$\text{ctd}(\mathcal{R}_{a, b}) \leq 2a + b$$

□

Corollary 6.10. *For $m \leq n$ the following holds:*

$$\text{td}(R_{m,n}) \leq 2m + n + 2\lfloor \log_2(n-1) \rfloor + 1 \in 2m + n + \mathcal{O}(\log n)$$

Proof. Define $a := m - 1$ and $b := n - 1$, so $a \leq b$ is satisfied. It holds that $\text{Emb}(R_{m,n}) \subseteq \mathcal{R}_{m-1,n-1} = \mathcal{R}_{a,b}$. By Lemma 6.8 there exists an ε -decomposition tree D of $\mathcal{R}_{a,b}$ with height $2a + b$ and curve count $2\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor + 4$. Choosing $\varepsilon = 1$, we obtain a 1-decomposition tree of $\mathcal{R}_{a,b}$. Because $\text{Emb}(R_{m,n}) \subseteq \mathcal{R}_{a,b}$, by Theorem 5.11 the following upper bound holds:

$$\begin{aligned} \text{td}(R_{m,n}) &\leq h(D) + \text{cc}(D) + 1 \\ &\leq 2a + b + 2\lfloor \log_2(\frac{b}{\varepsilon}) \rfloor + 4 \\ &= 2(m-1) + n-1 + 2\lfloor \log_2(\frac{n-1}{1}) \rfloor + 4 \\ &= 2m + n + 2\lfloor \log_2(n-1) \rfloor + 1 \end{aligned}$$

□

6.4 The Wedge

Similar to the way we proved the upper bound for the rotated rectangle, we now prove upper bounds for the continuous and discrete wedge. Again, a recursive ε -decomposition tree is necessary that we construct in the following Lemma 6.11. This time the tree also builds on the decomposition tree of the rotated rectangle given by Lemma 6.8.

Lemma 6.11. *For the shape $\mathcal{W}_a$ and any $\varepsilon > 0$ with $\varepsilon \leq a$ there exists an ε -decomposition tree D that satisfies the following properties:*

1. $h(D) \leq \frac{1+\sqrt{57}}{8}a \approx 1.069a$
2. $\text{cc}(D) \leq \lfloor 2\log_2(\frac{a}{\varepsilon}) \rfloor + 3$

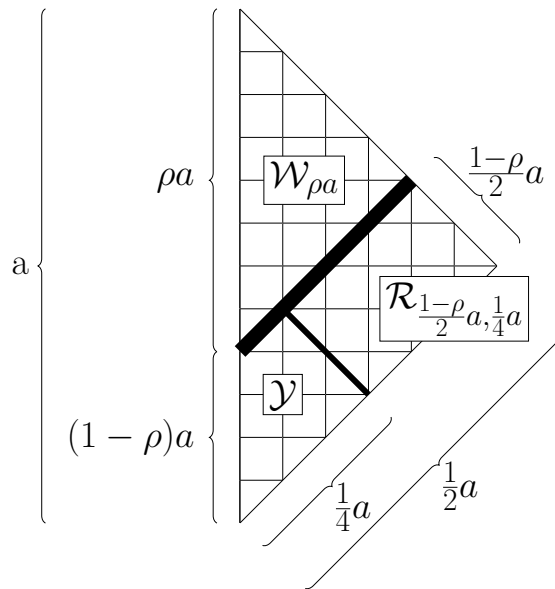


Figure 6.2: Decomposition of the continuous wedge $\mathcal{W}_a$.

Proof. Consider the shape $\mathcal{W}_a$ and any $\varepsilon > 0$ with $\varepsilon \leq a$. Define $\rho := \frac{13-\sqrt{57}}{8}$ and verify $\frac{1}{2} < \rho < \frac{1}{\sqrt{2}}$. Construct the decomposition tree D for $\mathcal{W}_a$ according to Figure 6.2. It uses two cuts:

1. The thick black line of length $\frac{\rho}{2}a$ cuts the wedge into a smaller wedge $\mathcal{W}_{\rho a}$ and a shape that we name $\mathcal{X}_{a,\rho}$.
2. The thinner black line of length $\frac{1-\rho}{2}a$ cuts $\mathcal{X}_{a,\rho}$ into $\mathcal{R}_{\frac{1-\rho}{2}a, \frac{1}{4}a}$ and a shape that we name $\mathcal{Y}$.

The shape $\mathcal{Y}$ is a subset of $\mathcal{R}_{\frac{1-\rho}{2}a, \frac{1}{4}a}$, so we can construct the tree:

$$\mathcal{W}_a \xrightarrow[\frac{1}{1}]{\frac{\rho}{2}a} \begin{cases} \mathcal{W}_{\rho a} \\ \mathcal{X}_{a,\rho} \xrightarrow[\frac{1}{1}]{\frac{1-\rho}{2}a} \mathcal{R}_{\frac{1-\rho}{2}a, \frac{1}{4}a} \end{cases}$$

For $t \in \mathbb{N}_0$ we define $a_t := \rho^t a$ and construct the decomposition tree:

$$\mathcal{W}_{a_t} \xrightarrow[\frac{1}{1}]{\frac{\rho}{2}a_t} \begin{cases} \mathcal{W}_{a_{t+1}} \\ \mathcal{X}_{a_t,\rho} \xrightarrow[\frac{1}{1}]{\frac{1-\rho}{2}a_t} \mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t} \end{cases}$$

To complete the tree to only having small leaves, we define it recursively and insert an ε -decomposition tree for the remaining rotated rectangle. For $\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}$ by Lemma 6.8 there exists the ε -decomposition tree $D_\varepsilon(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t})$ of height $(1-\rho)a_t + \frac{1}{4}a_t$ and curve count $2\lceil \log_2(\frac{a_t}{4\varepsilon}) \rceil + 4$. Note that the requirement $\frac{1-\rho}{2}a_t \leq \frac{1}{4}a_t$ holds because:

$$\frac{1-\rho}{2} < \frac{1-\frac{1}{2}}{2} = \frac{1}{4}$$

Now define the following recursive decomposition tree for any $\mathcal{W}_{a_t}$:

If $a_t < \varepsilon$, then define $D_\varepsilon(\mathcal{W}_{a_t}) := \mathcal{W}_{a_t}$, otherwise:

$$D_\varepsilon(\mathcal{W}_{a_t}) := \mathcal{W}_{a_t} \xrightarrow[\frac{1}{1}]{\frac{\rho}{2}a_t} \begin{cases} D_\varepsilon(\mathcal{W}_{a_{t+1}}) \\ \mathcal{X}_{a_t,\rho} \xrightarrow[\frac{1}{1}]{\frac{1-\rho}{2}a_t} D_\varepsilon(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}) \end{cases}$$

Every leaf in $D_\varepsilon(\mathcal{W}_{a_t})$ has $\|\cdot\|_\infty$ -diameter smaller ε because it is either in some $D_\varepsilon(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t})$ which is an ε -decomposition tree or is the shape $\mathcal{W}_{a_t}$ with $a_t < \varepsilon$ and therefore has $\text{diam}_{\|\cdot\|_\infty}(\mathcal{W}_{a_t}) = a_t < \varepsilon$. We set $D := D_\varepsilon(\mathcal{W}_a) = D_\varepsilon(\mathcal{W}_{a_0})$ and conclude that D is an ε -decomposition tree of $\mathcal{W}_a$.

We now calculate the height of the tree. Define $\tilde{t} := \lfloor 2\log_2(a/\varepsilon) \rfloor + 1$. Because $\varepsilon \leq a$, we know that $\tilde{t} \geq 1$. Verify:

$$\rho^2 < \left(\frac{1}{\sqrt{2}}\right)^2 = \frac{1}{2}$$

It follows that:

$$\begin{aligned} a_{\tilde{t}} &= \rho^{\tilde{t}} a \\ &= \rho^{\lfloor 2\log_2(a/\varepsilon) \rfloor + 1} a \\ &< \rho^{2\log_2(a/\varepsilon)} a \\ &= (\rho^2)^{\log_2(a/\varepsilon)} a \\ &\leq \left(\frac{1}{2}\right)^{\log_2(a/\varepsilon)} a \\ &= \varepsilon \end{aligned}$$

By backward induction, we now prove that $h(D_\varepsilon(\mathcal{W}_{a_t})) \leq (\frac{7}{4} - \rho)a_t$ for all $t \in \{0, \dots, \tilde{t}\}$.

Base case ($t = \tilde{t}$): We know $a_{\tilde{t}} < \varepsilon$, so $D_\varepsilon(\mathcal{W}_{a_{\tilde{t}}}) = \mathcal{W}_{a_{\tilde{t}}}$ and therefore:

$$h(D_\varepsilon(\mathcal{W}_{a_{\tilde{t}}})) = 0 < (\frac{7}{4} - \rho)a_{\tilde{t}}$$

Inductive step ($t \in \{0, \dots, \tilde{t} - 1\}$): Assume that $h(D_\varepsilon(\mathcal{W}_{a_{t+1}})) < (\frac{7}{4} - \rho)a_{t+1}$ holds.

Verify that $(\frac{7}{4} - \rho)\rho = \frac{7}{4} - \frac{3}{2}\rho$ holds:

$$\begin{aligned} \left(\frac{7}{4} - \rho\right)\rho &= \left(\frac{13}{4} - \rho\right)\rho - \frac{3}{2}\rho \\ &= \left(\frac{13}{4} - \left(\frac{13 - \sqrt{57}}{8}\right)\right)\rho - \frac{3}{2}\rho \\ &= \left(\frac{13 + \sqrt{57}}{8}\right)\rho - \frac{3}{2}\rho \\ &= \left(\frac{13 + \sqrt{57}}{8}\right)\left(\frac{13 - \sqrt{57}}{8}\right) - \frac{3}{2}\rho \\ &= \frac{169 - 57}{64} - \frac{3}{2}\rho \\ &= \frac{112}{64} - \frac{3}{2}\rho = \frac{7}{4} - \frac{3}{2}\rho \end{aligned}$$

We know if $a_t < \varepsilon$ then $D_\varepsilon(\mathcal{W}_{a_t})$ has height zero, which is upper bounded by $(\frac{7}{4} - \rho)a_t$, otherwise:

$$\begin{aligned} &h(D_\varepsilon(\mathcal{W}_{a_t})) \\ &= h\left(\mathcal{W}_{a_t} \xrightarrow[\frac{1}{1}]{\frac{\rho}{2}a_t} \left\{ \begin{array}{l} D_\varepsilon(\mathcal{W}_{a_{t+1}}) \\ \mathcal{X}_{a_t, \rho} \xrightarrow[\frac{1}{1}]{\frac{1-\rho}{2}a_t} D_\varepsilon(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}) \end{array} \right\}\right) \\ &= \frac{\rho}{2}a_t + \max \left\{ h(D_\varepsilon(\mathcal{W}_{a_{t+1}})) \right. \\ &\quad \left. \frac{1-\rho}{2}a_t + h\left(D_\varepsilon\left(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}\right)\right) \right\} \\ &\leq \frac{\rho}{2}a_t + \max \left\{ h(D_\varepsilon(\mathcal{W}_{a_{t+1}})) \right. && \text{(Lemma 6.8)} \\ &\quad \left. \frac{1-\rho}{2}a_t + (1 - \rho)a_t + \frac{1}{4}a_t \right\} \\ &\leq \frac{\rho}{2}a_t + \max \left\{ \begin{array}{l} (\frac{7}{4} - \rho)a_{t+1} \\ (\frac{3}{2} + \frac{1}{4} - \frac{3}{2}\rho)a_t \end{array} \right\} && \text{(induction hypothesis)} \\ &= \frac{\rho}{2}a_t + \max \left\{ \begin{array}{l} (\frac{7}{4} - \rho)\rho a_t \\ (\frac{7}{4} - \frac{3}{2}\rho)a_t \end{array} \right\} \\ &= (\frac{7}{4} - \rho)a_t && \left(\frac{7}{4} - \frac{3}{2}\rho = (\frac{7}{4} - \rho)\rho\right) \end{aligned}$$

This finishes the induction and with $a = a_0$ we conclude the first property:

$$h(D) = h(D_\varepsilon(\mathcal{W}_a)) \leq (\frac{7}{4} - \rho)a = \left(\frac{7}{4} - \frac{13 - \sqrt{57}}{8}\right)a = \frac{1 + \sqrt{57}}{8}a$$

We now calculate the curve count of the tree.

Use backward induction to prove: $\text{cc}(D_\varepsilon(\mathcal{W}_{a_t})) \leq \tilde{t} - t + 2$ for all $t \in \{0, \dots, \tilde{t}\}$

Base case ($t = \tilde{t}$):

$$\text{cc}(D_\varepsilon(\mathcal{W}_{a_{\tilde{t}}})) := \text{cc}(\mathcal{W}_{a_{\tilde{t}}}) = 0 \leq \tilde{t} - \tilde{t} + 2$$

Inductive step ($t \in \{0, \dots, \tilde{t} - 1\}$): Assume $\text{cc}((D_\varepsilon(\mathcal{W}_{a_{t+1}}))) = \tilde{t} - (t + 1) + 2$ holds.

We first upper bound the curve count of $D_\varepsilon(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t})$:

$$\begin{aligned} \text{cc}\left(D_\varepsilon\left(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}\right)\right) &= 2\lfloor \log_2\left(\frac{\frac{1}{4}a_t}{\varepsilon}\right) \rfloor + 4 && \text{(Lemma 6.8)} \\ &\leq 2\log_2\left(\frac{a_t}{4\varepsilon}\right) + 4 \\ &= 2\log_2\left(\frac{a_t}{\varepsilon}\right) \\ &= 2\log_2\left(\rho^t \frac{a}{\varepsilon}\right) \\ &= 2\log_2\left(\frac{a}{\varepsilon}\right) + 2\log_2(\rho)t \\ &\leq \lfloor 2\log_2(a/\varepsilon) \rfloor + 1 + 2\log_2\left(\frac{1}{\sqrt{2}}\right)t \\ &\leq \tilde{t} - t \end{aligned}$$

With that we upper bound the curve count of $D_\varepsilon(\mathcal{W}_{a_t})$. We know if $a_t < \varepsilon$ then $\text{cc}(D_\varepsilon(\mathcal{W}_{a_t})) = 0$, which is upper bounded by $\tilde{t} - t + 2$, otherwise:

$$\begin{aligned} \text{cc}(D_\varepsilon(\mathcal{W}_{a_t})) &= \text{cc}\left(\mathcal{W}_{a_t} \xrightarrow[\frac{1}{1}]{\frac{\rho a_t}{2}} \begin{cases} D_\varepsilon(\mathcal{W}_{a_{t+1}}) \\ \mathcal{X}_{a_t, \rho} \xrightarrow[\frac{1}{1}]{\frac{1-\rho}{2}a_t} D_\varepsilon\left(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}\right) \end{cases}\right) \\ &= 1 + \max \left\{ \begin{aligned} &\text{cc}(D_\varepsilon(\mathcal{W}_{a_{t+1}})) \\ &1 + \text{cc}\left(D_\varepsilon\left(\mathcal{R}_{\frac{1-\rho}{2}a_t, \frac{1}{4}a_t}\right)\right) \end{aligned} \right\} \\ &\leq 1 + \max \left\{ \begin{aligned} &\tilde{t} - (t + 1) + 2 \\ &1 + \tilde{t} - t \end{aligned} \right\} && \text{(induction hypothesis)} \\ &= 1 + \max \left\{ \begin{aligned} &\tilde{t} - t + 1 \\ &\tilde{t} - t + 1 \end{aligned} \right\} \\ &\leq \tilde{t} - t + 2 \end{aligned}$$

This finishes the induction and for the curve count of D we conclude:

$$\begin{aligned} \text{cc}(D) &= \text{cc}(D(\mathcal{W}_{a_0})) \\ &\leq \tilde{t} - 0 + 2 \\ &= \lfloor 2\log_2\left(\frac{a}{\varepsilon}\right) \rfloor + 1 + 2 \\ &= \lfloor 2\log_2\left(\frac{a}{\varepsilon}\right) \rfloor + 3 \end{aligned}$$

With that the second property of the tree is proven as well. $\square$

Corollary 6.12. $\text{ctd}(\mathcal{W}_a) \leq \frac{1+\sqrt{57}}{8}a \approx 1.069a$

Proof. For any $\varepsilon > 0$ with $\varepsilon \leq a$, the ε -decomposition tree D of Lemma 6.11 provides an upper bound for the ε -treedepth of $\mathcal{W}_a$:

$$\text{ctd}_\varepsilon(\mathcal{W}_a) \leq h(D) \leq \frac{1 + \sqrt{57}}{8}a$$

Because ε can be chosen arbitrarily small, we conclude:

$$\text{ctd}(\mathcal{W}_a) \leq \frac{1 + \sqrt{57}}{8}a \approx 1.069a$$

□

Corollary 6.13. $\text{td}(W_k) \leq \frac{1+\sqrt{57}}{8}k + \lfloor 2 \log_2(k-1) \rfloor + 3 \approx 1.069k + \mathcal{O}(\log k)$

Proof. Define $a := k - 1$. It holds that $\text{Emb}(W_k) \subseteq \mathcal{R}_{k-1} = \mathcal{W}_a$. By Lemma 6.11 there exists an ε -decomposition tree D of $\mathcal{W}_a$ with height $\frac{1+\sqrt{57}}{8}a$ and curve count $\lfloor 2 \log_2(\frac{a}{\varepsilon}) \rfloor + 3$. Choosing $\varepsilon = 1$, we obtain a 1-decomposition tree of $\mathcal{W}_a$. Because $\text{Emb}(W_k) \subseteq \mathcal{W}_a$, by Theorem 5.11 the following upper bound holds:

$$\begin{aligned} \text{td}(W_k) &\leq h(D) + \text{cc}(D) + 1 \\ &\leq \frac{1 + \sqrt{57}}{8}a + \lfloor 2 \log_2(\frac{a}{\varepsilon}) \rfloor + 3 + 1 \\ &= \frac{1 + \sqrt{57}}{8}(k-1) + \lfloor 2 \log_2(\frac{k-1}{1}) \rfloor + 4 \\ &\leq \frac{1 + \sqrt{57}}{8}k + \lfloor 2 \log_2(k-1) \rfloor + 3 \\ &\approx 1.069k + \mathcal{O}(\log k) \end{aligned}$$

□

6.5 The Triangle

To prove the treedepth of the continuous and discrete triangle we reduce the problem to the ε -decomposition tree of the continuous wedge.

Lemma 6.14. *For the shape $\mathcal{T}_a$ and any $\varepsilon > 0$ with $\varepsilon \leq a$ there exists an ε -decomposition tree D that satisfies the following properties:*

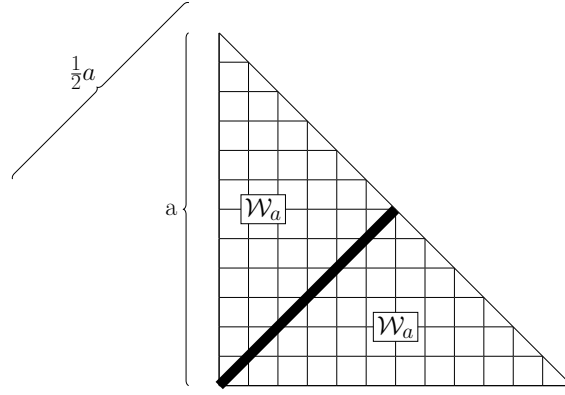
1. $h(D) \leq \frac{5+\sqrt{57}}{8}a \approx 1.569a$
2. $\text{cc}(D) \leq \lfloor 2 \log_2(\frac{a}{\varepsilon}) \rfloor + 4$

Proof. As pictured in Figure 6.3, the triangle $\mathcal{T}_a$ can be cut in half with a diagonal straight line of length $a/2$ such that two isometric copies of the wedge $\mathcal{W}_a$ remain. We derive the following decomposition tree:

$$\mathcal{T}_a \xrightarrow[1]{\frac{1}{2}a} \mathcal{W}_a$$

For an arbitrary $\varepsilon > 0$ with $\varepsilon \leq a$, by Lemma 6.11 there exists the ε -decomposition tree $D_\varepsilon(\mathcal{W}_a)$ of $\mathcal{W}_a$ of height $\frac{1+\sqrt{57}}{8}a$ and curve count $\lfloor 2 \log_2(\frac{a}{\varepsilon}) \rfloor + 3$. This allows us to construct the desired ε -decomposition tree of $\mathcal{T}_a$:

$$D := \mathcal{T}_a \xrightarrow[1]{\frac{1}{2}a} D_\varepsilon(\mathcal{W}_a)$$


 Figure 6.3: Decomposition of the continuous triangle $\mathcal{T}_a$.

Calculate the height of D :

$$\begin{aligned}
 h(D) &= h\left(\mathcal{T}_a \xrightarrow[1]{\frac{1}{2}a} D_\varepsilon(\mathcal{W}_a)\right) \\
 &= \frac{1}{2}a + h(D_\varepsilon(\mathcal{W}_a)) \\
 &\leq \frac{1}{2}a + \frac{1 + \sqrt{57}}{8}a \quad (\text{Lemma 6.11}) \\
 &= \frac{5 + \sqrt{57}}{8}a
 \end{aligned}$$

And finally calculate the curve count of D :

$$\begin{aligned}
 \text{cc}(D) &= \text{cc}\left(\mathcal{T}_a \xrightarrow[1]{\frac{1}{2}a} D_\varepsilon(\mathcal{W}_a)\right) \\
 &= 1 + \text{cc}(D_\varepsilon(\mathcal{W}_a)) \\
 &\leq 1 + \lfloor 2\log_2\left(\frac{a}{\varepsilon}\right) \rfloor + 3 \quad (\text{Lemma 6.11}) \\
 &= \lfloor 2\log_2\left(\frac{a}{\varepsilon}\right) \rfloor + 4
 \end{aligned}$$

With that the two properties of the tree D are proven. $\square$

Corollary 6.15. $\text{ctd}(\mathcal{T}_a) \leq \frac{5+\sqrt{57}}{8}a \approx 1.569a$

Proof. For any $\varepsilon > 0$ with $\varepsilon \leq a$, the ε -decomposition tree D of Lemma 6.14 provides an upper bound for the ε -treedepth of $\mathcal{T}_a$:

$$\text{ctd}_\varepsilon(\mathcal{T}_a) \leq h(D) \leq \frac{5 + \sqrt{57}}{8}a$$

Because ε can be chosen arbitrarily small, we conclude:

$$\text{ctd}(\mathcal{T}_a) \leq \frac{5 + \sqrt{57}}{8}a \approx 1.569a$$

$\square$

Corollary 6.16. $\text{td}(T_k) \leq \frac{5+\sqrt{57}}{8}k + \lfloor 2\log_2(k-1) \rfloor + 4 \approx 1.569k + \mathcal{O}(\log k)$

Proof. Define $a := k - 1$. It holds that $\text{Emb}(T_k) \subseteq \mathcal{T}_{k-1} = \mathcal{T}_a$. By Lemma 6.14 there exists an ε -decomposition tree D of $\mathcal{T}_a$ of height $\frac{5+\sqrt{57}}{8}a$ and curve count $\lfloor 2\log_2(\frac{a}{\varepsilon}) \rfloor + 4$. Choosing $\varepsilon = 1$, we obtain a 1-decomposition tree of $\mathcal{T}_a$. Because $\text{Emb}(T_k) \subseteq \mathcal{T}_a$, by Theorem 5.11 the following upper bound holds:

$$\begin{aligned} \text{td}(T_k) &\leq h(D) + \text{cc}(D) + 1 \\ &\leq \frac{5+\sqrt{57}}{8}a + \lfloor 2\log_2(\frac{a}{\varepsilon}) \rfloor + 4 + 1 \\ &= \frac{5+\sqrt{57}}{8}(k-1) + \lfloor 2\log_2(\frac{k-1}{1}) \rfloor + 5 \\ &\leq \frac{5+\sqrt{57}}{8}k + \lfloor 2\log_2(k-1) \rfloor + 4 \\ &\approx 1.569k + \mathcal{O}(\log k) \end{aligned}$$

□

6.6 The Square

We now prove upper bounds for the square shape and square grid by providing an ε -decomposition tree of the square. This tree builds on the ε -decomposition trees of the continuous triangle and rotated rectangle that are constructed in Lemma 6.14 and 6.8.

Lemma 6.17. *For the shape $\mathcal{S}_a$ and any $\varepsilon > 0$ with $\varepsilon \leq a$ and any $\rho \in [\frac{1}{2}, 1)$ there exists an ε -decomposition tree D_ρ that satisfies the following properties:*

1. $h(D_\rho) \leq (1 + \frac{\rho}{2})a + a \cdot \max \left\{ \frac{5+\sqrt{57}}{8}\rho, \frac{5}{2} - 2\rho \right\}$
2. $\text{cc}(D_\rho) \leq \lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 6$

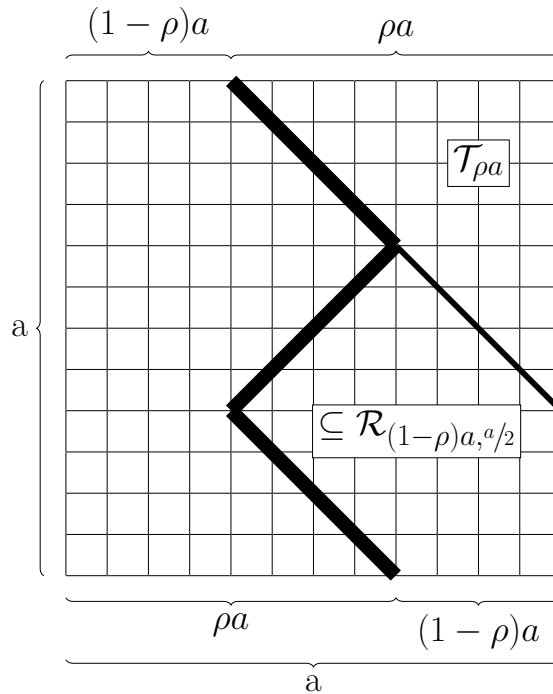


Figure 6.4: Decomposition of the square $\mathcal{S}_a$.

Proof. For any $\rho \in [\frac{1}{2}, 1)$, Figure 6.4 depicts a sequence of two cuts in $\mathcal{S}_a$:

1. The thick black cut consists of three lines: the two outer ones of length $\frac{\rho}{2}a$ and the middle one of length $(1 - \rho)a$. The exact placement depends on ρ but for every ρ the points where the lines meet are on the diagonal from the bottom left to top right corner. The cut divides $\mathcal{S}_a$ into two isometric shapes, that we name $\mathcal{X}_{a,\rho}$.
2. The thinner black cut of length $\frac{\rho}{2}a$ cuts $\mathcal{X}_{a,\rho}$ into $\mathcal{T}_{\rho a}$ and a subset of $\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}$

We derive the following decomposition tree:

$$\mathcal{S}_a \xrightarrow{\frac{a}{1}} \mathcal{X}_{a,\rho} \xrightarrow{\frac{\frac{\rho}{2}a}{1}} \begin{cases} \mathcal{T}_{\rho a} \\ \mathcal{R}_{(1-\rho)a, \frac{1}{2}a} \end{cases}$$

For $\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}$ by Lemma 6.8 there exists the ε -decomposition tree $D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a})$ of height $2(1 - \rho)a + \frac{1}{2}a$ and curve count $2\lfloor \log_2(\frac{a}{2\varepsilon}) \rfloor + 4$.

By Lemma 6.14 there exists the ε -decomposition tree $D_\varepsilon(\mathcal{T}_{\rho a})$ of $\mathcal{T}_{\rho a}$ of height $\frac{5+\sqrt{57}}{8}a$ and curve count $\lfloor 2\log_2(\frac{a}{\varepsilon}) \rfloor + 4$. Note that $(1 - \rho) \leq 1/2$, so the requirement $(1 - \rho)a \leq a/2$ for Lemma 6.8 holds.

We extend the following decomposition tree to the following:

$$D_\rho := \mathcal{S}_a \xrightarrow{\frac{a}{1}} \mathcal{X}_{a,\rho} \xrightarrow{\frac{\frac{\rho}{2}a}{1}} \begin{cases} D_\varepsilon(\mathcal{T}_{\rho a}) \\ D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}) \end{cases}$$

This is an ε -decomposition tree of $\mathcal{S}_a$ because the two trees it is based on are ε -decomposition trees.

We now calculate the height of the tree D :

$$\begin{aligned} h(D_\rho) &= h\left(\mathcal{S}_a \xrightarrow{\frac{a}{1}} \mathcal{X}_{a,\rho} \xrightarrow{\frac{\frac{\rho}{2}a}{1}} \begin{cases} D_\varepsilon(\mathcal{T}_{\rho a}) \\ D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}) \end{cases}\right) \\ &= a + \frac{\rho}{2}a + \max\left\{h(D_\varepsilon(\mathcal{T}_{\rho a})), h(D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}))\right\} \\ &\leq (1 + \frac{\rho}{2})a + \max\left\{\frac{5+\sqrt{57}}{8}\rho a, 2(1 - \rho)a + \frac{1}{2}a\right\} \\ &= (1 + \frac{\rho}{2})a + a \cdot \max\left\{\frac{5+\sqrt{57}}{8}\rho, \frac{5}{2} - 2\rho\right\} \end{aligned}$$

And finally, we calculate the curve count of D :

$$\begin{aligned} \text{cc}(D_\rho) &= \text{cc}\left(\mathcal{S}_a \xrightarrow{\frac{a}{1}} \mathcal{X}_{a,\rho} \xrightarrow{\frac{\frac{\rho}{2}a}{1}} \begin{cases} D_\varepsilon(\mathcal{T}_{\rho a}) \\ D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}) \end{cases}\right) \\ &= 2 + \max\left\{\text{cc}(D_\varepsilon(\mathcal{T}_{\rho a})), \text{cc}(D_\varepsilon(\mathcal{R}_{(1-\rho)a, \frac{1}{2}a}))\right\} \\ &\leq 2 + \max\left\{\lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 4, \lfloor 2\log_2(\frac{a}{2\varepsilon}) \rfloor + 4\right\} \\ &= 2 + \max\left\{\lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 4, \lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 4\right\} \\ &= \lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 6 \end{aligned}$$

With that the two properties of D_ρ are proven. $\square$

To derive specific upper bounds using the ε -decomposition tree D_ρ constructed in the above Lemma 6.17, we choose a specific value for ρ . We aim for the height of D_ρ too be as small as possible, to get the best linear factor on the upper bound. The height is:

$$h(D_\rho) \leq (1 + \frac{\rho}{2})a + a \cdot \max \left\{ \frac{5+\sqrt{57}}{8}\rho, \frac{5}{2} - 2\rho \right\}$$

Because $1 + \frac{\rho}{2}$, $\frac{5+\sqrt{57}}{8}\rho$ and $\frac{5}{2} - 2\rho$ are all linear in ρ , we can determine the optimal value for ρ by finding the intersection of the two latter linear terms:

$$\begin{aligned} & \frac{5 + \sqrt{57}}{8}\rho = \frac{5}{2} - 2\rho \\ \iff & \frac{21 + \sqrt{57}}{8}\rho = \frac{5}{2} \\ \iff & \rho = \frac{20}{21 + \sqrt{57}} \end{aligned}$$

So $\rho = \frac{20}{21+\sqrt{57}}$ is the best choice. We conclude upper bounds for the square and the square grid.

Corollary 6.18.

$$\text{ctd}(\mathcal{S}_a) \leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}a \approx 2.450a$$

Proof. For $\rho := \frac{20}{21+\sqrt{57}}$ and any $\varepsilon > 0$ with $\varepsilon \leq a$, take the decomposition tree D_ρ of $\mathcal{S}_a$ provided by Lemma 6.17. Its height is:

$$h(D_\rho) \leq (1 + \frac{\rho}{2})a + a \cdot \max \left\{ \frac{5+\sqrt{57}}{8}\rho, \frac{5}{2} - 2\rho \right\}$$

By the choice of ρ we know that $\frac{5+\sqrt{57}}{8}\rho = \frac{5}{2} - 2\rho$. We explicitly calculate the height:

$$\begin{aligned} h(D_\rho) & \leq (1 + \frac{\rho}{2})a + (\frac{5}{2} - 2\rho)a \\ & = (\frac{7}{2} - \frac{3}{2}\rho)a \\ & = (\frac{7}{2} - \frac{3}{2} \frac{20}{21 + \sqrt{57}})a \\ & = (\frac{7(21 + \sqrt{57})}{2(21 + \sqrt{57})} - \frac{60}{2(21 + \sqrt{57})})a \\ & = \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}a \end{aligned}$$

This gives us an upper bound on the ε -treedepth of $\mathcal{S}_a$:

$$\text{ctd}_\varepsilon(\mathcal{S}_a) \leq h(D_\rho) \leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}a$$

Because ε can be chosen arbitrarily small, we conclude:

$$\text{ctd}(\mathcal{S}_a) \leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}a \approx 2.450a$$

$\square$

Corollary 6.19.

$$\begin{aligned} \text{td}(G_k) &\leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}k + \lfloor 2\log_2(k-1) \rfloor + 5 \\ &\approx 2.450k + \mathcal{O}(\log k) \end{aligned}$$

Proof. Define $a := k - 1$. It holds that $\text{Emb}(G_k) \subseteq \mathcal{S}_{k-1} = \mathcal{S}_a$. By Lemma 6.17 there exists an ε -decomposition tree D of $\mathcal{S}_a$ of curve count $\lfloor 2\log_2(\frac{\rho a}{\varepsilon}) \rfloor + 6$ and the following height:

$$h(D_\rho) \leq (1 + \frac{\rho}{2})a + a \cdot \max \left\{ \frac{5+\sqrt{57}}{8}\rho, \frac{5}{2} - 2\rho \right\}$$

For $\rho := \frac{20}{21+\sqrt{57}}$ we have already explicitly calculated the height of the tree D_ρ in Corollary 6.18:

$$h(D_\rho) = \frac{87+7\sqrt{57}}{42+2\sqrt{57}}a$$

Choosing $\varepsilon = 1$, we obtain a 1-decomposition tree of $\mathcal{S}_a$. Because $\text{Emb}(G_k) \subseteq \mathcal{S}_a$, by Theorem 5.11 the following upper bound holds:

$$\begin{aligned} \text{td}(G_k) &\leq h(D_\rho) + \text{cc}(D_\rho) + 1 \\ &\leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}a + \lfloor 2\log_2(a) \rfloor + 6 + 1 \\ &= \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}(k-1) + \lfloor 2\log_2(k-1) \rfloor + 7 \\ &\leq \frac{87 + 7\sqrt{57}}{42 + 2\sqrt{57}}k + \lfloor 2\log_2(k-1) \rfloor + 5 \\ &\approx 2.450k + \mathcal{O}(\log k) \end{aligned}$$

□

For all upper bounds on the treedepth of the grid graphs, the curve count of the continuous decomposition tree has introduced a term of logarithmic order compared to the upper bound for the shape. This is because we only used cuts that consist of one curve and we made the size of the shapes shrink exponentially. It might seem like this must be the case for all decomposition trees of shapes. But it is still possible to construct a decomposition tree that uses far more cuts than necessary and therefore drives up the curve count. This brings up the question if for any given ε -decomposition tree there exists an ε -decomposition tree of at most the same height but with a curve count bounded within logarithmic order.

It is also possible that a cut has to consist of multiple curves. For example when considering a ring shape, a short cut would have to consist of two separate lines to disconnect the ring.

6.7 Discrete Approach for the Rotated Rectangle

We now take a closer look at the discrete rotated rectangle. The decomposition strategy is similar to the one we used in the continuous setting. We derive the same linear term but prove that the logarithmic term can be dropped.

Lemma 6.20. *For all $m, n \in \mathbb{N} \setminus \{1\}$ with $m \leq n$, it holds that*

$$\text{td}(R_{m,n}) \leq m - 1 + \text{td}(R_{m, \lceil n/2 \rceil})$$

Proof. The rotated rectangle graph with side lengths m and n can be separated into $R_{m, \lfloor n/2 \rfloor}$ and $R_{m, \lceil n/2 \rceil}$ by removing $m - 1$ vertices diagonally in the middle. An example for this partition is given in Figure 6.5 for $a = 7$ and $b = 9$.

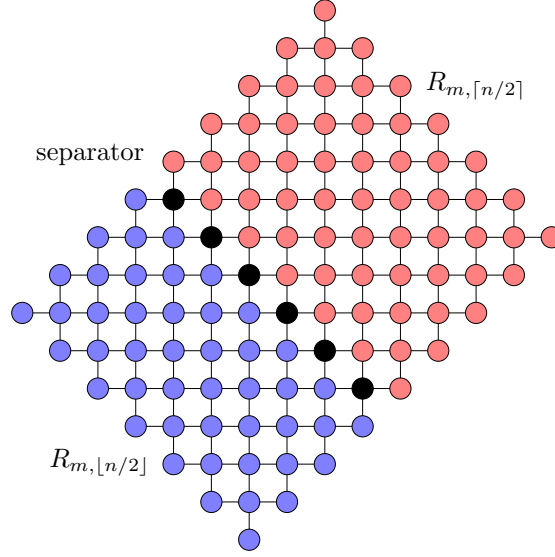


Figure 6.5: Separation of $R_{m,n}$ for $m = 7$ and $n = 9$.

We know that $R_{m, \lfloor n/2 \rfloor} \subseteq R_{m, \lceil n/2 \rceil}$, so using our borrowed Theorem 3.1 we can conclude the recursive formula:

$$\begin{aligned} \text{td}(R_{m,n}) &\leq m - 1 + \max \begin{cases} \text{td}(R_{m, \lceil n/2 \rceil}) \\ \text{td}(R_{m, \lfloor n/2 \rfloor}) \end{cases} \\ &= m - 1 + \text{td}(R_{m, \lceil n/2 \rceil}) \end{aligned}$$

□

Lemma 6.21. *For all $m, n \in \mathbb{N}$ with $m \leq n \leq 2m$, the following upper bound holds:*

$$\text{td}(R_{m,n}) \leq p(m, n)$$

$$\text{where } p(m, n) := 2m + n - 2 - \text{bc}(m - 1) - \text{bc}(n - 1)$$

Here, bc is the bit count operation, that outputs the count of ones in the binary representation of a positive integer.

Proof. We use strong induction on $s(m, n) := m + n$. Let $m, n \in \mathbb{N}$ with $m \leq n \leq 2m$.

Base case ($s \in \{2, 3\}$): For $s = 2$, we are looking at $R_{1,1}$ and for $s = 3$ the only option is $R_{1,2}$, because if $m \geq 2$ then $n \geq m \geq 2$, so $s(m, n) \geq 4$. Both, $R_{1,1}$ and $R_{1,2}$ have tree-depth 1 because they are empty graphs and the formula holds:

$$\begin{aligned} p(1, 1) &= 2 \cdot 1 + 1 - 2 - \text{bc}(1 - 1) - \text{bc}(1 - 1) \\ &= 1 - \text{bc}(0) - \text{bc}(0) \\ &= 1 - 0 - 0 \\ &= \text{td}(R_{1,1}) \end{aligned}$$

$$\begin{aligned}
p(1, 2) &= 2 \cdot 1 + 2 - 2 - \text{bc}(1 - 1) - \text{bc}(2 - 1) \\
&= 2 - \text{bc}(0) - \text{bc}(1) \\
&= 2 - 0 - 1 \\
&= 1 \\
&= \text{td}(R_{1,2})
\end{aligned}$$

Inductive step ($s \geq 4$): Assume that for all $\tilde{m}, \tilde{n} \in \mathbb{N}$ with $\tilde{m} \leq \tilde{n} \leq 2\tilde{m}$ and $s(\tilde{m}, \tilde{n}) < s(m, n)$ the upper bound $\text{td}(R_{\tilde{m}, \tilde{n}}) \leq p(\tilde{m}, \tilde{n})$ holds.

To use the recursive formula of Lemma 6.20, we need to confirm that m is at least 2. This follows from $s \geq 4$, because if m was 1, then n would have to be $4 - 1 = 3$ but then $n \leq 2m$ would not hold. We use the recursive formula and define $m' := \lceil n/2 \rceil$ and $n' := m$ to get:

$$\text{td}(R_{m,n}) \leq m - 1 + \text{td}(R_{m, \lceil n/2 \rceil}) = m - 1 + \text{td}(R_{n', m'}) = m - 1 + \text{td}(R_{m', n'})$$

To call induction on $R_{m', n'}$ we need to prove two things:

First, we need to confirm that the precondition $m' \leq n' \leq 2m'$ still holds:

$$m' = \lceil n/2 \rceil \leq \lceil 2m/2 \rceil = m = n'$$

$$n' = m \leq n = 2 \cdot n/2 \leq 2\lceil n/2 \rceil = 2m'$$

Second, we need to confirm that $s(m', n')$ is strictly less than $s(m, n)$. We know $m \geq 2$ holds, so $n \geq m \geq 2$ does too. This means that $\lceil n/2 \rceil$ is strictly less than n . It follows:

$$s(m', n') = m' + n' = \lceil n/2 \rceil + m < n + m = s(m, n)$$

Now we can call induction to get:

$$\begin{aligned}
\text{td}(R_{m,n}) &\leq m - 1 + \text{td}(R_{m', n'}) \\
&\leq m - 1 + 2m' + n' - 2 - \text{bc}(m' - 1) - \text{bc}(n' - 1)
\end{aligned}$$

Case 1: n is odd.

$$\begin{aligned}
&\text{td}(R_{m,n}) \\
&\leq m - 1 + 2m' + n' - 2 - \text{bc}(m' - 1) - \text{bc}(n' - 1) \\
&= m - 1 + 2\lceil n/2 \rceil + m - 2 - \text{bc}(\lceil n/2 \rceil - 1) - \text{bc}(m - 1) \\
&= 2m + 2\frac{n+1}{2} - 3 - \text{bc}(\frac{n+1}{2} - 1) - \text{bc}(m - 1) && n \text{ is odd} \\
&= 2m + n + 1 - 3 - \text{bc}(\frac{n-1}{2}) - \text{bc}(m - 1) \\
&= 2m + n - 2 - \text{bc}(n - 1) - \text{bc}(m - 1) && \cdot 2 \text{ does not change bc} \\
&= p(m, n)
\end{aligned}$$

Case 2: b is even.

$$\begin{aligned}
 \text{td}(R_{m,n}) &\leq m - 1 + 2m' + n' - 2 - \text{bc}(m' - 1) - \text{bc}(n' - 1) \\
 &= m - 1 + 2\lceil n/2 \rceil + m - 2 - \text{bc}(\lceil n/2 \rceil - 1) - \text{bc}(m - 1) \\
 &= 2m + 2\frac{n}{2} - 3 - \text{bc}(\frac{n}{2} - 1) - \text{bc}(m - 1) && n \text{ is even} \\
 &= 2m + n - 3 - \text{bc}(\frac{n-2}{2}) - \text{bc}(m - 1) \\
 &= 2m + n - 3 - \text{bc}(n - 2) - \text{bc}(m - 1) && \cdot 2 \text{ does not change bc} \\
 &= 2m + n - 3 - (\text{bc}(n - 2 + 1) - 1) - \text{bc}(m - 1) && \text{flipping LSB from 0 to 1} \\
 &= 2m + n - 2 - \text{bc}(n - 1) - \text{bc}(m - 1) \\
 &= p(m, n)
 \end{aligned}$$

This completes the induction and finishes the proof.

□

Note that in the proof, except for the inductive call, all transformations are equalities. This means that the explicit expression is optimal for the recursive formula in Lemma 6.20.

This upper bound looks familiar to the one stated in Corollary 6.10 that was proven using the continuous decomposition tree. The linear term $2m + n$ is the same but the extra term $2\lfloor \log_2(n - 1) \rfloor + 1$ is dropped and the new bound is slightly below the linear expression because of the term $-2 - \text{bc}(n - 1) - \text{bc}(m - 1)$. This indicates that the treedepth of a grid graph might even be lower than the continuous treedepth of the corresponding shape.

In the new bound the terms $-\text{bc}(n - 1)$ and $-\text{bc}(m - 1)$ appear. They are a measure of how well the numbers m and n can be repeatedly divided by 2 without having to round. For example if $m = 2^k$ for some $k \in \mathbb{N}$, then $\text{bc}(m - 1) = \log_2 m$. In this case repeatedly halving the rotated rectangle along the m -side would never result in having to round up which is represented by a high bit count of m being subtracted from the bound. It also holds that $\text{bc}(2^k) = 1$ for all $k \in \mathbb{N}_0$ which suggests that the $2m + n$ bound is quite tight even for large k .

7. Conclusion

This research was driven mainly by the objective to improve upper bounds on the treedepth of grid graphs. The old upper bound of the k -by- k grid of $2.519k \pm o(k)$ is now improved to $2.45k + O(\log k)$. In the process, we also improved the upper bounds of other grid graphs such as wedges and triangles. We studied the rotated rectangle and used the results for the upper bounds of all other grid graphs. For the old upper bound, this grid graph was studied under the name *rhombus* and only considering equal side lengths [BNCL⁺12].

In pursuit of better upper bounds, we introduced the continuous treedepth problem and showed a connection to the treedepth of grid graphs. This tool simplifies the search for better upper bounds because describing strategies directly on the grid graphs is more complicated. The selection of the separators depends on the divisibility of the graph dimensions, and additionally, floors and ceilings complicate the recursion. On the other hand, the tool adds the curve count when transforming a continuous to a discrete decomposition tree, which creates a gap between the upper bound of the shape and the upper bound of the grid graph.

For the upper bounds proven in this thesis, the curve count always introduced an extra logarithmic term. It seems like this could be generalized and leads to the open question:

Question 7.1. *For any parameterized grid graph H_n and a shape S_n such that $\text{Emb}(H_n) \subseteq S_n$, does $\text{td}(G_n) \leq \text{ctd}_1(S_n) + O(\log n)$ hold?*

In general, it is an interesting open question how close the continuous treedepth and the discrete treedepth can be brought together. We know that $\text{td}(G_k)$ is bounded between linear terms in k and believe that there is a tight connection to $\text{ctd}(\mathcal{S}_k)$. This brings up the following question:

Question 7.2. *Does the limit $\lim_{k \rightarrow \infty} \text{td}(G_k)/k$ exist, and is it equal to $\text{ctd}(\mathcal{S}_1)$?*

It is likely that the upper bound $2.45k + O(\log k)$ of $\text{td}(G_k)$ can be improved significantly further. Looking at the treedepth of G_k for specific small k , it seems like the treedepth is about $2k$. For the other studied grid graphs, especially for the rotated rectangle, we believe that we are close to the actual bound. But the question about tighter upper and lower bounds for these graphs still remains:

Question 7.3. *Do the following statements hold?*

- $\text{td}(R_{m,n}) \in 2m + n + O(1)$
- $\text{td}(W_k) \in k + O(\log k)$
- $\text{td}(T_k) \in 1.5 k + O(\log k)$

Another question came up when calculating the treedepth of G_k for specific small k . If one colors the vertices of the grid in a checkerboard pattern - essentially a two-coloring of the bipartite graph - it seems like one can always find an optimal elimination tree with the leaves being all vertices of one color class. That means the separators only use the vertices of the other color class. We formalize this question:

Question 7.4. *For G_k , let $A \cup B = V(G_k)$ be the bipartition. Does it hold that there exists an elimination tree T of G_k of height $\text{td}(G_k)$ with the leaves being exactly the vertices in A or exactly the vertices in B ?*

An idea to prove this would be to show that for every elimination tree of G_k there exists an elimination tree of at most the same height with all leaves being the vertices in the larger color class. If this is true, finding the treedepth of the k -by- k -grid G_k would reduce to finding the treedepth of the graph H that results when one color class A is deleted and an edge is inserted between every two vertices in the leftover class B that were connected via a path with inner vertices only in A . The resulting graph H can be pictured as the strong grid rotated by 45° as shown in Figure 7.1. It always holds that $\text{td}(G_k) \leq \text{td}(H) + 1$ but if the statement in the posed question is true, this equation becomes an equality.

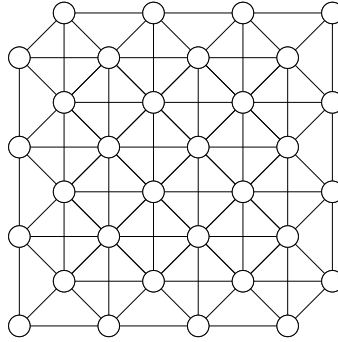


Figure 7.1: Strong grid resulting from deletion operation on one color class.

We also want to motivate the research on the continuous treedepth itself. Stating the problem is very intuitive but solutions are often not trivial. Also, the choice of the norm can simplify or complicate the problem. Similar to the question for different grid graphs we pose:

Question 7.5. *For a norm $\|\cdot\|$ and a shape S , what is $\text{ctd}_{\|\cdot\|}(S)$? Specifically, do the following statements hold, considering the Chebyshev and Euclidean norm?*

- $\text{ctd}_{\|\cdot\|_\infty}(\mathcal{R}_{a,b}) = 2a + b$
- $\text{ctd}_{\|\cdot\|_\infty}(\mathcal{W}_a) = a$
- $\text{ctd}_{\|\cdot\|_\infty}(\mathcal{T}_a) = 1.5 a$
- $\text{ctd}_{\|\cdot\|_2}(\mathcal{S}_a) = 3a$

A possible generalization of the continuous treedepth problem is defining it in higher dimensions. For example, in $\mathbb{R}^3$ the problem would translate to the decomposition of bodies using planes. Similar to the connection we built between the continuous treedepth of shapes and the treedepth of grid graphs, the higher dimensional problem could be connected to the treedepth of higher dimensional grid graphs.

Bibliography

- [Alp10] Hannah Alpert. Rank numbers of grid graphs. *Discrete Mathematics*, 310(23):3324–3333, 2010.
- [BDJ⁺98] Hans Bodlaender, Jitender Deogun, Klaus Jansen, Ton Kloks, Dieter Kratsch, Haiko Müller, and Zsolt Tuza. Rankings of graphs. *SIAM Journal on Discrete Mathematics*, 11:168–181, 02 1998.
- [BGHK95] H.L. Bodlaender, J.R. Gilbert, H. Hafsteinsson, and T. Kloks. Approximating treewidth, pathwidth, frontsize, and shortest elimination tree. *Journal of Algorithms*, 18(2):238–255, 1995.
- [BNCL⁺12] Amotz Bar-Noy, Panagiotis Cheilaris, Michael Lampis, Valia Mitsou, and Stathis Zachos. Ordered coloring of grids and related graphs. *Theoretical Computer Science*, 444:40–51, 2012. Structural Information and Communication Complexity – SIROCCO 2009.
- [CC14] Chandra Chekuri and Julia Chuzhoy. Polynomial bounds for the grid-minor theorem. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC ’14, page 60–69, New York, NY, USA, 2014. Association for Computing Machinery.
- [Che13] Sitan Chen. On the rank number of grid graphs. <https://arxiv.org/abs/1208.1814>, 2013.
- [DKKM93] J.S. Deogun, A.J.J. Kloks, D. Kratsch, and H. Müller. *On vertex ranking for permutation and other graphs*. Computing science notes. Technische Universiteit Eindhoven, 1993.
- [GLP96] J. Ghoshal, R. Laskar, and D. Pillone. Minimal rankings. *Networks*, 28(1):45–53, 1996.
- [GLP99] J. Ghoshal, Renu C. Laskar, and Dan Pillone. Further results on minimal rankings. *Ars Comb.*, 52, 1999.
- [IOO18] Yoichi Iwata, Tomoaki Ogasawara, and Naoto Ohsaka. On the Power of Tree-Depth for Fully Polynomial FPT Algorithms. In Rolf Niedermeier and Brigitte Vallée, editors, *35th Symposium on Theoretical Aspects of Computer Science (STACS 2018)*, volume 96 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 41:1–41:14, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [KR18] Ken-ichi Kawarabayashi and Benjamin Rossman. *A Polynomial Excluded-Minor Approximation of Treedepth*, pages 234–246. SIAM, 01 2018.
- [NdM12] Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity: Graphs, Structures, and Algorithms*. Springer Publishing Company, Incorporated, 2012.

- [NON09] Sarah Novotny, Juan Ortiz, and Darren A. Narayan. Minimal k-rankings and the rank number of pn_2 . *Information Processing Letters*, 109(3):193–198, 2009.
- [NPSW23] Jesper Nederlof, Michał Pilipczuk, Celine M. F. Swennenhuis, and Karol Węgrzycki. Hamiltonian cycle parameterized by treedepth in single exponential time and polynomial space. *SIAM Journal on Discrete Mathematics*, 37(3):1566–1586, 2023.
- [Pot88] Alex Pothén. The complexity of optimal elimination trees. Technical Report CS 88-16, Pennsylvania State University - Department of Computer Science, 1988.