Computational Metric Embeddings

by

Anastasios Sidiropoulos

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2008

Author ....
Department of Electrical Engineering and Computer Science
May 18, 2008

Certified by ....
Piotr Indyk
Associate Professor
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Computational Metric Embeddings

by

Anastasios Sidiropoulos

Submitted to the Department of Electrical Engineering and Computer Science
on May 18, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

We study the problem of computing a low-distortion embedding between two metric spaces. More precisely given an input metric space M we are interested in computing in polynomial time an embedding into a host space M' with minimum multiplicative distortion. This problem arises naturally in many applications, including geometric optimization, visualization, multi-dimensional scaling, network spanners, and the computation of phylogenetic trees. We focus on the case where the host space is either a euclidean space of constant dimension such as the line and the plane, or a graph metric of simple topological structure such as a tree.

For Euclidean spaces, we present the following upper bounds. We give an approximation algorithm that, given a metric space that embeds into $\mathbb{R}^1$ with distortion c, computes an embedding with distortion $c^{O(1)}\Delta^{3/4}$ (Δ denotes the ratio of the maximum over the minimum distance). For higher-dimensional spaces, we obtain an algorithm which, for any fixed $d \geq 2$, given an ultrametric that embeds into $\mathbb{R}^d$ with distortion c, computes an embedding with distortion $c^{O(1)}$. We also present an algorithm achieving distortion $c \log^{O(1)} \Delta$ for the same problem.

We complement the above upper bounds by proving hardness of computing optimal, or near-optimal embeddings. When the input space is an ultrametric, we show that it is NP-hard to compute an optimal embedding into $\mathbb{R}^2$ under the ℓ_∞ norm. Moreover, we prove that for any fixed $d \geq 2$, it is NP-hard to approximate the minimum distortion embedding of an n-point metric space into $\mathbb{R}^d$ within a factor of $\Omega(n^{1/(17d)})$.

Finally, we consider the problem of embedding into tree metrics. We give a $O(1)$-approximation algorithm for the case where the input is the shortest-path metric of an unweighted graph. For general metric spaces, we present an algorithm which, given an n-point metic that embeds into a tree with distortion c, computes an embedding with distortion $(c \log n)^{O(\sqrt{\log \Delta})}$. By composing this algorithm with an algorithm for embedding trees into $\mathbb{R}^1$, we obtain an improved algorithm for embedding general metric spaces into $\mathbb{R}^1$.

Thesis Supervisor: Piotr Indyk

Title: Associate Professor

Acknowledgments

I am grateful to my research supervisor Piotr Indyk for the privilege of working with him.

I would like to thank Ioannis Caragiannis, Stavros Cosmadakis, Efstratios Gallopoulos, Christos Kaklamanis, for their help and influence during my undergraduate studies in the University of Patras.

I also thank my co-authors, Noga Alon, Mihai Badoiu, George Christodoulou, Julia Chuzhoy, Erik Demaine, Martin Farach-Colton, Jon Feldman, Piotr Indyk, Christiane Lammersen, Jiří Matoušek, Vahab Mirrokni, S. Muthukrishnan, Krzysztof Onak, Yuri Rabinovich, Christian Sohler, Cliff Stein, Zoya Svitkina, and Mohammad Taghi-Hajiaghayi.

I thank the family of Paris Kanelakis, and the Onassis Public Benefit Foundation, for supporting my research with fellowships during the first two years of my graduate studies.

I want to thank Jiří Matoušek for his hospitality while visiting Charles University, and for the inspiring discussions I had with him.

I also wish to thank Roberto Tamassia for providing pointers to the applicability in visualization of some of the results discussed in this thesis.

Finally, I want to thank my friends Nikos, Theofilos, Thodoris, Giannis, Periklis, Dimitra, Giannis, Giannis, Jean, Manolis, Christos, Michelle, Joyce, and Dimitris, and my mother Litsa, my grandmother Katerina, and my sister Maria, for all their love. This thesis is dedicated to them.

Anastasios Sidiropoulos

Spring 2008

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Embedding distance matrices into geometric spaces is a fundamental problem occurring in many applications. Intuitively, an embedding is a mapping between two metric spaces that preserves the geometry. More precisely, a metric embedding of a metric space $M = (X, D)$ into a host space $M' = (X', D')$ is a mapping $f : X \to X'$. The *distortion* of such an embedding f is defined as the minimum c, such that there exists $r > 0$, with

$$D(x, y) \leq r \cdot D'(f(x), f(y)) \leq c \cdot D(x, y)$$

The distortion is a parameter that quantifies the extend to which an embedding preserves the geometry of the original space. Note that for example, distortion $c = 1$ implies that the mapping is an isometry. Another useful property of the above definition is that the the distortion is an invariant under scaling of the distances. Moreover, the distortion of the concatenation of two embeddings is equal to the product of their respective distortions. As we shall see later in this chapter, this definition turns out to be particularly important in algorithmic applications.

The main focus of this thesis is computing low-distortion metric embeddings between interesting spaces.

1.1 Absolute and relative metric embeddings

Given a family A of metric spaces, and a host space M', a natural question to ask is what is the minimum c, such that each metric $M \in A$ can be embedded into M' with distortion at most c. We call an embedding with such a distortion guarantee an *absolute* embedding.

Typical examples of absolute embeddings include embedding all n-point metric spaces into a euclidean space, and embedding n-point subsets of a euclidean space into a space of smaller dimension (the so-called dimensionality reduction). Note that we are usually interested in embeddings that can be efficiently computed, although the above definition does not directly impose such a requirement.

Another natural problem concerning A and M' is given a space $M \in A$, find an embedding f of M into M' with the smallest possible distortion. Note that typically one can find an optimal, or near-optimal f just by a careful exhaustive enumeration, so the problem becomes interesting when we want an embedding that can be computed efficiently, and in particular in polynomial time. We call such an embedding a *relative* embedding.

Relative embeddings are important in the case where the worst-case distortion for embedding a metric $M \in A$ into M' is very high, yet some interesting metrics in M can be embedded with small distortion. For example, when the host space M' is a euclidean space of constant dimension such as the plane, the worst-case distortion for embedding all n-point metric spaces into M' is polynomial. However, in some applications, the interesting spaces are precisely those that can be embedded with small distortion.

1.2 Applications of relative embeddings

Is this section we discuss some of the most important applications of relative embeddings.

Geometric optimization Assume that there exists a α-approximation algorithm for a metric optimization problem restricted on a simple space, say TSP on the plane (there are numerous other such examples, e.g. k-Server on the line, Geometric MST, etc). A natural question to ask is the following:

What happens if the input space is "almost" a plane metric?

It is not immediate whether the problem now becomes intractable. The answer of course depends on the definition of closeness to a metric space. In some cases the input data might be distorted simply by adding some Gaussian noise on the numerical values. In certain geometric cases however, the distortion has a more global structure. Examples include an image that passes through a wide-angle lens, the surface of an elastic body under pressure, or a 3-dimensional terrain projected into a 2-dimensional map.

The answer to the above question turns out to be related to the approximability of relative embeddings. More precisely, assume that you have a β-approximation algorithm for the problem of relative embedding into the plane. Combined with a α-approximation for TSP on the plane, this implies a $\alpha \cdot \beta \cdot \gamma$-approximation for TSP on metrics that are γ-embeddable into the plane.

Visualization Visualizing a distance matrix typically involves mapping a set of points into a space of dimension at most 3, with small distortion. It is easy to show that there are n-point metric spaces, e.g. the uniform n-point metric, that require distortion $\Omega(n^{1/3})$ to be embedded into $\mathbb{R}^3$ (a better bound of $\Omega(n^{1/2})$ can be obtained for a more carefully constructed space [43]).

This polynomial distortion is considered to be prohibitively large for most visualization scenarios. Thus, one can ask for efficient algorithms that output an embedding with small distortion, when such an embedding exists.

Multi-dimensional scaling The main goal of the area of Multi-Dimensional Scaling is computing geometric structures that capture some given distance information. These geometric structures usually have to be simple enough, to provide the user with a meaningful interpretation. As in the case of visualization, the worst-case distortion for embedding into such spaces is very high, so one wants to obtain relative embeddings.

Evolutionary biology Phylogenetic trees are used in evolutionary biology to represent genetic distances between various species. Constructing such a tree is precisely a problem of computing a relative embedding of a given metric space into a tree metric (i.e. the shortest-path distance of a tree).

Graph spanners Given a graph $G = (V, E)$, an α-spanner of G is an edge-subgraph $H = (V, E')$ of G such that for any pair of vertices $u, v \in V(G)$, the shortest-path distance between u and v in H is at most α times the distance in G. The parameter α is called the *dilation* of H. In applications such as network routing the most interesting spanners are those with simple topology, e.g. trees. The problem of computing a tree spanner of minimum dilation for a weighted complete graph is equivalent to the problem of the problem of relative embedding into tree metrics. Specifically, Eppstein ([26], Open Problem 4) posed a question about algorithmic complexity of finding the *minimum-dilation spanning tree* of a given set of points in the plane. This problem is equivalent (up to a constant factor in the approximation factor) to a special case of our problem, where the input metric is induced by points in the plane.

1.3 Our contribution

1.3.1 Near-optimality of random projection for embedding into $\mathbb{R}^d$

Bourgain [15] has shown that any n-point metric space can be embedded into Euclidean space with distortion $O(\log n)$. Moreover, Johnson and Lindenstrauss [35] have shown that any Euclidean metric can be embedded into a $O(\varepsilon^{-2} \log n)$-dimensional space with distortion $1+\varepsilon$. The later result is obtained via a so-called random projection. That is, by projecting the given space into a randomly chosen, low-dimensional subspace.

The above strikingly simple procedure is arguably one of the most important algorithms for embeddings, and dimensionality-reduction in general. Matoušek [43] has shown that for the case of embedding into d-dimensional Euclidean space, random projection results in distortion $\tilde{O}(n^{2/d})$, for any fixed $d \geq 1$. He also showed that there exist metrics that require distortion $\Omega(n^{1/\lfloor (d+1)/2 \rfloor})$, establishing that random projection is almost optimal in the worst case.

Despite this worst-case optimality, it is easy to construct metric spaces that embed with very small distortion into $\mathbb{R}^d$, and at the same time *any* projection (in fact, any linear mapping) into $\mathbb{R}^d$ incurs high distortion. In other words, random projection is a $\tilde{\Theta}(n^{2/d})$-approximation algorithm for the problem of embedding an n-point metric into $\mathbb{R}^d$, with minimum distortion. This observation naturally leads to the following question.

> Is there a polynomial-time algorithm for embedding into d-dimensional space, with minimum distortion, that has approximation ratio better than $\Omega(n^{2/d})$?

We address this question by showing that unless P = NP, for any $d \geq 2$, there is no polynomial-time algorithm for this problem, with approximation ratio better than $\Omega(n^{1/(17d)})$. Our result implies that random projection is a near-optimal approximation algorithm for this problem. Note that since for fixed d all norms on $\mathbb{R}^d$ are equivalent up to a constant factor, the same result holds for all norms.

1.3.2 Beyond linear approximation

In light of the above $\Omega(n^{1/(17d)})$-hardness result for embedding into $\mathbb{R}^d$, it is clear that one cannot hope for approximation algorithms for embedding into constant-dimensional spaces, with poly-logarithmic approximation factors. On the other hand, it is still interesting to obtain algorithms that compute embeddings with distortion polynomially related to the optimal. A result of this type has been obtained in

[20] where it is shown that there exists a polynomial-time algorithm that given the shortest-path metric of an unweighted graph that c-embeds into $\mathbb{R}^1$, computes an embedding with distortion $O(c^2)$.

We obtain an approximation algorithm that given a metric space that c-embeds into $\mathbb{R}^1$, computes an embedding with distortion $c^{O(1)}\Delta^{3/4}$, where Δ denotes the ratio of the maximum over the minimum distance. We also present an algorithm for embedding general metrics into $\mathbb{R}^1$ with improved distortion guarantee in an interesting range of the parameters. More specifically, we compose a $(c \log n)^{O(\sqrt{\log \Delta})}$-distortion algorithm for embedding general metrics into trees, a $c^{O(1)}$-distortion algorithm for embedding trees into $\mathbb{R}^1$, to obtain a $(c \log n)^{O(\sqrt{\log \Delta})}$-distortion algorithm for embedding general metrics into $\mathbb{R}^1$.

For higher-dimensional spaces, we present an algorithm which for any fixed $d \geq 2$, given an ultrametric that embeds into $\mathbb{R}^d$ with distortion c, computes an embedding with distortion $c^{O(1)}$.

1.3.3 Special classes of spaces

Our hardness result for embedding into $\mathbb{R}^d$ leaves open the existence of improved approximation guarantees for embedding special classes of metrics. For the case where the input space is an ultrametric that c-embeds into $\mathbb{R}^d$, we give an algorithm that computes an embedding with distortion $c \log^{O(1)} \Delta$. This is the first algorithm for embedding such a class of graph-theoretic spaces into a space of constant dimension, with poly-logarithmic approximation ratio. We complement this bound by showing that it is NP-hard to compute an optimal embedding of an ultrametric into $\mathbb{R}^2$, under the ℓ_∞ norm.

1.3.4 Embedding into trees

Apart from the case of embedding into Euclidean spaces, we also consider the case where the host space is a tree metric. We give a $O(1)$-approximation algorithm for computing the minimum distortion embedding of the shortest-path metric of an

Paper	From	Into	Distortion	Comments
[41]	general metrics	L_2	c	uses SDP
[39]	weighted trees	L_p	$O(c)$	
[17]	**unweighted graphs**	**trees**	$O(c)$	
	general metrics	**trees**	$(c \log n)^{O(\sqrt{\log \Delta})}$	
[25]	unweighted graphs	sub-trees	$O(c \log n)$	
[51]	outerplanar graphs	sub-trees	c	
[21]	unweighted graphs	sub-trees	NP-complete	
[28]	planar graphs	sub-trees	NP-complete	
[4]	general metrics	ultrametrics	c	
[20]	unweighted graphs	$\mathbb{R}^1$	$O(c^2)$	implies $\sqrt{n}$-approx.
			c	c is constant
			$> ac$	a-hard for some $a > 1$
	unweighted trees	$\mathbb{R}^1$	$O(c^{3/2}\sqrt{\log c})$	
[18]	**general metrics**	$\mathbb{R}^1$	$O(\Delta^{3/4} c^{11/4})$	
	weighted trees	$\mathbb{R}^1$	$c^{O(1)}$	
	weighted trees	$\mathbb{R}^1$	$\Omega(n^{1/12} c)$	unless P = NP
[17]	**general metrics**	$\mathbb{R}^1$	$(c \log n)^{O(\sqrt{\log \Delta})}$	
[20]	subsets of a sphere	$\mathbb{R}^2$	$3c$	
[19]	**ultrametrics**	$\mathbb{R}^d$	$c^{O(d)}$	
[48]	**ultrametrics**	$\mathbb{R}^d$	$c \log^{O(d)} \Delta$	
[46]	**general metrics**	$\mathbb{R}^d$	$\Omega(n^{1/(17d)} c)$	unless P = NP, $d \geq 2$

Table 1.1: Results on relative embeddings. We use c to denote the optimal distortion, and n to denote the number of points in the input metric. The results presented in this thesis are in boldface.

unweighted graph into a tree. For general metric spaces, we present an algorithm which given an n-point metric that embeds into a tree with distortion c, computes an embedding with distortion $(c \log n)^{O(\sqrt{\log \Delta})}$.

1.4 Related work

Absolute embeddings It has been shown by Alon [3] that the $(1+\varepsilon)$-embedding due to Johnson and Lindenstrauss of subset of Euclidean space into $\ell_2^{O(\varepsilon^{-2} \log n)}$, is essentially optimal.

For the case of embedding ultrametrics into low-dimensional spaces, it has been shown by Bartal and Mendel ([10]) that for any $\varepsilon > 0$, any ultrametric can be embedded into $\ell_p^{O(\varepsilon^{-2} \log n)}$, with distortion $1 + \varepsilon$.

Matoušek [43] has shown that for any $d \geq 1$, any n-point metric can be embedded into $\mathbb{R}^d$ with distortion $\tilde{O}(n^{2/d})$. He also showed that there exist metric spaces for which any embedding has distortion at least $\Omega(n^{1/\lfloor (d+1)/2 \rfloor})$, which implies that the upper bound is almost tight. Gupta [29] has shown that the above upper bound can be improved to $O(n^{1/(d-1)})$ for the case of embedding trees into $\mathbb{R}^d$. Babilon et al. [6] showed that unweighted trees embed into $\mathbb{R}^2$ with distortion $O(n^{1/2})$. This result has been extended to unweighted outerplanar graphs by Bateni et al. [11], who also showed that there exist unweighted planar graphs that require distortion $\Omega(n^{2/3})$ for any embedding into $\mathbb{R}^2$.

Relative embeddings Badoiu et al. [20] have given a $O(n^{1/2})$-approximation algorithm for embedding unweighted graphs into $\mathbb{R}^1$. They also gave an improved $O(n^{1/3})$-approximation algorithm for the case where the input graph is an unweighted tree. unweighted trees and an $O(n^{1/2})$ approximation In the same paper, they also present a $O(1)$-approximation algorithm for embedding subsets of the 2-dimensional sphere into $\mathbb{R}^2$.

For the case of embedding into ℓ_1, Avis and Deza [5] have shown that it is NP-hard to decide whether a given metric space embeds isometrically (i.e. with distortion 1). Interestingly, it has been shown by Malitz and Malitz [42] (see also Edmonds [24]) that deciding isometric embedding into 2-dimensional ℓ_1 can be done in polynomial time, while Edmonds [24] has shown that it is NP-hard for 3-dimensional ℓ_1.

Linial et al. [41] observed that an embedding with the smallest possible distortion into ℓ_2 (or equivalently, into a Euclidean space of arbitrary dimension) can be computed in polynomial time via semidefinite programming. In contrast, it is well known that deciding *isometric* embeddability in ℓ_1 is NP-hard (see [23]). Lee et al. [39] obtained an $O(1)$-approximation algorithm for embedding weighted trees into ℓ_p.

In the context of Multi-Dimensional Scaling, the problem has been a subject of extensive applied research during the last few decades (e.g., see [47] web page, or [37]). However, almost all known algorithms are heuristic. As such, they can get stuck in local minima, and do not provide any global guarantees on solution quality ([37], section 2).

The study of the problem of approximating metrics by tree metrics has been initiated in [27, 1], where the authors give an $O(1)$-approximation algorithm for embedding metrics into tree metrics. They also provide exact algorithms for embeddings into simpler metrics, called *ultrametrics*. However, instead of the *multiplicative* distortion (defined as above), their algorithms optimize the *additive* distortion; that is, the quantity $\max_{p,q} |D(p,q) - D'(p,q)|$. The same problem has recently been studied also for the case of minimizing the L_p norm of the differences [32, 2]. In a recent paper [2], a $(\log n \log\log n)^{1/p}$-approximation has been obtained for this problem.

In general, minimizing an additive measure suffers from the "scale insensitivity" problem: local structures can be distorted in arbitrary way, while the global structure is highly over-constrained. Although the result of [2] holds even for a weighted version of the L_p norm, it does not imply an approximation for minimizing the multiplicative distortion. The multiplicative distortion, which we employ in this paper, does not suffer from the scale insensitivity problem.

The problem of embedding into a tree with minimum multiplicative distortion is closely related to the problem of computing a minimum-stretch spanning tree. We mention the work of [52, 21, 62, 50, 51, 28, 25]. For unweighted graphs, the best known approximation is an $O(\log n)$-approximation algorithm [25].

The problem of approximating the minimum distortion embedding has also been studied for the case where we are given two metric spaces M, M' of the same cardinality and we want to compute the minimum distortion bijection between them; we refer to [38, 49, 31, 22].

1.5 Further directions and open problems

1.5.1 $c^{O(1)}$-embeddings into $\mathbb{R}^d$

Our main hardness result for embedding into $\mathbb{R}^d$ shows that it is NP-hard to approximate the minimum-distortion embedding of an n-point metric space into $\mathbb{R}^d$ within a factor of $\Omega(n^{1/(17d)})$. Since by a result of Matoušek [43] any n-point metric space embeds into $\mathbb{R}^d$ with distortion $\tilde{O}(n^{2/d})$, it follows that our hardness result is essentially optimal, up to a constant factor in the exponent. However, this result leaves open the possibility that for any fixed $d \geq 1$, there exists an algorithm that given a metric space that c-embeds into $\mathbb{R}^d$, computes an embedding with distortion $c^{O(1)}$. Such a result would have been very interesting since when the input space embeds with small distortion (which is perhaps the most interesting case), the algorithm would compute an embedding with distortion which is also relatively small. We also remark that such a result has been obtained for the case of embedding unweighted graphs into $\mathbb{R}^1$, weighted trees into $\mathbb{R}^1$, and ultrametrics into $\mathbb{R}^d$.

A recent result by Matoušek and Sidiropoulos [45] shows that such an algorithm does not exist for $d \geq 3$. More specifically, they showed that there exist constants $\alpha, \beta > 0$, such that for any fixed $d \geq 3$ it is NP-hard to distinguish between n-points metric spaces that embed into $\mathbb{R}^d$ with distortion at most α, or at least $\Omega(n^{\beta/d})$. The case of embedding into $\mathbb{R}^1$ and $\mathbb{R}^2$ remains an intriguing open problem.

1.5.2 Embedding into trees

A strengthening of the above $c^{O(1)}$-embedding question of general metrics into $\mathbb{R}^1$ can be obtained by considering the problem of embedding into trees. In particular, one can show using Lemma 15 for composing relative embeddings, that if there is a polynomial-time $c^{O(1)}$-distortion algorithm for embedding general metrics into trees, then there is also such an algorithm for embedding general metrics into $\mathbb{R}^1$. In fact, it is not even known whether there exists a $O(1)$-approximation algorithm for embedding into trees. Resolving these questions is an interesting open problem.

1.6 Notation and definitions

A *metric space* is a pair $M = (X, D)$, where X is a finite set, and $D : X \times X \to \mathbb{R}_{\geq 0}$. We will typically refer to the elements of X as *points*. For each pair $x, y \in X$, we say that $D(x, y)$ is the *distance between x and y in M*. The function D satisfies the following properties:

Positive definiteness: For each $x, y \in X$, $D(x, y) = 0$ iff $x = y$.

Symmetry: For each $x, y \in X$, $D(x, y) = D(y, x)$.

Triangle Inequality: For each $x, y, z \in X$, $D(x, y) \leq D(x, z) + D(z, y)$.

A metric space $M = (X, D)$ is called an *ultrametric* if for any $x, y, z \in X$, $D(x, y) \leq \max\{D(x, z), D(z, y)\}$. Other classes of metric spaces that we are interested in throughout this thesis are the tree metrics and the unweighted graph metrics, which are the shortest-path metrics of (weighted) graph-theoretic trees and unweighted graphs respectively.

Chapter 2

Embedding into $\mathbb{R}^1$

In this chapter, we consider the problem of embedding into $\mathbb{R}^1$. The algorithms of [20] were designed for *unweighted graphs* and thus provide only very weak guarantees for the problem. Specifically, assume that the minimum interpoint distance between the points is 1 and the maximum distance is Δ. Then, by scaling, one can obtain algorithms for weighted graphs, with approximation factor multiplied by Δ.

Our main result is an algorithm that, given a general metric c-embeddable into the line, constructs an embedding with distortion $O(\Delta^{4/5}c^{13/5})$. The algorithm uses a novel method for traversing a weighted graph. It also uses a modification of the unweighted-graph algorithm from [20] as a subroutine, with a more general analysis.

The results presented in this chapter are from [18].

2.1 Overview of the algorithm

In this section we will present a polynomial-time algorithm that given a metric $M = (X, D)$ of spread Δ that c-embeds into the line, computes an embedding of M into the line, with distortion $O(c^{11/4}\Delta^{3/4})$. Since it is known [43] that any n-point metric embeds into the line with distortion $O(n)$, we can assume that $\Delta = O(n^{4/3})$.

We view the metric $M = (X, D)$ as a complete graph G defined on vertex set X, where the weight of each edge $e = \{u, v\}$ is $D(u, v)$. As a first step, our algorithm partitions the point set X into sub-sets $X_1, \ldots, X_\ell$, as follows. Let W be a large integer to be specified later. Remove all the edges of weight greater than W from G, and denote the resulting connected components by $C_1, \dots, C_\ell$. Then for each $i : 1 \le i \le \ell$, X_i is the set of vertices of C_i. Let G_i be the subgraph of G induced by X_i. Our algorithm computes a low-distortion embedding for each G_i separately, and then concatenates the embeddings to obtain the final embedding of M. In order for the concatenation to have small distortion, we need the length of the embedding of each component to be sufficiently small (relatively to W). The following simple lemma, essentially shown in [43], gives an embedding that will be used as a subroutine.

Lemma 1. *Let $M = (X, D)$ be a metric with minimum distance 1, and let T be a spanning tree of M. Then we can compute in polynomial time an embedding of M into the line, with distortion $O(cost(T))$, and length $O(cost(T))$.*

Proof. Embed M according to the order of appearance of the points of M in a DFS traversal of T. Since each edge is traversed only a constant number of times, the total length and distortion of the embedding follows. $\square$

Our algorithm proceeds as follows. For each $i : 1 \le i \le \ell$, we compute a spanning tree T_i of G_i, that has the following properties: the cost of T_i is low, and there exists a walk on T_i that gives a small distortion embedding of G_i. We can then view the concatenation of the embeddings of the components as if it is obtained by a walk on a spanning tree T of G. We show that the cost of T is small, and thus the total length of the embedding of G is also small. Since the minimum distance between components is large, the inter-component distortion is small.

2.2 Embedding the components

In this section we concentrate on some component G_i, and we show how to embed it into a line.

Let H be the graph on vertex set X_i, obtained by removing all the edges of length at least W from G_i, and let H' be the graph obtained by removing all the edges of length at least cW from G_i. For any pair of vertices $x, y \in X_i$, let $D_H(x, y)$

and $D_{H'}(x,y)$ be the shortest-path distances between x and y in H and H', respectively. Recall that by the definition of X_i, H is a connected graph, and observe that $D_H(x,y) \geq D_{H'}(x,y) \geq D(x,y)$.

Lemma 2. *For any $x, y \in X_i$, $D_{H'}(x,y) \leq cD(x,y)$.*

Proof. Let f be an optimal non-contracting embedding of G_i, with distortion at most c. Consider any pair u, v of vertices that are embedded consecutively in f. We start by showing that $D(u,v) \leq cW$. Let T be the minimum spanning tree of H. If edge $\{u,v\}$ belongs to T, then $D(u,v) \leq W$. Otherwise, since T is connected, there is an edge $e = \{u', v'\}$ in tree T, such that both u and v are embedded inside e. But then $D(u', v') \leq W$, and since the embedding distortion is at most c, $|f(u) - f(v)| \leq |f(u') - f(v')| \leq cW$. As the embedding is non-contracting, $D(u,v) \leq cW$ must hold.

Consider now some pair $x, y \in X_i$ of vertices. If no vertex is embedded between x and y, then by the above argument, $D(x,y) \leq cW$, and thus the edge $\{x,y\}$ is in H' and $D_{H'}(x,y) = D(x,y)$. Otherwise, let $z_1, \ldots, z_k$ be the vertices appearing in the embedding f between x and y (in this order). Then the edges $\{x, z_1\}, \{z_1, z_2\}, \ldots, \{z_{k-1}, z_k\}, \{z_k, y\}$ all belong to H', and therefore

$$\begin{aligned}
D_{H'}(x,y) &\leq D_{H'}(x,z_1) + D_{H'}(z_1,z_2) + \ldots D_{H'}(z_{k-1},z_k) + D_{H'}(z_k,y) \\
&= D(x,z_1) + D(z_1,z_2) + \ldots D(z_{k-1},z_k) + D(z_k,y) \\
&\leq |f(x) - f(z_1)| + |f(z_1) - f(z_2)| + \ldots \\
&\quad +|f(z_{k-1}) - f(z_k)| + |f(z_k) - f(y)| \\
&= |f(x) - f(y)| \leq cD(x,y)
\end{aligned}$$

□

We can now concentrate on embedding graph H'. Since the weight of each edge in graph H' is bounded by $O(cW)$, we can use a modified version of the algorithm of [20] to embed each G_i. First, we need the following technical Claim.

Claim 1. *There exists a shortest path $p = v_1, \ldots, v_k$, from u to u' in H', such that for any i, j, with $|i - j| > 1$, $D(v_i, v_j) = \Omega(W|i - j|)$.*

Proof. Pick an arbitrary shortest path, and repeat the following: while there exist consecutive vertices x_1, x_2, x_3 in p, with $D_{H'}(x_1, x_3) < cW$, remove x_2 from p, and add the edge $\{x_1, x_3\}$ in p. $\square$

The algorithm works as follows. We start with the graph H', and we guess points u, u', such that there exists an optimal embedding of G_i having u and u' as the left-most and right-most point respectively. Let $p = (v_1, \ldots, v_k)$ be the shortest path from u to u' on H' (here $v_1 = u$ and $v_k = u'$), that is given by Claim 1. We partition X_i into clusters $V_1, \ldots, V_k$, as follows. Each vertex $x \in X_i$ belongs to cluster V_j, that minimizes $D(x, v_j)$.

Our next step is constructing super-clusters $U_1, \ldots, U_s$, where the partition induced by $\{V_j\}_{j=1}^k$ is a refinement of the partition induced by $\{U_j\}_{j=1}^s$, such that there is a small-cost spanning tree T' of G_i that "respects" the partition induced by $\{U_j\}_{j=1}^s$. More precisely, each edge of T' is either contained in a super-cluster U_i, or it is an edge of the path p. The final embedding of G_i is obtained by a walk on T', that traverses the super-clusters $U_1, \ldots, U_s$ in this order.

Note that there exist metrics over G_i for which any spanning tree that "respects" the partition induced by V_j's is much more expensive that the minimum spanning tree. Thus, we cannot simply use $U_j = V_j$.

We now show how to construct the super-clusters $U_1, \ldots, U_s$. We first need the following three technical claims, which constitute a natural extensions of similar claims from [20] to the weighted case.

Claim 2. *For each $i : 1 \leq i \leq k$, $\max_{u \in V_i} \{D(u, v_i)\} \leq c^2 W/2$.*

Proof. Let $u \in V_i$. Consider the optimal embedding f. Since $f(v_1) = \min_{w \in X} f(w)$, and $f(v_k) = \max_{w \in X} f(w)$, it follows that there exists j, with $1 \leq j < k$, such that

$$\min\{f(v_j), f(v_{j+1})\} < f(u) < \max\{f(v_j), f(v_{j+1})\}.$$

Assume w.l.o.g., that $f(v_j) < f(u) < f(v_{j+1})$. We have $D(u, v_j) \geq D(u, v_i)$, since $u \in V_i$. Since f is non-contracting, we obtain $f(u) - f(v_j) \geq D(u, v_j) \geq D(u, v_i)$.

Similarly, we have $f(v_{j+1})-f(u) \geq D(u,v_i)$. Thus, $f(v_{j+1})-f(v_j) \geq 2D(u,v_i)$. Since $\{v_j, v_{j+1}\} \in E(G')$, we have $D(v_j, v_{j+1}) \leq cW$. Thus, $c \geq \frac{f(v_{j+1})-f(v_j)}{D(v_{j+1},v_j)} \geq \frac{2D(u,v_i)}{cW}$. $\square$

Claim 3. *For each $r \geq 1$, and for each $i : 1 \leq i \leq k-r+1$, $\sum_{j=i}^{i+r-1} |V_i| \leq c^2W(c+r-1)+1$.*

Proof. Let $A = \bigcup_{j=1}^{i+r-1} V_i$. Let $x = \operatorname{argmin}_{u\in A} f(u)$, and $y = \operatorname{argmax}_{u\in A} f(u)$. Let also $x \in V_i$, and $y \in V_j$. Clearly, $|f(v_i)-f(v_j)| \leq cD(v_i,v_j) \leq cD_{G'}(v_i,v_j) \leq c^2W|i-j| \leq c^2W(r-1)$. By Claim 2, we have $D(x,v_i) \leq c^2W/2$, and $D(y,v_j) \leq c^2W/2$. Thus, $|f(x)-f(v_i)| \leq cD(x,v_i) \leq c^3W/2$, and similarly $|f(y)-f(v_j)| \leq c^3W/2$. It follows that $|f(x)-f(y)| \leq |f(x)-f(v_i)|+|f(v_i)-f(v_j)|+|f(v_j)-f(y)| \leq c^3W+c^2W(r-1)$. Note that by the choice of x, y, and since the minimum distance in M is 1, and f is non-contracting, we have $\sum_{j=i}^{i+r-1} |V_i| \leq |f(x)-f(y)|+1$, and the assertion follows. $\square$

Claim 4. *If $\{x,y\} \in E(H')$, where $x \in V_i$, and $y \in V_j$, then $D(v_i,v_j) \leq cW + c^2W$, and $|i-j| = O(c^2)$.*

Proof. Since $\{x,y\} \in E(G')$, we have $D(x,y) \leq cW$. By Claim 2, we have $D(x,v_i) \leq c^2W/2$, and $D(y,v_j) \leq c^2W/2$. Thus, $D(v_i,v_j) \leq D(v_i,x)+D(x,y)+D(y,v_j) \leq cW + c^2W$.

By Lemma 2, we have that $D_{G'}(v_i,v_j) \leq cD(v_i,v_j) \leq c^2W + c^3W$. Since every edge of G' has length at least 1, we have $|i-j| \leq D_{G'}(v_i,v_j) \leq c^2W + c^3W$. $\square$

Let α be an integer with $0 \leq \alpha < c^4W$. We partition the set X_i into super-clusters $U_1, \ldots, U_s$, such that for each $l : 1 \leq l \leq s$, U_l is the union of c^4W consecutive clusters V_j, where the indexes j are shifted by α. We refer to the above partition as *α-shifted*.

Claim 5. *Let T be an MST of G_i. We can compute in polynomial time a spanning tree T' of G_i, with cost(T') $= O($cost$(T))$, and an α-shifted partition of X_i, such that for any edge $\{x,y\}$ of T', either both $x, y \in U_l$ for some $l : 1 \leq l \leq s$, or $x = v_j$ and $y = v_{j+1}$ for some $j : 1 \leq j < k$.*

Proof. Observe that since H is connected, all the edges of T can have length at most W, and thus T is a subgraph of both H and H'. Consider the α-shifted partition obtained by picking $\alpha \in \{0, \dots, c^4W - 1\}$, uniformly at random. Let T' be the spanning tree obtained from T as follows: For all edges $\{x, y\}$ of T, such that $x \in V_i \subseteq U_{i'}$, and $y \in V_j \subseteq U_{j'}$, where $i' \neq j'$, we remove $\{x, y\}$ from T, and we add the edges $\{x, v_i\}$, $\{y, v_j\}$, and the edges on the subpath of p from v_i to v_j. Finally, if the resulting graph T' contains cycles, we remove edges in an arbitrary order, until T' becomes a tree. Note that although T' is a spanning tree of G_i, it is not necessarily a subtree of H'.

Clearly, since the edges $\{x, v_i\}$, and $\{y, v_j\}$ that we add at each iteration of the above procedure are contained in the sets $U_{i'}$, and $U_{j'}$ respectively, it follows that T' satisfies the condition of the Claim.

We will next show that the expectation of $\text{cost}(T')$, taken over the random choice of α, is $O(\text{cost}(T))$. For any edge $\{x, y\}$ that we remove from T, the cost of T' is increased by the sum of $D(x, v_i)$ and $D(y, v_j)$, plus the length of the shortest path from v_i to v_j in H'. Observe that the total increase of $\text{cost}(T')$ due to the subpaths of p that we add, is at most $\text{cost}(T)$. Thus, it suffices to bound the increase of $\text{cost}(T')$ due to the edges $\{x, v_i\}$, and $\{y, v_j\}$.

By Claim 2, $D(x, v_i) \leq c^2W/2$, and $D(y, v_j) \leq c^2W/2$. Thus, for each edge $\{x, y\}$ that we remove from T, the cost of the resulting T' is increased by at most $O(c^2W)$.

For each i, the set $U_i \cup U_{i+1}$ contains $\Omega(c^4W)$ consecutive clusters V_j. Also, by Claim 4 the difference between the indexes of the clusters V_{t_1}, V_{t_2} containing the endpoints of an edge, is at most $|t_1 - t_2| = O(c^2)$. Thus, the probability that an edge of T is removed, is at most $O(\frac{1}{c^2W})$, and the expected total cost of the edges in $E(T') \setminus E(T)$ is $O(|X_i|) = O(\text{cost}(T))$. Therefore, the expectation of $\text{cost}(T')$, is at most $O(\text{cost}(T))$. The Claim follows by the linearity of expectation, and by the fact that there are only few choices for α. $\square$

Let $U_1, \dots, U_s$ be an α-shifted partition, satisfying the conditions of Claim 5, and let T' be the corresponding tree. Clearly, the subgraph $T'[U_i]$ induced by each U_i is a connected subtree of T'. For each U_i, we construct an embedding into the line by applying Lemma 1 on the spanning tree $T'[U_i]$. By Claim 3, $|U_i| = O(c^6W^2)$, and by Claim 2, the cost of the spanning tree $T'[U_i]$ of U_i is at most $O(|U_i|c^2W) = O(c^8W^3)$.

Therefore, the embedding of each U_i, given by Lemma 1 has distortion $O(c^8W^3)$, and length $O(c^8W^3)$.

Finally, we construct an embedding for G_i by concatenating the embeddings computed for the sets $U_1, U_2, \ldots, U_s$, while leaving sufficient space between each consecutive pair of super-clusters, so that we satisfy non-contraction.

Lemma 3. *The above algorithm produces a non-contracting embedding of G_i with distortion $O(c^8W^3)$ and length $O(cost(MST(G_i)))$.*

Proof. Let g be the embedding produced by the algorithm. Clearly, g is non-contracting. Consider now a a pair of points $x, y \in X$, such that $x \in U_i$, and $y \in U_j$. If $|i-j| \leq 1$, then $|g(x)-g(y)| = O(c^8W^3)$, and thus the distortion of $D(x,y)$ is at most $O(c^8W^3)$.

Assume now that $|i-j| \geq 2$, and $x \in V_{i'}$, $y \in V_{j'}$. Then $|g(x)-g(y)| = O(|i-j| \cdot c^8W^3)$. On the other hand, $D(x,y) \geq D(v_{i'}, v_{j'}) - D(v_{i'}, x) - D(v_{j'}, y) \geq D(v_{i'}, v_{j'}) - c^2W \geq D_{H'}(v_{i'}, v_{j'})/c - c^2W \geq |i'-j'|/c - c^2W = \Omega(|i-j|c^4W^2)$. Thus, the distortion on $\{x,y\}$ is $O(c^7W^2)$. In total, the maximum distortion of the embedding g is $O(c^8W^3)$.

In order to bound the length of the constructed embedding, consider a walk on T' that visits the vertices of T according to their appearance in the line, from left to right. It is easy to see that this walk traverses each edge at most 4 times. Thus, the length of the embedding, which is equal to the total length of the walk is at most $4\text{cost}(T') = O(\text{cost}(T))$. $\square$

2.3 The final embedding

We are now ready to give a detailed description of the final algorithm. Assume that the minimum distance in M is 1, and the diameter is Δ. Let $H = (X, E)$ be a graph, such that an edge $(u,v) \in E$ iff $D(u,v) \leq W$, for a threshold W, to be determined later. We use the algorithm presented above to embed every connected component $G_1, \ldots, G_k$ of H. Let $f_1, f_2, \ldots, f_k$ be the embeddings that we get for the components $G_1, G_2, \ldots G_k$ using the above algorithm, and let T be a minimum spanning tree of G.

It is easy to see that T connects the components G_i using exactly $k-1$ edges.[1] We compute our final embedding f as follows. Fix an arbitrary Eulerian walk of T. Let P be the permutation of $(G_1, G_2, \ldots, G_k)$ that corresponds to the order of the first occurrence of any node of G_i in our traversal. Compute embedding f by concatenating the embeddings f_i of components G_i in the order of this permutation. Let T_i be the minimum spanning tree of G_i. Between every 2 consecutive embeddings in the permutation f_i and f_j, leave space $\max_{u \in G_i, v \in G_j} \{D(u,v)\} = D(a,b) + O(\text{cost}(T_i)) + O(cost(T_j))$, where $D(a,b)$ is the smallest distance between components G_i and G_j. This implies the next two Lemmas

Lemma 4. *The length of f is at most $O(c\Delta)$.*

Proof. The length of f is the sum of the lengths of all f_i and the space that we leave between every 2 consecutive f_i, f_j's. Then, by Lemma 3, the length of f_i is $O(c \cdot \text{cost}(T_i))$. Thus, the sum of the lengths of all f_i's is $O(c \cdot \text{cost}(T))$. The total space that we leave between all pairs of consecutive embeddings f_i is $\text{cost}(T) + 2\sum_{i=1}^{k} O(\text{cost}(T_i)) = O(\text{cost}(T))$. Therefore the total length of the embedding f is $O(\text{cost}(T))$. At the same time, the cost of T is at most the length of the optimal embedding f, which is $O(c\Delta)$. The statement follows. ☐

Lemma 5. *Let $a \in G_i, b \in G_j$ for $i \neq j$. Then $W \leq D(a,b) \leq |f(a) - f(b)| \leq O(c\Delta) \leq O(cD(a,b)\frac{\Delta}{W})$*

Proof. The first part $D(a,b) \leq |f(a) - f(b)|$ is trivial by construction, since we left enough space between components G_i and G_j. Since a and b are in difference connected components, we have $D(a,b) > W$. Using Lemma 4 we have that $|f(a) - f(b)| = O(c\Delta) = O(c\Delta\frac{D(a,b)}{W}) = O(cD(a,b)\frac{\Delta}{W})$. ☐

Theorem 1. *Let $M = (X, D)$ be a metric with spread Δ, that embeds into the line with distortion c. Then, we can compute in polynomial time an embedding of M into the line, of distortion $O(c^{11/4}\Delta^{3/4})$.*

[1] Follows from correctness of Kruskal's algorithm. These $k-1$ edges are exactly the last edges to be added because they are bigger than W and within components we have edges smaller than W

Proof. Consider any pair of points. If they belong to different components, their distance distortion is $O(c\Delta/W)$ (Lemma 5). If they belong to the same component, their distance distortion is $O(c^8W^3)$ (Lemma 3). Setting $W = \Delta^{1/4}c^{-7/4}$ gives the claimed distortion bound. $\square$

Chapter 3

Embedding into trees and improved embeddings into $\mathbb{R}^1$

In this chapter we consider the problem of approximating minimum distortion for embedding general metrics into tree metrics, i.e., shortest path metric over (weighted) trees. Specifically, if the input metric is an unweighted graph, we give a $O(1)$-approximation algorithm for this problem. For general metrics, we give an algorithm such that if the input metric is c-embeddable into some tree metric, produces an embedding with distortion $\alpha(c \log n)^{O(\log_\alpha \Delta)}$, for any $\alpha \geq 1$. In particular, by setting $\alpha = 2^{\sqrt{\log \Delta}}$, we obtain distortion $(c \log n)^{O(\sqrt{\log \Delta})}$. Alternatively, when $\Delta = n^{O(1)}$, by setting $\alpha = n^{\varepsilon}$, we obtain distortion $n^{\varepsilon}(c \log n)^{O(1/\varepsilon)}$. This in turn yields an $O(n^{1-\beta})$-approximation for some $\beta > 0$, since it is always possible to construct an embedding with distortion $O(n)$ in polynomial time [43].

Further, we show that by composing our approximation algorithm for embedding general metrics into trees, with the approximation algorithm from [18] for embedding trees into the line, we obtain an improved [1] approximation algorithm for embedding general metrics into the line. The distortion guarantee from Theorem 1 is $c^{O(1)}\Delta^{3/4}$, while the composition results in distortion $(c \log n)^{O(\sqrt{\log \Delta})}$. In fact, we provide a general framework for composing relative embeddings which could be useful elsewhere.

[1]Strictly speaking, the guarantees are incomparable, but the dependence on Δ in our algorithm is a great improvement over the earlier bound.

For the special case where the input is an unweighted graph metric, we also study the relation between embedding into trees, and embedding into spanning subtrees. An $O(\log n)$-approximation algorithm is known [25] for this problem. We show that if an unweighted graph metric embeds into a tree with distortion c, then it also embeds into a spanning subtree with distortion $O(c \log n)$. We also exhibit an infinite family of graphs that almost achieves this bound; each graph in the family embeds into a tree with distortion $O(\log n)$, while any embedding into a spanning subtree has distortion $\Omega(\log^2 n / \log \log n)$. We remark that by composing the upper bound with our $O(1)$-approximation algorithm for unweighted graphs, we recover the result of [25].

The results presented in this chapter are from [17].

3.1 Preliminaries

The input to our problem is a graph $G = (V, E)$. For $u, v \in V(G)$ let $D_G(u, v)$ denote the shortest-path distance between u and v in G. We assume that all the edges of G have weight at least 1. If G is weighted let W_G denote the maximum edge weight of G, and let $W_G = 1$ otherwise.

For any finite metric space $M = (X, D)$, we assume that the minimum distance in M is at least 1. M is called a *tree metric* iff it is the shortest-path metric of a subset of the vertices of a weighted tree. For a graph $G = (V, E)$, and $\gamma \geq 1$ we say that G γ-approximates M if $V(G) \subseteq X$, and for each $u, v \in V(G)$, $D(u, v) \leq D_G(u, v) \leq \gamma D(u, v)$. We say that M c-embeds into a tree if there exists an embedding of M into a tree with distortion at most c. When considering an embedding into a tree, we assume unless stateted otherwise that the tree might contain steiner nodes. By a result of Gupta [30], after computing the embedding we can remove the steiner nodes losing at most a $O(1)$ factor in the distortion (and thus also in the approximation factor).

Definition 1 (α-restricted subgraphs)**.** *For a weighted graph $G = (V, E)$, and for $\alpha > 0$, the* α-restricted subgraph of G *is defined as the graph obtained from G after* *removing all the edges of weight greater than α. Similarly, for a metric $M = (X, D)$, the α-restricted subgraph of M is defined as the weighted graph on vertex set X, where an edge $\{u, v\}$ appears in G iff $D(u, v) \leq \alpha$, and the weight of every edge $\{u, v\}$ is equal to $D(u, v)$.*

3.2 A forbidden-structure characterization of tree-embeddability

Before we describe our algorithms, we give a combinatorial characterization of graphs that embed into trees with small distortion. For any $c > 1$, the characterization defines a forbidden structure that cannot appear in a graph that embeds into a tree with distortion at most c. This structure will be later used when analyzing our algorithms to show that the computed embedding is close to optimal.

Lemma 6. *Let $G = (V, E)$ be a (possibly weighted) graph. If there exist nodes $v_0, v_1, v_2, v_3 \in V(G)$, and $\lambda > 0$, such that*

- *for each $i : 0 \leq i < 4$, there exists a path p_i, with endpoints v_i, and $v_{i+1 \bmod 4}$, and*
- *for each $i : 0 \leq i < 4$, $D_G(p_i, p_{i+2 \bmod 4}) > \lambda W_G$,*

then, any embedding of G into a tree has distortion greater than λ.

Proof. Let $W = W_G$. Consider an optimal non-contracting embedding f of G, into a tree T. For any $u, v \in V(G)$, let $P_{u,v}$ denote the path from $f(u)$ to $f(v)$, in T. For each i, with $0 \leq i < 4$, define T_i as the minimum subtree of T, which contains all the images of the nodes of p_i. Since each T_i is minimum, it follows that all the leaves of T_i are nodes of $f(p_i)$.

Claim 6. *For each i, with $0 \leq i < 4$, we have $T_i = \bigcup_{\{u,v\} \in E(p_i)} P_{u,v}$.*

Proof. Assume that the assertion is not true. That is, there exists $x \in V(T_i)$, such that for any $\{u, v\} \in E(p_i)$, the path $P_{u,v}$ does not visit x. Clearly, $x \notin V(p_i)$, and thus x is not a leaf. Let $T_i^1, T_i^2, \ldots, T_i^j$, be the connected components obtained by removing x from T_i. Since for every $\{u,v\} \in E(p_i)$, $P_{u,v}$ does not visit x, it follows that there is no edge $\{u,v\} \in E(p_i)$, with $u \in T_i^a$, $v \in T_i^b$, and $a \neq b$. This however, implies that p_i is not connected, a contradiction. $\square$

Claim 7. *For each i, with $0 \leq i < 4$, we have $T_i \cap T_{i+2 \mod 4} = \emptyset$.*

Proof. Assume that the assertion does not hold. That is, there exists i, with $0 \leq i < 4$, such that $T_i \cap T_{i+2 \mod 4} \neq \emptyset$. We have to consider the following two cases:

Case 1: *$T_i \cap T_{i+2 \mod 4}$ contains a node from $V(p_i) \cup V(p_{i+2 \mod 4})$.* W.l.o.g., we assume that there exists $w \in V(p_{i+2 \mod 4})$, such that $w \in T_i \cap T_{i+2 \mod 4}$. By Claim 6, it follows that there exists $\{u,v\} \in E(p_i)$, such that $f(w)$ lies on $P_{u,v}$. This implies $D_T(f(u), f(v)) = D_T(f(u), f(w)) + D_T(f(w), f(v))$. On the other hand, we have $D_G(p_i, p_{i+2 \mod 4}) > \lambda W$, and since f is non-contracting, we obtain $D_T(f(u), f(v)) > 2\lambda W$. Thus, $c \geq D_T(f(u), f(v))/D_G(u,v)$. Since $\{u,v\} \in E(G)$, and the maximum edge weight in G is at most W, we have $D_G(u,v) \leq W$, and thus $c > 2\lambda$.

Case 2: *$T_i \cap T_{i+2 \mod 4}$ does not contain nodes from $V(p_i) \cup V(p_{i+2 \mod 4})$.* Let $w \in T_i \cap T_{i+2 \mod 4}$. By Claim 6, there exist $\{u_1, v_1\} \in E(p_i)$, and $\{u_2, v_2\} \in E(p_{i+2 \mod 4})$, such that w lies in both P_{u_1,v_1}, and P_{u_2,v_2}. We have

$$\begin{aligned}
D_T(f(u_1), f(v_1)) + D_T(f(u_2), f(v_2)) &= D_T(f(u_1), f(w)) + D_T(f(w), f(v_1)) \\
&\quad + D_T(f(u_2), f(w)) + D_T(f(w), f(v_2)) \\
&\geq D_T(f(u_1), f(u_2)) + D_T(f(v_1), f(v_2)) \\
&\geq D_G(u_1, u_2) + D_G(v_1, v_2) \\
&\geq 2D_G(p_i, p_{i+2 \mod 4}) \\
&> 2\lambda W
\end{aligned}$$

Thus, we can assume that $D_T(f(u_1), f(v_1)) > \lambda W$. It follows that $c \geq \frac{D_T(f(u_1), f(v_1))}{D_G(u_1, v_1)} > \lambda$. $\square$

Moreover, since p_i, and $p_{i+1 \mod 4}$, share an end-point, we have $T_i \cap T_{i+1 \mod 4} \neq \emptyset$. By Claim 7, it follows, that $\bigcup_{i=0}^{3} T_i \subseteq T$, contains a cycle, a contradiction. $\square$

3.3 Tree-like decompositions

In this section we describe a graph partitioning procedure which is a basic step in our algorithms. Intuitively, the procedure partitions a graph into a set of clusters, and arranges the clusters in a tree, so that the structure of the tree of clusters resembles the structure of the original graph.

Formally, the procedure takes as input a (possibly weighted) graph $G = (V, E)$, a vertex $r \in V(G)$, and a parameter $\lambda \geq 1$. The output of the procedure is a pair $(T^G_{\mathcal{K}}, \mathcal{K}_G)$, where $\mathcal{K}_G$ is a partition of $V(G)$, and $T^G_{\mathcal{K}}$ is a rooted tree with vertex set $\mathcal{K}_G$.

The partition $\mathcal{K}_G$ of $V(G)$ is defined as follows. For integer i, let

$$V_i \;=\; \{v \in V(G) | W_G(i-1)\lambda \leq D_G(r, v) < W_G i\lambda\}.$$

Initially, $\mathcal{K}_G$ is empty. Let t be the maximum index such that V_t is non-empty. Let $Y_i = \bigcup_{j=i}^{t} V_j$. For each $i \in [t]$, and for each connected component Z of $G[Y_i]$ that intersects V_i, we add the set $Z \cap V_i$, to the partition $\mathcal{K}_G$. Observe that some clusters in $\mathcal{K}_G$ might induce disconnected subgraphs in G.

$T^G_{\mathcal{K}}$ can now be defined as follows. For each $K, K' \in \mathcal{K}_G$, we add the edge $\{K, K'\}$ in $T^G_{\mathcal{K}}$ iff there is an edge in G between a vertex in K and a vertex in K'. The root of $T^G_{\mathcal{K}}$ is the cluster containing r. The resulting pair $(T^G_{\mathcal{K}}, \mathcal{K}_G)$ is called a (r, λ)-*tree-like decomposition of* G.

Figure 3-1 depicts the described decomposition.

Proposition 1. *$T^G_{\mathcal{K}}$ is a tree.*

Proof. Let $u, v \in V(G)$. Since G is connected, there is a path p from u to v in G. Let $p = x_1, \ldots, x_{|p|}$. For each $i \in \{1, \ldots, |p|\}$, let $K_i \in \mathcal{K}_G$ be such that $x_i \in K_i$. It is easy to verify that the sequence $\{K_i\}_{i=1}^{|p|}$ contains a sub-sequence that corresponds to a path in $T^G_{\mathcal{K}}$. Thus, $T^G_{\mathcal{K}}$ is connected.

It is easy to show by induction on i that for $i = t, \ldots, 1$, the subset $L_i \subseteq \mathcal{K}_G$ that is obtained by partitioning $\bigcup_{j=i}^{t} V_j$, induce a forest in $T^G_{\mathcal{K}}$. Since $L_1 = \mathcal{K}_G$, and $T^G_{\mathcal{K}}$ is

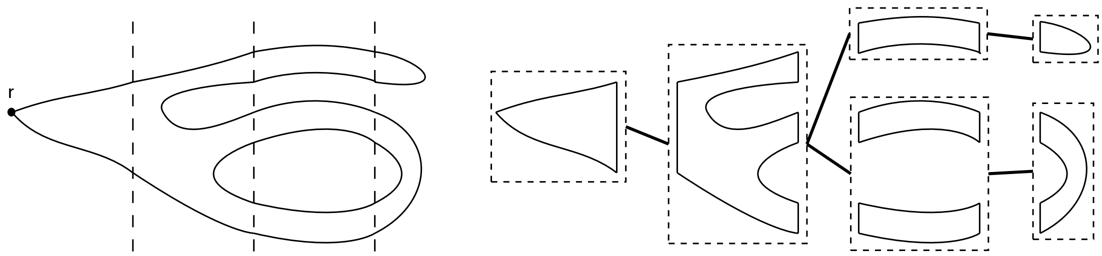


Figure 3-1: An example of a tree-like decomposition of a graph.

connected, it follows that $T_{\mathcal{K}}^G$ is a tree. $\square$

3.3.1 Properties of tree-like decompositions

Before using the tree-like decompositions in our algorithms, we will show that for a certain range of the decomposition parameters, they exhibit some usefull properties.

We will first bound the diameter of the clusters in $\mathcal{K}_G$. The intuition behind the proof is as follows. If a cluster K is long enough, then starting from a pair of vertices in $x, y \in K$ that are far from each other, and tracing the shortest paths from x and y to r, we can discover the forbidden structure of lemma 6 in G. Applying lemma 6 we obtain a lower bound on the optimal distortion, contradicting the fact that G embeds into a tree with small distortion.

Lemma 7. *Let $G = (V, E)$ be a graph that γ-embeds into a tree, let $r \in V(G)$, and let $(T_{\mathcal{K}}^G, \mathcal{K}_G)$ be a (r, γ)-tree-like decomposition of G. Then, for any $K \in \mathcal{K}_G$, and for any $u, v \in K$, $D_G(u, v) \leq 20\gamma W_G$.*

Proof. Assume that the assertion is not true, and pick $K \in \mathcal{K}_G$, and vertices $x, y \in K$, such that $D_G(x, y) > 20\gamma W_G$. Recall that $\mathcal{K}_G$ was obtained by partitioning the vertices of G according to their distance from r. Let q_x, and q_y be the shortest paths from x to r, and from y to r respectively. Let $K_1, \ldots, K_\tau$ be the branch in $T_{\mathcal{K}}^G$, such that $r \in K_1$, and $K_\tau = K$. By the construction of $\mathcal{K}_G$, we have that for any $i \in [\tau]$, for any $z \in K_i$, $D_G(r, z) \leq iW_G\gamma$. Thus, $D_G(x, y) \leq D_G(x, r) + D_G(r, y) \leq 2\tau W_G c$. Since $D_G(x, y) > 20\gamma W_G$, it follows that $\tau > 10$.

Consider now the sub-path p^x of q_x that starts from x, and terminates to the first vertex x' of $K_{\tau-2}$ visited by q_x. Define similarly p^y as the sub-path of q_y that starts from y, and terminates to the first vertex y' of $K_{\tau-2}$ visited by q_y. We will first show that $D_G(p^x, p^y) > \gamma W_G$. Observe that by the construction of $\mathcal{K}_G$, we have that $D_G(x, x') \leq 2\gamma W_G$, and also $D_G(y, y') \leq 2\gamma W_G$. Since p^x, and p^y are shortest paths, we have that for any $z \in p^x$, $D_G(x, z) \leq 2\gamma W_G$, and similarly for any $z \in p^y$, $D_G(y, z) \leq 2\gamma W_G$. Pick $z \in p^x$, and $z' \in p^y$, such that $D_G(z, z')$ is minimized. We have $D_G(x, y) \leq D_G(x, z) + D_G(z, z') + D_G(z', y) \leq D_G(z, z') + 4\gamma W_G$. Thus, $D_G(p^x, p^y) = D_G(z, z') \geq D_G(x, y) - 4\gamma W_G > 20\gamma W_G - 4\gamma W_G = 16\gamma W_G$.

Let now $p^{x\prime}$ be the remaining sub-path of q_x, starting from x', and terminating to r, and define $p^{y\prime}$ similarly. Let p^{xy} be the path from x' to y', obtained by concatenating $p^{x\prime}$, and $p^{y\prime}$.

By the construction of $\mathcal{K}_G$ it follows that if we remove from G all the vertices in the sets $K_1, K_3, \ldots, K_{\tau-1}$, then x and y remain in the same connected component. In other words, we can pick a path p^{yx} from x to y, that does not visit any of the vertices in $\bigcup_{j=1}^{\tau-1} K_j$. It follows that the distance between any vertex of p^{yx}, and any vertex in $\bigcup_{j=1}^{\tau-2} K_j$, is greater than γW_G. Thus, $D_G(p^{xy}, p^{yx}) > \gamma W_G$.

We have thus shown that there are vertices $x, y, y', x' \in V(G)$, and paths p^x, p^y, p^{xy}, p^{yx}, satisfying the conditions of Lemma 6. It follows that the optimal distortion required to embed G into a tree is greater than γ, a contradiction. $\square$

Using the bound on the diameter of the clusters in $\mathcal{K}_G$, we can show that for certain values of the parameters, the distances in the tree of clusters approximate the distances in the original graph.

Lemma 8. *Let $G = (V, E)$ be a graph that γ-embeds into a tree, let $r \in V(G)$, and let $(T_{\mathcal{K}}^G, \mathcal{K}_G)$ be a (r, γ)-tree-like decomposition of G. Then, for any $K_1, K_2 \in \mathcal{K}_G$, and for any $x_1 \in K_1$, $x_2 \in K_2$, $(D_{T_{\mathcal{K}}^G}(K_1, K_2) - 2)W_G\gamma \leq D_G(x_1, x_2) \leq (D_{T_{\mathcal{K}}^G}(K_1, K_2) + 2)20W_G\gamma$.*

Proof. Let $\delta = D_{T_{\mathcal{K}}^G}(K_1, K_2)$. We begin by showing the first inequality. We have to consider the following cases:

Case 1: *K_1 and K_2 are on the same path from the root to a leaf of $T^G_{\mathcal{K}}$.* Let the path between K_1 and K_2 in $T^G_{\mathcal{K}}$ be $K_1, H_1, H_2, \ldots, H_{\delta-1}, K_2$. Assume that the assertion is not true. That is, $D_G(x_1, x_2) < (\delta - 2)W_G\gamma$. Thus, $D_G(r, x_2) \leq D_G(r, x_1) + D_G(x_1, x_2) < D_G(r, x_1) + (\delta - 1)W_G\gamma$. Assume that $r \in K_r$, for some $K_r \in \mathcal{K}_G$, and w.l.o.g. that K_1 is an ancestor of K_2 in $T^G_{\mathcal{K}}$. Let the distance between K_r and K_1 in $T^G_{\mathcal{K}}$ be k. Then, the distance between K_r and K_2 is at most $k' = k + D_G(x_1, x_2)/(W_G\gamma)$. This implies that $\delta = k' - k < \delta - 1$, a contradiction.

Case 2: *K_1 and K_2 are not on the same path from the root to a leaf of $T^G_{\mathcal{K}}$.* Let K_a be the nearest common ancestor of K_1 and K_2 in $T^G_{\mathcal{K}}$. Observe that any path from x to y in G passes through K_a. Thus, we have $D_G(x, y) \geq D_G(K_x, K_a) + D_G(K_a, K_y)$. Let δ_i, for $i \in \{1, 2\}$ be the distance between K_a and K_i in $T^G_{\mathcal{K}}$. Then, by an argument similar to the above, we obtain that $D_G(K_x, K_a) \geq (\delta_1 - 1)W_G\gamma$, and also $D_G(K_y, K_a) \geq (\delta_2 - 1)W_G\gamma$. Since K_a is the nearest common ancestor of K_1 and K_2, it follows that K_a separates K_1 from K_2 in G. Thus, $D_G(x, y) \geq D_G(K_x, K_y) \geq D_G(K_x, K_a) + D_G(K_y, K_a) \geq (\delta - 2)W_Gc$.

We now show the second inequality. Consider an edge $\{K, K'\}$ of $T^G_{\mathcal{K}}$. Since K and K' are connected in $T^G_{\mathcal{K}}$ it follows that there exists an edge in G between a vertex in K and a vertex in K'. Since the maximum edge weight of G is W_G, we obtain $D_G(K, K') \leq W_G$.

Since by Lemma 7, the diameter of each $K \in \mathcal{K}_G$ is at most $20W_G\gamma$, it follows that $D_G(x_1, x_2) \leq \delta W_G + (\delta + 1)20W_G\gamma < (\delta + 2)20W_G\gamma$. □

3.4 Approximation algorithm for embedding unweighted graphs

In this section we give a $O(1)$-approximation algorithm for the problem of embedding the shortest path metric of an unweighted graph into a tree. Informally, the algorithm works as follows. Let $G = (V, E)$ be an unweighted graph, such that G can be embedded into an unweighted tree with distortion c. At a first step, we compute a tree-like decomposition $(T_{\mathcal{K}}^G, \mathcal{K}_G)$ of G. For each cluster in $\mathcal{K}_G$ we embed the vertices of the cluster in a star. We then connect the starts to form a tree embedding of G by connecting stars that correspond to clusters that are adjacent in $T_{\mathcal{K}}^G$.

Formally, the algorithm can be described with the following steps.

Step 1. We pick $r \in V(G)$, and we compute a (r, c)-tree-like decomposition $(T_{\mathcal{K}}^G, \mathcal{K}_G)$ of G.

Step 2. We construct a tree T as follows. Let $\mathcal{K}_G = \{K_1, \ldots, K_t\}$. For each $i \in [t]$, we construct a star with center a new vertex ρ_i, and leaves the vertices in K_i. Next, for each edge $\{K_i, K_j\}$ in $T_{\mathcal{K}}^G$, we add an edge $\{\rho_i, \rho_j\}$ in T.

By proposition 1, we know that the resulting graph T is indeed a tree, so we can focus of bounding the distortion of T. By lemma 7, the diameter of each cluster in $\mathcal{K}_G$ is at most $20cW_G = 20c$. Let $x_1, x_2 \in V(G)$, with $x_1 \in K_1$, and $x_2 \in K_2$, for some $K_1, K_2 \in \mathcal{K}_G$. We have $D_T(x_1, x_2) = 2 + D_T(\rho_1, \rho_2) = 2 + D_{T_{\mathcal{K}}^G}(K_1, K_2)$. By lemma 8 we obtain that $D_T(x_1, x_2) \leq 4 + D_G(x_1, x_2)/c \leq 5D_G(x_1, x_2)$. Also by the same lemma, $D_T(x_1, x_2) \geq D_G(x_1, x_2)/(20c)$. By combining the above it follows that the distortion is at most $100c$.

Theorem 2. *There exists a polynomial time, constant-factor approximation algorithm, for the problem of embedding an unweighted graph into a tree, with minimum multiplicative distortion.*

3.5 Well-separated tree-like decompositions

Before we describe our algorithm for embeddings general metrics, we need to introduce a refined decomposition procedure. As in the unweighted case, we want to obtain a partition of the input metric space in a set of clusters, solve the problem independently for each cluster, and join the solutions to obtain a solution for the input metric.

The key properties of the tree-like decomposition used in the case of unweighted graphs are the following: (1) the distances in the tree of clusters approximate the distances in the original graph, and (2) the diameter of each cluster is small.

Observe that if the graph is weighted with maximum edge weight W_G, and the clusters have small diameter, then the distance between two adjacent clusters of a tree-like decomposition can be any value between 1 and W_G. Thus, the tree of clusters cannot approximate the original distances by a factor better than W_G.

We address this problem by introducing a new decomposition that allows the diameter of each cluster to be arbitrary large, while guaranteeing that (1) the distance between clusters is sufficiently large, and (2) after solving the problem independently for each cluster, the solutions can be merged together to obtain a solution for the input metric.

Formally, let $G = (V, E)$ be a graph that γ-embeds into a tree. Let also $r \in V(G)$, and $\alpha \geq 1$ be a parameter. Intuitively, the parameter α controls the distance between clusters in the resulting partition.

A (r, γ, α)-*well-separated tree-like decomposition* is a triple $(T^G_{\mathcal{K}}, \mathcal{K}_G, \mathcal{A}_G)$, were $(T^G_{\mathcal{K}}, \mathcal{K}_G)$ is a (r, γ)-tree-like decomposition of G, and $\mathcal{A}_G$ is defined as follows.

For a set $A \subseteq V(G)$, let $Z_A = \{K \in \mathcal{K}_G | K \cap A \neq \emptyset\}$. Define $T^{G,A}_{\mathcal{K}}$ to be the vertex-induced subgraph $T^G_{\mathcal{K}}[Z_A]$.

Proposition 2. *Let $A \subseteq V(G)$, such that $G[A]$ is connected. Then, $T^{G,A}_{\mathcal{K}}$ is a subtree of $T^G_{\mathcal{K}}$*

Proof. Since $G[A]$ is connected, it suffices to show that any edge e of G is either contained in some $K \in \mathcal{K}_G$, or the end-points of e are contained in sets $K, K' \in \mathcal{K}_G$, such that there is an edge between K and K' in $T^G_{\mathcal{K}}$. Assume that this is not true, and pick an edge $\{v_1, v_2\} \in E(G)$, with $v_1 \in K_1$, and $v_2 \in K_2$, for some $K_1, K_2 \in \mathcal{K}_G$, such that there is no edge between K_1 and K_2 in $T^G_{\mathcal{K}}$.

Let $K_r \in \mathcal{K}_G$ be such that $r \in K_r$. Assume first that K_1 is on the path from K_2 to $K_r \in \mathcal{K}_G$ in $T^G_{\mathcal{K}}$. This implies however that $D(v_1, v_2) > W_G$, contradicting the fact that $\{v_1, v_2\} \in E(G)$.

It remains to consider the case where K_1 is not in the path from K_2 to K_r, and K_2 is not in the path from K_1 to K_r in $T^G_{\mathcal{K}}$. Then by the construction of $\mathcal{K}_G$ we know that any path from a vertex in K_1 to a vertex in K_2 in G has to pass through an ancestor of K_1, and K_2. Thus, there is not edge between K_1 and K_2 in G, a contradiction. $\square$

$\mathcal{A}_G$ is computed in two steps:

Step 1. We define a partition $\bar{\mathcal{A}}_G$. $\bar{\mathcal{A}}_G$ contains all the connected components of G obtained after removing all the edges of weight greater than $W_G/(\gamma^{3/2}\alpha)$.

Step 2. We set $\mathcal{A}_G := \bar{\mathcal{A}}_G$. While there exist $A_1, A_2 \in \mathcal{A}_G$ such that the diameter of $T_{\mathcal{K}}^{G,A_1} \cap T_{\mathcal{K}}^{G,A_2}$ is greater than 50γ, we remove A_1, and A_2 from $\mathcal{A}_G$, and we add $A_1 \cup A_2$ in $\mathcal{A}_G$. We repeat until there are no more such pairs A_1, A_2.

3.5.1 Properties of well-separated tree-like decompositions

We now show the main properties of a well-separated tree-like decomposition that will be used by our algorithm for embedding general metrics. They are summarized in the following two lemmas.

Intuitively, the first lemma shows that the distance between different clusters is sufficiently large, and at the same time they don't share long parts of the tree $T_{\mathcal{K}}^G$. The technical importance of the later property will be justified in the next section. It is worth mentioning however that intuitively, the fact that the intersections are short will allow us to arrange the clusters of $\mathcal{A}_G$ in a tree, without intersections, incurring only a small distortion.

Lemma 9. *For any $A_1, A_2 \in \mathcal{A}_G$, $D_G(A_1, A_2) \geq W_G/(\gamma^{3/2}\alpha)$, and $T_{\mathcal{K}}^{G,A_1} \cap T_{\mathcal{K}}^{G,A_2}$ is a subtree of $T_{\mathcal{K}}^G$ with diameter at most 50γ.*

Proof. For any $A_1, A_2 \in \bar{\mathcal{A}}_G$, we have that $D(A_1, A_2) \geq W_G/(\gamma^{3/2}\alpha)$. Since $\mathcal{A}_G$ is obtained by only merging sets, the first property holds. Moreover, the construction of $\mathcal{A}_G$ clearly terminates, and the second property follows by the termination condition of the construction procedure. $\square$

The next lemma will be used to argue that when recursing in a cluster, the corresponding induced metric can be sufficiently approximated by a graph with small maximum edge weight.

Lemma 10. *For any $A \in \mathcal{A}_G$, the $W_G/(\gamma^{1/2}\alpha)$-restricted subgraph of $G[A]$, is connected.*

Proof. For an embedding of G into a tree T, and for disjoint $A_1, A_2 \subset V(G)$, we say that A_1 *splits* A_2 *in* T, if A_2 intersects at least 2 connected components of $T[V(G)\backslash A_1]$.

Claim 8. *Let $A_1, A_2 \subset V(G)$, with $A_1 \cap A_2 = \emptyset$, such that $G[A_1]$, and $G[A_2]$ are both connected. Assume that the diameter of $T_{\mathcal{K}}^{G,A_1} \cap T_{\mathcal{K}}^{G,A_2}$ is greater than 50γ. Consider an optimal non-contracting embedding of G into a tree T, with distortion γ. Then, either A_1 splits A_2 in T, or A_2 splits A_1 in T.*

Proof. Since $G[A_1]$, and $G[A_2]$ are both connected, it follows by Proposition 2 that $T_{\mathcal{K}}^{G,A_1}$, and $T_{\mathcal{K}}^{G,A_2}$ are both connected subtrees of $T_{\mathcal{K}}^{G}$. Pick a path $p = K_1, K_2, \ldots, K_l$ in $T_{\mathcal{K}}^{G}$, with $l > 50\gamma$, that is contained in $T_{\mathcal{K}}^{G,A_1} \cap T_{\mathcal{K}}^{G,A_2}$.

Assume that the assertion is not true. Let $A_1' = A_1 \cap (\bigcup_{i=1}^{l} K_i)$, and let $A_2' = A_2 \cap (\bigcup_{i=1}^{l} K_i)$. Let T_1 be the minimum connected subtree of T that contains A_1', and similarly let T_2 be the minimum connected subtree of T that contains A_2'. It follows that $T_1 \cap T_2 = \emptyset$.

Let x_1 be the unique vertex of T_1 which is closest to T_2. Since T_1 is minimal, x_1 disconnects T_1. Moreover, since $G[A_1]$ is connected, it follows that there exists $\{w, w'\} \in E(G)$, such that the path from w to w' in T passes through x_1. Since $D_G(w, w') \leq W_G$, we obtain that there exists $x_1^* \in \{w, w'\}$, with $D_T(x_1^*, x_1) \leq D_T(w, w')/2 \leq \gamma D_G(w, w')/2 \leq \gamma W_G/2$.

By Lemma 7, it follows that for any $x \in A_1'$, there exists $x' \in A_2'$, such that $D_G(x, x') \leq 20 W_G \gamma$. Moreover, for any $x \in A_1'$, $D_T(x, T_2) = D_T(x, x_1) + D_T(x_1, T_2)$. Thus, for any $x \in A_1'$, $D_T(x, x_1^*) \leq D_T(x_1, x_1^*) + D_T(x, x_1) \leq \gamma W_G/2 + D_T(x, T_2) \leq \gamma W_G/2 + \gamma D_G(x, A_2') \leq 21 W_G \gamma^2$.

Pick $z \in A_1' \cap K_1$, and $z' \in A_1' \cap K_l$. By the triangle inequality, $D_T(z, z') \leq D_T(z, x_1^*) + D_T(x_1^*, z') \leq 42 W_G \gamma^2$. On the other hand, the distance between K_1, and K_l in $T_{\mathcal{K}}^{G}$ is $l-1$. Thus, by Lemma 8 we obtain that $D_G(z, z') \geq (l-3) W_G \gamma > 45 W_G \gamma^2$, which contradicts that fact that the embedding of M into T is non-contracting. ☐

Fix an optimal non-contracting embedding of G into a tree T, with distortion γ.

For $k \geq 0$, let $\mathcal{A}_G^k$ be the partition $\mathcal{A}_G$ after k iterations of Step 2 have been performed, with $\mathcal{A}_G^0 = \bar{\mathcal{A}}_G$.

Assume that the assertion is not true, and pick the smallest k, such that there exists $A \in \mathcal{A}_G^k$, such that the $W_G/(\gamma^{1/2}\alpha)$-restricted subgraph of $G[A]$ is not connected. Assume that A is obtained by joining $A_1, A_2 \in \mathcal{A}_G^{k-1}$. By the minimality of k, it follows that the $W_G/(\gamma^{1/2}\alpha)$-restricted subgraphs of $G[A_1]$, and $G[A_2]$ respectively are connected. Thus, $D_G(A_1, A_2) > W_G/(\gamma^{1/2}\alpha)$.

By claim 8, we can assume w.l.o.g. that A_2 splits A_1. Thus, by removing A_2 from T, we obtain a collection of connected components F_1. Consider the partition F_1' of A_1 defined by restricting F_1 on A_1. Formally, $F_1' = \{f \cap A_1 | f \in F_1, f \cap A_1 \neq \emptyset\}$. We have to consider the following cases:

Case 1: *There exists $Z \in \bar{\mathcal{A}}_G$, with $Z \subseteq A_1$, such that Z intersects at least two sets in F_1'.* By considering only edges of weight at most $W_G/(\gamma^{3/2}\alpha)$, the induced subgraph $G[Z]$ is connected. It follows that there exist $z_1, z_2 \in Z$, with $D_G(z_1, z_2) \leq W_G/(\gamma^{3/2}\alpha)$, such that the path from z_1 to z_2 in T passes through A_2. Thus, $D_T(z_1, z_2) \geq 2D_G(A_1, A_2) > 2W_G/(\gamma^{1/2}\alpha) \geq 2\gamma D(z_1, z_2)$, contradicting the fact that the expansion of T is at most γ.

Case 2: *For any $Z \in \bar{\mathcal{A}}_G$, with $Z \subseteq A_1$, we have $Z \subseteq Z'$, for some $Z' \in F_1'$.* Observe that for ant $t \geq 0$, any element in $\mathcal{A}_G^t$ is obtained as the union of elements of $\bar{\mathcal{A}}_G$. Thus, we can pick the minimum $j \geq 1$, such that there exist $B_1, B_2 \in \mathcal{A}_G^{j-1}$, such that during iteration j of Step 2, the set $B = B_1 \cup B_2$ is obtained, with $B \subseteq A_1$, and such that $B_1 \subseteq Z_1'$, and $B_2 \subseteq Z_2'$, for some $Z_1', Z_2' \in F_1'$, In other words, we pick the minimum j such that we can find sets $B_1, B_2 \in \mathcal{A}_G^{j-1}$, that are contained in A_2, and neither of them is split by A_2 in T. W.l.o.g., we can assume that B_2 splits B_1 in T. Thus, there exist $C_1, C_2 \subseteq B_1$, such that any path between C_1 and C_2 in T passes through B_2. Moreover, any path from B_1 to B_2 in T passes through A_2. Thus, any path from C_1 to C_2 in T passes through A_2. This however contradicts the minimality of j. The scenario is depicted in Fig 3-2. $\square$

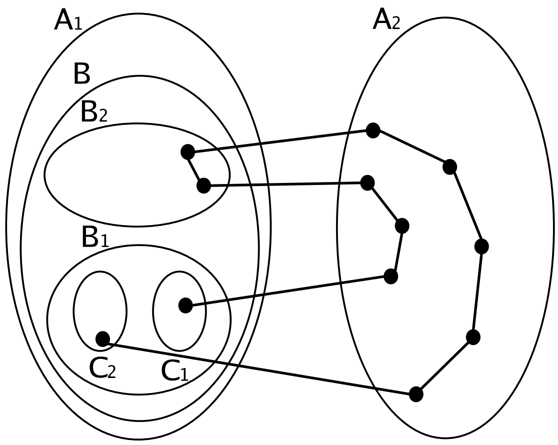


Figure 3-2: Case 2 of the proof of Lemma 10.

3.6 Approximation algorithm for embedding general metrics

In this section we present an approximation algorithm for embedding general metrics into trees. Before we get into the technical details of the algorithm, we give an informal description. The main idea is to partition the input metric M using a well-separated tree-like decomposition, and then solve the problem independently for each cluster of the partition by recursion. After solving all the sub-problems, we can combine the partial solutions to obtain a solution for M. There are a few points that need to be highlighted:

Termination of the recursion. As pointed out in the description of the well-separated tree-like decompositions, the clusters of the resulting partition might have arbitrarily long diameter. In particular, we cannot guarantee that by recursively decomposing each cluster we obtain sub-clusters of smaller diameter. To that extend, our recursion deviates from standard techniques since the sub-problems are not necessarily smaller in a usual sense. Instead, our decomposition procedure guarantees that at each recursive step, the metric of each cluster can be approximated by a graph with smaller maximum edge length. This can be thought as restricting the problem to a smaller metric scale.

Merging the partial solutions. The partial solution for each cluster in the recursion is an embedding of the cluster into a tree. As in the algorithm for unweighted graphs, we merge the partial solutions using the tree $T_{\mathcal{K}}^G$ of the well-separated tree-like decomposition as a rough approximation of the resulting tree. However, in the case of a well-separated decomposition, the parts of $T_{\mathcal{K}}^G$ that correspond to different clusters of the partition $\mathcal{A}_G$ might overlap. Moreover, since some of the clusters might be long, we need to develop an elaborate procedure for merging the different trees into a tree for M, without incurring large distortion.

3.6.1 The main inductive step

We will now describe the main inductive step of the algorithm. Let $M = (X, D)$ be a finite metric that c-embeds into a tree. At each recursive step performed on a cluster A^* of M, the algorithm is given a graph G with vertex set A, that c-approximates M. In order to recurse in sub-problems, we compute a well-separated tree-like decomposition of G. We chose the parameters of the well-separated decomposition so that each sub-cluster A, can be c-approximated by a graph that has maximum edge weight significantly smaller than the maximum edge weight of G. Formally, the main recursive step is as follows.

Procedure RecursiveTree

Input: A graph G with maximum edge weight W_G, that c-approximates M.

Output: An embedding of G into a tree S.

Step 1: Partitioning. If G contains only one vertex, then we output a trivial tree containing only this vertex. Otherwise, we proceed as follows. We pick $r \in V(G)$, and compute a (r, c^2, α)-well-separated tree-like decomposition $(T_{\mathcal{K}}^G, \mathcal{K}_G, \mathcal{A}_G)$ of G, where $\alpha > 0$ will be determined later.

Step 2: Recursion. For any $A \in \mathcal{A}_G$, let G_A be the W_G/α-restricted subgraph, with $V(G_A) = A$. We recursively execute the procedure RecursiveTree on G_A, and we obtain a tree S^A.

Step 3: Merging the solutions. In this final step we merge the trees S^A to obtain S.

We define a tree T as follows. We first remove from $T_{\mathcal{K}}^G$ all the edges between vertices at level $i50c^2$, and $i50c^2+1$, for any integer $i : 1 \leq i \leq n/(50c^2)$. For any connected component C of the resulting forest, T contains a vertex C. Two vertices $C, C' \in V(T)$ are connected, iff there is an edge between C, and C' in $T_{\mathcal{K}}^G$. We consider T to be rooted at the vertex which corresponds to the subtree of $T_{\mathcal{K}}^G$ that contains r. Furthermore, for each $A_i \in \mathcal{A}_G$, we define a subtree T_i of T as follows: T_i contains all the vertices C of T, such that $T_{\mathcal{K}}^{G,A_i}$ visits C. We will use the following Lemma to connect all the T_is in a larger tree:

Lemma 11. *There exists a polynomial-time algorithm that computes an unweighted tree T', and for any $i \in [k]$ a mapping $\phi_i : V(T_i) \to V(T')$, such that*

- *for any $i, j \in [k]$, $\phi_i(T_i) \cap \phi_j(T_j) = \emptyset$,*
- *for any $i, j \in [k]$, for any $v_i \in V(T_i)$, and $v_j \in V(T_j)$, $D_T(v_i, v_j) \leq D_{T'}(\phi_i(v_i), \phi_j(v_j)) \leq 20(D_T(v_i, v_j) + 1) \log n$.*

The proof of the above Lemma is given in the following Section. Note that the tree T' might contain vertices $C \in V(T)$, such that for any $K \in \mathcal{K}_G$, $K \notin C$. We call such a vertex *steiner*. First, for each steiner vertex $C \in V(T')$ we add a vertex $v_C \in V(S)$. We have to add the following types of edges:

- For any $C_1, C_2 \in V(T')$, such that both C_1, and C_2 are steiner vertices, we add the edge $\{v_{C_1}, v_{C_2}\}$ in S, with weight $W_G/(c^3\alpha)$.
- For any $C_1, C_2 \in V(T')$, such that C_2, is a steiner vertex, and there exists $A_1 \in \mathcal{A}_G$, such that $C_1 \in \phi_1(T_1)$, we pick $K_1 \in T_{\mathcal{K}}^{G,A_1}$, with $K_1 \in C_1$, and an arbitrary $x_1 \in K_1$, and we add the edge $\{x_1, v_{C_1}\}$ in S. The weight of this new edge is $W_G/(c^3\alpha)$.
- For any pair $A_1, A_2 \in \mathcal{A}_G$, with $A_1 \neq A_2$, such that there exists an edge in T' connecting $\phi_1(T_1)$ with $\phi_2(T_2)$, we add an edge between S^{A_1}, and S^{A_2}.

We pick the edge that connects S^{A_1} with S^{A_2} as follows. Pick $C_1, C_2 \in V(T)$, with $C_1 \in T_1$, and $C_2 \in T_2$, such that there is an edge between $\phi_1(C_1)$, and $\phi_2(C_2)$ in T'. We pick an arbitrary pair of points x_1, x_2, with $x_1 \in K_1 \in C_1$, and $x_2 \in K_2 \in C_2$, for some $K_1, K_2 \in \mathcal{K}_G$, and we connect S^{A_1} with S^{A_2} by adding the edge $\{x_1, x_2\}$ of length $D(x_1, x_2)$.

Given the metric $M = (X, D)$, the algorithm first computes a weighted complete graph $G_0 = (V, E)$, with $V(G_0) = X$, such that the weight of each edge $\{u, v\} \in E(G)$ is equal to $D(u, v)$. Let Δ be the diameter of M. Clearly, G_0 is a Δ-restricted subgraph. The algorithm then executes the procedure RecursiveTree on G_0, and outputs the resulting tree S.

Before we bound the distortion of the resulting embedding, we first need to show that at each recursive call of the procedure RecursiveTree, the graph G satisfies the input requirements. Namely, we have to show that G c-approximates M. Clearly, this holds for G_0. Thus, it suffices to show that the property is maintained for each graph G_A, were $A \in \mathcal{A}_G$. Observe that since G c-approximates M, and M c-embeds into a tree, it follows that G c^2-embeds into a tree. Since $(T_{\mathcal{K}}^G, \mathcal{K}_G, \mathcal{A}_G)$ is a (r, c^2)-well-separated decomposition, we can assume the properties of lemmas 9, and 10, for $\gamma = c^2$.

Lemma 12. *For any $A \in \mathcal{A}_G$, G_A c-approximates M.*

Proof. The next claim is similar to a lemma given in [18], modified for the case of embedding into trees.

Claim 9. *Let $\alpha > 0$. Let G be an α-restricted subgraph of M, and let G' be an αc-restricted subgraph of M, with $V(G) = V(G')$. If G is connected, then for any $u, v \in V(G)$, $D(u, v) \leq D_{G'}(u, v) \leq cD(u, v)$.*

Proof. Let M' be the restriction of M on $V(G)$. Consider a non-contracting embedding of M' into a tree T' with distortion at most c. Consider an edge $\{u, v\} \in E(T')$. We will first show that $D(u, v) \leq \alpha c$. Let S be a minimum spanning tree of G. If $\{u, v\} \in E(S)$, then since G is connected, it follows that $D(u, v) \leq \alpha$. Assume now that $\{u, v\} \notin E(S)$. Let T_u and T_v be the two subtrees of T', obtained after removing the edge $\{u, v\}$, and assume that T_u contains u, and T_v contains v. Let $p = x_1, \ldots, x_{|p|}$ be the unique path in S with $u = x_1$, and $v = x_{|p|}$. Observe that the sequence of vertices visited by p start from a vertex in T_v, and terminate at a vertex in T_u. Thus, there exists $i \in [|p| - 1]$, such that $v_i \in T_v$, while $v_{i+1} \in T_u$. It follows that the edge $\{u, v\}$ lies in the path from v_i to v_{i+1} in T', and thus $D_{T'}(u, v) \leq D_{T'}(v_i, v_{i+1})$. Since $\{v_i, v_{i+1}\}$ is an edge of S, we have by the above argument that $D(v_i, v_{i+1}) \leq \alpha$. Since the embedding in T has expansion at most c, it follows that $D_{T'}(v_i, v_{i+1}) \leq \alpha c$. Thus, $D_{T'}(u, v) \leq \alpha c$.

Consider now some pair $x, y \in V(G)$. If no vertex is embedded between x and y, then by the above argument, $D(x, y) \leq \alpha c$, and thus the edge $\{x, y\}$ is in G' and $D_{G'}(x, y) = D(x, y)$. Otherwise, let $z_1, \ldots, z_k$ be the vertices appearing in T' between x and y (in this order). Then the edges $\{x, z_1\}, \{z_1, z_2\}, \ldots, \{z_{k-1}, z_k\}, \{z_k, y\}$ all belong to G', and therefore

$$\begin{aligned}
D_{G'}(x, y) &\leq D_{G'}(x, z_1) + D_{G'}(z_1, z_2) + \ldots D_{G'}(z_{k-1}, z_k) + D_{G'}(z_k, y) \\
&= D(x, z_1) + D(z_1, z_2) + \ldots D(z_{k-1}, z_k) + D(z_k, y) \\
&\leq D_{T'}(x, z_1) + D_{T'}(z_1, z_2) + \ldots + D_{T'}(z_{k-1}, z_k) + D_{T'}(z_k, y) \\
&= D_{T'}(x, y) \leq cD(x, y)
\end{aligned}$$

□

By the construction of the set $\mathcal{A}_G$, it follows that a W_G/c^2-restricted subgraph with vertex set A, is connected. Thus, by claim 9, D_{G_A} c-approximates D. □

The next two lemmas bound the distortion of the resulting embedding of G into S. The fact that the contraction is small follows by the fact that the distance between the clusters in $\mathcal{A}_G$ is sufficiently large. The expansion on the other hand, depends on the maximum depth of the recursion. This is because at each recursive call, when we merge the trees S^A to obtain S, we incur an extra $c^{O(1)} \log n$-factor in the distortion. Since at every recursive call the maximum edge weight of the input graph decreases by a factor of α, the parameter α can be used to adjust the recursion depth in order to optimize the final distortion.

Lemma 13. *The contraction of S is $O(c^7\alpha)$.*

Proof. In order to bound the contraction of S, it is sufficient to bound the contraction between pairs of vertices $x_1, x_2 \in V(G)$, such that either $\{x_1, x_2\} \in S$, or between x_1 and x_2 there are only steiner nodes in S.

We will prove the assertion by induction on the recursive steps of the algorithm. Consider an execution of the recursive procedure RecursiveTree, with input a graph G with maximum edge weight W_G. If G contains only one vertex, then assertion is trivially true. Otherwise, assume that all the recursively computed trees S^A satisfy the assertion.

Consider such a pair $x_1, x_2 \in V(G)$, and assume that in the path from x_1 to x_2 in S, there are $k \geq 0$ steiner nodes. If there exists $A \in \mathcal{A}_G$, such that $x_1, x_2 \in A$, then the assertion follows by the inductive hypothesis.

Assume now that there exist $A_1, A_2 \in \mathcal{A}_G$, with $A_1 \neq A_2$, such that $x_1 \in A_1$, and $x_2 \in A_2$. It follows that $D_S(x_1, x_2) = (k+1)W_G/(c^3\alpha)$. Pick $C_1, C_2 \in V(T)$, and $K_1, K_2 \in \mathcal{K}_G$, such that $x_1 \in K_1 \in C_1$, and $x_2 \in K_2 \in C_2$. We have $D_{T'}(\phi_1(C_1), \phi_2(C_2)) = k+1$. By Lemma 11, we obtain $D_T(C_1, C_2) \leq k+1$. Thus, $D_{T^G_{\mathcal{K}}}(K_1, K_2) \leq (k+2)50c^2$. By Lemma 8, $D(x_1, x_2) \leq ((k+2)50c^2+2)W_G c^2$. Thus, the contraction on x_1, x_2 is $\frac{D_S(x_1,x_2)}{D(x_1,x_2)} \leq \frac{((k+2)50c^2+2)W_G c^2}{(k+1)W_G/(c^3\alpha)} < 104c^7\alpha$. □

Lemma 14. *The expansion of S is at most $(c^{O(1)} \log n)^{\log_\alpha \Delta}$.*

Proof. We will prove the assertion by induction on the recursive steps of the algorithm.

Consider an execution of the recursive procedure RecursiveTree, with input a graph G with maximum edge weight W_G. If G contains only one vertex, then the expansion of the computed tree is at most 1. Otherwise, at Step 2 we partition $V(G)$ into $\mathcal{A}_G$, and at Step 3, for each $A \in \mathcal{A}_G$ we define the graph G_A, and recursively execute RecursiveTree on G_A, obtaining an embedding of G_A into a tree S^A. Assume that for each $A \in \mathcal{A}_G$, the expansion on S^A is at most ξ.

Consider $x, y \in V(G)$. Assume that $x \in A_{i_x}$, and $y \in A_{i_y}$, for some $A_{i_x}, A_{i_y} \in \mathcal{A}_G$. If $A_{i_x} = A_{i_y}$, then the expansion is at most ξ, be the inductive hypothesis. We can thus assume that $A_{i_x} \neq A_{i_y}$. Pick $K_x, K_y \in \mathcal{K}_G$, and $C_x, C_y \in V(T)$, such that $x \in K_x \in C_x$, and $y \in K_y \in C_y$. Let p be the path between $\phi_{i_x}(C_x)$, and $\phi_{i_y}(C_y)$ in T'.

Let also q be the path from x to y in S. Assume that q visits the sets in $\mathcal{A}_G$ in the order $A_{t_1}, A_{t_2}, \ldots, A_{t_k}$. Let v_i, and v'_i be the first and the last respectively vertex of A_{t_i} visited by q Similarly, let $\phi_{j_i}(C_i)$, $\phi_{j_i}(C'_i)$ and be the first, and the last respectively vertex of $\phi_{j_i}(T_{j_i})$ visited by p. For each $j \in [k]$, pick $K_i, K'_i \in \mathcal{K}_G$, such that $v_i \in K_i$, and $v'_i \in K'_i$.

Let $\delta = W_G/(c^3\alpha)$. We have:

$$
\begin{aligned}
D_S(x,y) &= \sum_{j=1}^{k} D_S(v_j, v'_j) + \sum_{j=1}^{k-1} D_S(v_{j'}, v_{j+1}) \\
&\leq \xi \sum_{j=1}^{k} D(v_j, v'_j) + \delta \sum_{j=1}^{k-1} D_{T'}(\phi_{j_i}(C'_i), \phi_{j_{i+1}}(C_{i+1})) \\
&\leq \xi W_G c^2 \sum_{j=1}^{k} (2 + D_{T^G_{\mathcal{K}}}(K_j, K'_j)) + 20\delta \log n \sum_{j=1}^{k-1} (1 + D_T(C'_i, C_{i+1})) \\
&\leq \xi W_G c^2 \sum_{j=1}^{k} (2 + 100c^2 D_T(C_j, C'_j)) + 20\delta \log n \sum_{j=1}^{k-1} (1 + D_T(C'_i, C_{i+1})) \\
&\leq (102\xi W_G c^4 + 40\delta \log n) D_T(C_x, C_y) \\
&\leq (102\xi W_G c^4 + 40\delta \log n) D_{T^G_{\mathcal{K}}}(K_x, K_y) \\
&\leq (102\xi W_G c^4 + \frac{40 W_G \log n}{c^3 \alpha})(\frac{D(x,y)}{W_G c} + 2)
\end{aligned}
$$

Since $A_{i_x} \neq A_{i_y}$, it follows that $D(x,y) \geq \delta = W_G/(c^3\alpha)$. Thus,

$$\begin{aligned} D_S(x,y) &\leq (102\xi W_G c^4 + \frac{40W_G \log n}{c^3\alpha})(\frac{D(x,y)}{W_G c} + 2c^3\alpha\frac{D(x,y)}{W_G}) \\ &\leq (102\xi c^4 + \frac{40\log n}{c^3\alpha})3c^3\alpha D(x,y) \\ &\leq (306\xi c^7\alpha + 120\log n)D(x,y) \end{aligned}$$

Given a graph of maximum edge weight W_G, the procedure RecursiveTree might perform recursive calls on graphs with maximum edge weight $c^3\delta = W_G/\alpha$. Since the minimum distance in M is 1, and the spread of M is Δ, it follows that the maximum number of recursive calls can be at most $\log\Delta/\log\alpha$. Thus,

$$D_S(x,y) \leq (c^{O(1)}\log n)^{\log_\alpha \Delta} D(x,y)$$

$\square$

Theorem 3. *There exists a polynomial-time algorithm which given a metric $M = (X,D)$ that c-embeds into a tree, computes an embedding of M into a tree, with distortion $(c\log n)^{O(\sqrt{\log\Delta})}$.*

Proof. By Lemmata 13, and 14, it follows that the distortion of S is $c^{O(1)}\alpha(c^{O(1)}\log n)^{\log_\alpha \Delta}$. By setting $\alpha = 2^{\sqrt{\log\Delta}}$, we obtain that the distortion is at most $(c\log n)^{O(\sqrt{\log\Delta})}$. $\square$

3.6.2 Proof of lemma 11

In this section we give the proof of Lemma 11.

Claim 10. *For any $A_i, A_j \in \mathcal{A}_G$, with $A_i \neq A_j$, either $T_i \cap T_j = \emptyset$, or there exists $v \in V(T)$, and $v_1,\ldots,v_l$, for some $l \geq 0$, such that $v_1,\ldots,v_l$ are children of v, and $T_i \cap T_j = \{v, v_1,\ldots,v_l\}$.*

Proof. It follows immediately from the fact that for any $A_i, A_j \in \mathcal{A}_G$, the diameter of $T_{\mathcal{K}}^{G,A_i} \cap T_{\mathcal{K}}^{G,A_j}$ is at most $50c^2$. $\square$

Let r be the root of T. Initially, T' contains a single vertex r'. To simplify the discussion, we assume w.l.o.g., that r is a leaf vertex of T. We also assume that for every edge $\{u, v\} \in E(T)$, there is a tree T_i that contains $\{u, v\}$. This is because if there is no such tree, then we can simply introduce a new subtree T_i, that contains only the vertices u, and v.

For every T_i that visits r, we introduce in T' a copy $\phi_i(T_i)$ of T_i, and we connect $\phi_i(r)$ to r'.

We proceed by visiting the vertices of T in a top-down fashion. Assume that we are visiting a vertex $v \in V(T)$, with parent $p(v)$, and children $v_1, \ldots, v_t$. At this step, we are going to introduce in T' a copy $\phi_i(T_i)$ of T_i, for every T_i that visits v, and we have not considered yet. We consider the following cases:

Case 1: There is no T_i that visits v, and $p(v)$.

Let T_a be a subtree that visits $p(v)$. For every T_b that visits v, and we have not considered yet, we introduce in T' a copy $\phi_b(T_b)$ of T_b, and we connect $\phi_b(v)$ to $\phi_a(p(v))$.

Case 2: There exists T_i that visits v, and $p(p(v))$, and there is no $j \neq i$, such that T_j visits v, and $p(v)$.

For every T_b that visits v, and we have not considered yet, we introduce in T' a copy $\phi_b(T_b)$ of T_b, and we connect $\phi_b(v)$ to $\phi_i(v)$.

Case 3: There is no T_i that visits v, and $p(p(v))$, and there exists T_j that visits v, and $p(v)$.

Let $a \in [k]$ be the minimum integer such that T_a visits v, and $p(v)$. For every T_b that visits v, and we have not considered yet, we introduce in T' a copy $\phi_b(T_b)$ of T_b, and we connect $\phi_b(v)$ to $\phi_a(v)$.

Case 4: There exists T_i that visits v, and $p(p(v))$, and there exists T_j, with $i \neq j$, that visits v, and $p(v)$.

Let $a \in [k]$ be the minimum integer with $a \neq i$, such that T_a visits v, and $p(v)$. For every T_b that visits v, and we have not considered yet, we introduce in T'

a copy $\phi_b(T_b)$ of T_b. With probability 1/2, we connect $\phi_b(v)$ to $\phi_i(v)$, and with probability 1/2, we connect $\phi_b(v)$ to $\phi_a(v)$.

Claim 11. *T' is a tree.*

Proof. T' is a forest since each $\phi_i(T_i)$ is a tree, and also each $\phi_i(T_i)$ is connected to exactly one $\phi_j(T_j)$, such that T_j was considered before i. Also, T' is connected since every vertex of T is contained in some subtree T_t. □

Claim 12. *For any $v \in V(T)$, there exists at most one $i \in [k]$, such that T_i visits both v, and $p(p(v))$.*

Proof. Assume that the assertion is not true. Let T_i, T_j be subtrees that visit both v, and $p(p(v))$. Then, T_i and T_j also visit $p(v)$. This however contradicts the definition of the subtrees $T_1, \ldots, T_k$. □

Claim 13. *Let $i, j \in [k]$, with $i \neq j$, be such that T_i, and T_j both visit a vertex $v \in V(T)$, but they do not visit $p(v)$. Then, with probability at least 1/2, there exists $t \in [k]$, such that T_t visits v, and $p(v)$, and both $\phi_i(v)$, and $\phi_j(v)$ are connected to $\phi_t(v)$.*

Proof. Recall the procedure for constructing T', described above. Consider the step in which we add to T' the subtrees that visit the vertex v, and v is their highest vertex in T. Clearly T_i, and T_j are both in this set of subtrees. Observe that in cases 1, 2, and 3, the first event of the assertion happens with probability 1. This is because all the trees that we consider are connected to the same subtree.

In the remaining case 4, there are subtrees $T_{i'}$, $T_{j'}$ such that each subtree that we consider is going to be connected to $T_{i'}$ with probability 1/2, and to $T_{j'}$ with probability 1/2. Thus, with probability 1/2, T_i and T_j are going to be connected to the same subtree. □

Claim 14. *Let $i, j \in [k]$, with $i \neq j$, be such that T_i visits v, and does not visit $p(v)$, and T_j visits both v, and $p(v)$, for some $v \in V(T)$. Then, with probability at least 1/4, there exists $L \leq 4$, and $t(1), \ldots, t(L)$, such that*

- *$t(1) = i$, and $t(L) = j$,*
- *for each $l \in [L-1]$, $\phi_{t(l)}(T_{t(l)})$ is connected to $\phi_{t(l+1)}(T_{t(l+1)})$.*

Proof. We have to consider the following cases:

Case 1: T_j visits $p(p(v))$.

In this case, $\phi_i(v)$ is connected to $\phi_j(v)$ with probability at least $1/2$.

Case 2: T_j does not visit $p(p(v))$.

Let w be the smallest integer, such that T_w visits v, and $p(v)$, but does not visit $p(p(v))$. If $w = j$, then $\phi_i(v)$ is connected to $\phi_j(v)$ with probability at least $1/2$.

Otherwise, if $w \neq j$, then with probability at least $1/2$, $\phi_i(v)$ is connected to $\phi_w(v)$. Moreover, by Claim 13, with probability at least $1/2$, there exists $w' \in [k]$, such that both $\phi_w(p(v))$, and $\phi_j(p(v))$, are connected to $\phi_{w'}(p(v))$. Observe that the above two events are independent. Thus, with with probability at least $1/4$, the sequence of subtrees $T_i, T_w, T_{w'}, T_j$, satisfy the conditions of the assertion.

□

Claim 15. *Let T_i, T_j be two subtrees such that they both visit some vertex $v \in V(T)$. Then, with probability at least $1 - n^{-4}$, there exists $L = O(\log n)$, such that for any T_i, T_j, there exists a sequence of subtrees $T_{t(1)}, \ldots, T_{t(L)}$, with*

- *$t(1) = i$, and $t(L) = j$, and*
- *for any $l \in [L-1]$, $\phi_{t(l)}(T_{t(l)})$ is connected to $\phi_{t(l+1)}(T_{t(l+1)})$.*

Proof. By the previous claim, we know that with constant probability there exists a path of length at most 3 between $\phi_i(T_i)$ and $\phi_j(T_j)$ in T'. If this happens, then we have a small path between $\phi_i(T_i)$ and $\phi_j(T_j)$. Otherwise, we look at the trees $\phi_{i'}(T_{i'})$ and $\phi_{j'}(T_{j'})$ which are connected to $\phi_i(T_i)$ and $\phi_j(T_j)$ towards the root, and they visit the vertex $p(p(v))$. Note that with constant probability (by the previous claim again) there exists a path of length at most 4 between $\phi_{i'}(T_{i'})$ and $\phi_{j'}(T_{j'})$. By continuing this argument towards the root $6 \log n$ times, it follows that with probability $1 - n^{-6}$ there exists a path of length at most $20 \log n$. By an union bound argument it follows that with probability $1 - n^{-4}$ every $\phi_i(T_i)$ and $\phi_j(T_j)$ which have a vertex in common are connected by a path of length at most $20 \log n$ in T'. $\square$

Claim 16. *Let T_i, T_j be two subtrees such that they both visit some vertex $v \in V(T)$. Then, with probability at least $1 - n^{-4}$, for any $v_i \in V(T_i)$, and for any $v_j \in V(T_j)$, $D_T(v_i, v_j) \leq D_{T'}(\phi_i(v_i), \phi_j(v_j)) \leq (D_T(v_i, v_j) + 1)O(\log n)$.*

Proof. Observe that since the diameter of the intersection of the two subtrees is at most 2, in order to approximate the distance between $\phi_i(v_i)$ and $\phi_j(v_j)$ for all v_i, v_j, it suffices to approximate the distance between $\phi_i(v)$ and $\phi_j(v)$. By the previous claim, it easily follows that there a path of length $20 \log n$ that connects $\phi_i(v)$ to $\phi_j(v)$. $\square$

In order to finish the proof, it suffices to consider pairs T_i, T_j that do not intersect. Let T_i, T_j be such a pair of subtrees, and let x_i, x_j be the closest pair of vertices between T_i, and T_j. Let p be the path between x_i to x_j in T. Assume that p visits the subtrees $T_i, T_{t(1)}, \ldots, T_{t(l)}, T_j$. We further assume w.l.o.g., that for each $T_{t(s)}$, p visits at least one edge from $T_{t(s)}$, that does not belong to any other $T_{t(s')}$, with $s \neq s'$. Assume that for each $s \in [l]$, p enters $T_{t(s)}$ in a vertex y_s, and leaves $T_{t(s)}$ at a vertex z_s. We have

$$\begin{aligned}
D_{T'}(\phi_i(x_i), \phi_j(x_j)) &= D_{T'}(\phi_i(x_i), \phi_{t(1)}(y_1)) + \sum_{s=1}^{l} D_{T'}(\phi_{t(s)}(y_s), \phi_{t(s)}(z_s)) + \\
&\quad \sum_{s=1}^{l-1} D_{T'}(\phi_{t(s)}(z_s), \phi_{t(s+1)}(y_{s+1})) + D_{T'}(\phi_{t(l)}(z_l), \phi_j(x_j)) \\
&\leq O(l \cdot \log n) + \sum_{s=1}^{l} D_T(y_s, z_s) \\
&= O(D_T(x_i, y_i) \log n)
\end{aligned}$$

Similarly to the proof of the above claim, we observe that since the intersection of any two trees is short, and we approximate the distance between the closest pair of T_i, and T_j, it follows that we also approximate the distance between any pair of vertices of T_i, and T_j.

This concludes the proof of Lemma 11.

3.7 Improved embeddings into $\mathbb{R}^1$ via composing relative embeddings

In this section we obtain a polynomial time algorithm for embedding a metric M into the line. The idea of the algorithm is to embed the metric first into a tree metric using the algorithm from Theorem 3 and then use a result from [18] to embed the tree into the line. The resulting approximation factor is better than the one given by Theorem 1, in a certain range of the parameters.

Let F, F' be families of n-point metric spaces. We say that an algorithm A is an *$\alpha(c)$-distortion algorithm from F to F'*, if on input $X \in F$, it outputs $X' \in F'$, and an embedding $f : X \to X'$, with distortion $\alpha(c)$, where c is the optimal distortion for embedding X into a metric in F'. We also say that *F β-embeds into F'*, if for any $X \in F$, there exists $X' \in F'$, such that X can be embedded into X', with distortion at most β.

Lemma 15. *Let F_1, F_2, F_3 be families of n-point metric spaces, such that F_3 β-embeds into F_2. Let A_1 be an $\alpha_1(c)$-distortion algorithm from F_1 to F_2, and let A_2 be an $\alpha_2(c)$-distortion algorithm from F_2 to F_3. Then, there exists a $\beta \cdot c \cdot \alpha_2(c \cdot \alpha_1(\beta \cdot c))$-algorithm from F_1 to F_3.*

Proof. Assume that we are given X_1 that c-embeds into F_3. It follows that X_1 embeds into F_2 with distortion $\beta \cdot c$. We compute using A_1 an embedding f_1 of X_1 into $X_2 \in F_2$, with distortion $\alpha_1(\beta \cdot c)$. In other words, the distances in X_2 $\alpha_1(\beta c)$-approximate the distances in X_1. Therefore, X_2 embeds into F_3 with distortion at most $d = c \cdot \alpha_1(\beta \cdot c)$. Using A_2, we compute an embedding f_2 of X_2 into $X_3 \in F_3$, with distortion $\alpha_2(d) = \alpha_2(c \cdot \alpha_1(\beta \cdot c))$. Since X_2 $\alpha_1(\beta c)$-approximates X_1, it follows that the composition $f_2 \circ f_1$ is an embedding of X_1 into F_3, with distortion at most $\beta \cdot c \cdot \alpha_2(c \cdot \alpha_1(\beta \cdot c))$. $\square$

Corollary 1. *There exists a polynomial-time algorithm that given a metric M of spread Δ that c-embeds into the line, computes an embedding of M into the line with distortion* $(c \log n)^{O(\sqrt{\log \Delta})}$.

Proof. We apply Lemma 15 with F_1 the family of all n-point metrics of spread at most Δ, F_2 the family of all n-point trees, and F_3 the family of all n-point line metrics. A_1 is the algorithm given in Theorem 3, A_2 is the $c^{O(1)}$-distortion algorithm for embedding trees into the line from [18], and $\beta = 1$, since each line metric is also a tree metric. $\square$

3.8 The relation between embedding into trees and embedding Into subtrees

In this section we study the relation between embedding into trees, and embedding into spanning subtrees. More specifically, let $G = (V, E)$ be an unweighted graph. Assume that G embeds into a tree with distortion c, and also that G embeds into a spanning subtree with distortion c^*.

Clearly, since every spanning subtree is also a tree, we have $c \leq c^*$. We are interested in determining how large the ratio c^*/c can be. We show that for every n_0, there exists $n \geq n_0$, and an n-vertex unweighted subgraph G, for which the ratio is $\Omega(\log n / \log \log n)$. We complement this lower bound by showing that for every unweighted graph G, the ratio is at most $O(\log n)$.

3.8.1 The lower bound

In this section we prove a gap between the distortion of embedding graph metrics into trees, and into spanning subtrees. We do this by giving an explicit infinite family of graphs.

Let $n > 0$ be an integer. We define inductively an unweighted graph $G = (V, E)$ with $\Theta(n)$ vertices, and prove that G $O(\log n)$-embeds into a tree, while any embedding of G into a subtree has distortion $\Omega(\log^2 n / \log \log n)$.

Let G_1 be a cycle on $\log n$ vertices. We say that the cycle of G_1 is *at level* 1. Given G_i, we obtain G_{i+1} as follows. For any edge $\{u, v\}$ that belongs to a cycle at level i, but not to a cycle at level $i-1$, we add a path $p_{u,v}$ of length $\log n - 1$ between u and v. We say that the resulting cycle induced by path $p_{u,v}$ and edge $\{u, v\}$ is at level $i+1$.

Let $G = G_{\log n/\log\log n}$. It is easy to see that $|V(G)| = \Theta(n)$. Moreover, every edge of G belongs to either only one cycle of size $\log n$ at level $\log n/\log\log n$, or exactly two cycles of size $\log n$; one at level i, and one at level $i+1$, for some i, with $1 \leq i < \log n/\log\log n$.

We associate with G a tree $T_C = (V(T_C), E(T_C))$, such that $V(T_C)$ is the set of cycles of length $\log n$ of G, and $\{C, C'\} \in E(T_C)$ iff C and C' share an edge. We consider T_C to be rooted at the unique cycle of G at level 1.

Lemma 16. *Any embedding of G into a subtree has distortion $\Omega(\log^2 n/\log\log n)$.*

Proof. Let T be a spanning subtree of G. Let $k = \log n/\log\log n$. We will compute inductively a set of cycles $\mathcal{C}$, while maintaining a set of edges $E' \subseteq E(G)$. Initially, we set $\mathcal{C} = C_1$, where C_1 is the cycle of G at level 1, and $E' = \emptyset$. At each iteration, we consider the subgraph

$$G' = \left(\bigcup_{C \in \mathcal{C}} C\right) \setminus E'.$$

We pick a cycle $C \notin \mathcal{C}$, such that C shares an edge e with G', and we add C in $\mathcal{C}$, and e in E'. Observe that at every iteration G' is a cycle. Thus, we can pick e and C such that $e \notin T$. The process ends when we cannot pick any more such e and C, with $e \notin T$.

Consider the resulting graph $G' = \left(\bigcup_{C \in \mathcal{C}} C\right) \setminus E'$. Since G' is a cycle, it follows that there exists an edge $e' = \{u, v\} \in G'$, such that $e' \notin T$. Since there is no cycle $C' \notin \mathcal{C}$, with $e' \in C'$, it follows that e' belongs to a cycle at level k. Thus, there exists a sequence of length $\log n$ cycles, $K_1, \ldots, K_k$, with $K_1 = C_1$, and $K_k = C'$, and such that $K_i \in \mathcal{C}$, for each i, with $1 \leq i \leq k$, and the there exists a common edge $e_i \in E'$ in K_i and K_{i+1}, for each i, with $1 \leq i < k$.

Consider the sequence of graphs obtained from G after removing the edges e', e_{k-1}, e_k, …, e_1, in this order. It is easy to see that after removing each edge, the distance between u and v in the resulting graph increases by at least $\Omega(\log n)$. Since none of there edges is in T, it follows that the distance between u and v in T is at least $k \log n = \log^2 n / \log\log n$. $\square$

Lemma 17. *There exists an embedding of G into a tree, with distortion $O(\log n)$.*

Proof. We will construct a tree $T = (V(T), E(T))$ as follows: Initially, we set $V(T) = V(G)$, and $E(T) = \emptyset$. For the cycle C_1 at level 1, we pick an arbitrary vertex $v_{C_1} \in C_1$. Next, for each $u \in C_1$, with $u \neq v_{C_1}$, we add an edge between u and v_{C_1} in T of length $D_G(u, v_{C_1})$.

For every other cycle C' at some level $i > 1$, let $e' = \{u', v'\}$ be the unique edge that C' shares with a cycle C'' at level $i-1$. We pick a vertex $v_{C'}$ arbitrarily between one of the two endpoints of e'. For every vertex $x \in C'$, with $x \neq v_{C'}$, we add an edge between x and $v_{C'}$ in T, of length $D_G(x, v_{C'})$.

Clearly, the resulting graph T is a tree. It is straightforward to verify that for every $\{u, v\} \in E(T)$, $D_T(u, v) = D_G(u, v)$, and thus the resulting embedding is non-contracting. It remains to bound the expansion for any pair of vertices $x, y \in V(G)$. We will consider the following cases.

Case 1. There exists a cycle $C \in V(T_C)$, such that $x, y \in C$: We have

$$\begin{aligned} D_T(x, y) &= D_T(x, v_C) + D_T(v_C, y) \\ &= D_G(x, v_C) + D_G(v_C, y) \\ &< \log n \\ &\leq D_G(x, y) \log n \end{aligned}$$

Case 2. There exist $C_x, C_y \in V(T_C)$, with $x \in C_x$, and $y \in C_y$, such that C_y lies on the path in T_C from C_x to the root of T_C: Consider the path $K_1, \ldots, K_l$ in T_C, with $C_x = K_1$, and $C_y = K_l$. For each i, with $1 \leq i < l$, let $e_i = \{x_i, y_i\} \in E(G)$ be the common edge of K_i and K_{i+1}. Note that the shortest path p from x to y in G visits at least one of the endpoints of each edge e_i. Assume w.l.o.g. that p visits $x_1, x_2, \ldots, x_{l-1}$ (in this order). Observe that each i, with $1 \leq i < l$, for each $v \in K_i$ we have either $D_T(x_i, v) = D_G(x_i, v)$, or $D_T(x_i, v) = D_G(x_i, y_i) + D_G(y_i, v) \leq D_G(x_i, v) + 2$. Thus, we obtain

$$\begin{aligned}
D_T(x,y) &\leq D_T(x,x_1) + D_T(x_1,x_2) + \ldots + D_T(x_{l-2},x_{l-1}) + D_T(x_{l-1},y) \\
&< D_G(x,x_1) + D_G(x_1,x_2) + \ldots + D_G(x_{l-2},x_{l-1}) + 2(l-2) \\
&\quad + D_G(x_{l-1},y) + \log n/2 \\
&< D_G(x,y) + 2\log n/\log\log n + (\log n)/2 \\
&< D_G(x,y) 3\log n
\end{aligned}$$

Case 3. There exist $C_x, C_y, C_z \in V(T_C)$, with $x \in C_x$, and $y \in C_y$, such that C_z is the nca of C_x and C_y in T_C: This Case is similar to Case 2.

□

Theorem 4. *For every $n_0 > 0$, there exists $n \geq n_0$, and an n-vertex unweighted graph G, such that the minimum distortion for embedding G into a tree is $O(\log n)$, while the minimum distortion for embedding G into any of its subtrees is $\Omega(\log^2 n/\log\log n)$.*

Proof. It follows by Lemmata 16 and 17. □

3.8.2 The upper bound

We now complement the lower bound given above with an almost matching upper bound for unweighted graphs. The idea is to first use the $O(1)$-approximation algorithm from Section 3.4 for embedding unweighted graphs into trees to obtain the clustering $\mathcal{K}_G$. Then, by slightly modifying this clustering, we can guarantee that each cluster induces a connected subgraph of the original graph, and thus it can be easily embedded into a spanning subtree. Next, for each cluster we define a new randomly chosen clustering. This new clustering will be used in the final step to merge the computed subtrees of the clusters, into a spanning subtree of the graph, while losing only a $O(\log n)$ factor in the distortion.

Let $G = (V, E)$ be an unweighted graph, that embeds into an unweighted tree with distortion c. For a subset $V' \subseteq V(G)$, and for every $u, v \in V'$, we denote by $D_{V'}(u, v)$ the shortest path distance between u and v in $G[V']$. If $G[V']$ is disconnected, we can assume that $D_{V'}(u, v) = \infty$.

Consider the set tree-like partition $(T_{\mathcal{K}}^G, \mathcal{K}_G)$ constructed by the algorithm of Section 3.4. Let $\mathcal{K}_G = \{K_{r_1}, K_{r_2}, \ldots\}$, and assume that $T_{\mathcal{K}}^G$ is rooted at K_r.

Let $F_{\mathcal{K}}$ be the forest obtained by removing from $T_{\mathcal{K}}^G$ all the edges between vertices at levels $21j$ and $21j + 1$, for all j, with $1 \leq j < \lfloor \text{depth}(T_{\mathcal{K}}^G)/21 \rfloor - 1$. Let $C(F_{\mathcal{K}})$ be the set of connected components of $F_{\mathcal{K}}$. Let

$$\mathcal{J} \;=\; \bigcup_{A \in C(F_{\mathcal{K}})} \left\{ \bigcup_{K_i \in A} K_i \right\}.$$

Clearly, $\mathcal{J}$ is a partition of $V(G)$. Let $T_{\mathcal{J}}$ be the tree on vertex set $\mathcal{J}$, where the edge $\{J_i, J_j\}$ is in $T_{\mathcal{J}}$ if there exist $\{K_{i'}, K_{j'}\} \in E(T_{\mathcal{K}}^G)$, such that $K_{i'} \in J_i$, and $K_{j'} \in J_j$. We consider $T_{\mathcal{J}}$ as being rooted at a vertex $J_r \in \mathcal{J}$, where $K_r \in J_r$.

Lemma 18. *For each $J_i \in \mathcal{J}$, $G[J_i]$ is connected.*

Proof. Assume w.l.o.g., that J_i is the union of sets of vertices K_j, for all $K_j \in A$, where $A \in C(F_{\mathcal{K}})$ is a subtree of $T_{\mathcal{J}}$. Assume that $K_{r'}$ is the vertex of A that is closest to K_r in $T_{\mathcal{K}}^G$. Let p_l be the unique path in A from $K_{r'}$ to a leaf K_l of A. Let also $J_i^l = \bigcup_{K_k \in p_l} K_k$. It suffices to show that for each leaf l, the induced subgraph $G[J_i^l]$ is connected.

Let $p_l = K_1, K_2, \ldots K_t$, where $K_{r'} = K_1$, and $K_l = K_t$. Note that $t \geq 21$. Assume now that $G[J_i^l]$ is disconnected, and let $C(G[J_i^l])$ be the set of connected components of $G[J_i^l]$.

Claim 17. *There exists t', with $1 \leq t' \leq t$, and $C_1 \neq C_2 \in C(G[J_i^l])$, such that $K_{t'} \cap C_1 \neq \emptyset$, and $K_{t'} \cap C_2 \neq \emptyset$.*

Proof. Assume that the assertion in not true. That is, for each t', with $1 \leq t' \leq t$, $K_{t'}$ is contained in a connected component $C'_{t'} \in C(G[J_i^l])$. Observe that for each t'', with $1 \leq t'' < t$, there exists at least one edge between $K_{t''}$ and $K_{t''+1}$. This means that all the $C'_{t'}$s are in fact the same connected component, and thus $C(G[J_i^l])$ contains a single connected component. It follows that J_i^l is connected, a contradiction. $\square$

Claim 18. *There exist $C_1, C_2 \in C(G[J_i^l])$, such that $K_{11} \cap C_1 \neq \emptyset$, and $K_{11} \cap C_2 \neq \emptyset$.*

Proof. Let t', with $1 \leq t' \leq t$, and $C_1, C_2 \in C(G[J_i^l])$ be given by Claim 17. If $t' = 11$, then there is nothing to prove.

Otherwise, pick $v_1 \in K_{t'} \cap C_1$, and $v_2 \in K_{t'} \cap C_2$. By the construction of $\mathcal{K}$, we have that there exists a path p from v_1 to v_2, such that p is the concatenation of the paths $q_{t'}, \ldots, q_1, q, q'_1, \ldots, q'_{t'}$, where for each $i \in [1, t']$, q_i and q'_i are paths of length at most c in K_i. Moreover, there exists a path $\bar{p}$ from v_1 to v_2, such that $\bar{p}$ is the concatenation of the paths $w_{t'}, \ldots, w_t, w, w'_t, \ldots, w'_{t'}$, where for each $i \in [t', t]$, w_i and w'_i are paths of length at most c in K_i.

If $t' > 11$, then pick $v'_1 \in q_{11}$, and $v'_2 \in q'_{11}$. Otherwise, if $t' < 11$, pick $w'_1 \in q_{11}$, and $v'_2 \in w'_{11}$. Clearly, in both cases we have $v'_1 \in C_1$, and $v'_2 \in C_2$. $\square$

Let now $C_1, C_2 \in C(G[J_i^l])$ be the connected components given by Claim 18. Pick $v_1 \in K_{t'} \cap C_1$, and $v_2 \in K_{t'} \cap C_2$. Let p be the shortest path between v_1 and v_2 in G. We observe that there are two possible cases for p:

Case 1: p is the concatenation of the paths $q_{11}, \ldots, q_1, q, q'_1, \ldots, q'_{11}$, where for each $i \in [1, 11]$, q_i and q'_i are contained in K_i.

Case 2: p is the concatenation of the paths $q_{11}, \ldots, q_t, q, q'_t, \ldots, q'_{11}$, where for each $i \in [11, t]$, q_i and q'_i are contained in K_i.

Since the above two Cases can be analyzed identically, we assume w.l.o.g. that p satisfies Case 1. Observe that for each $i \in [1, 11)$, each q_i and each q'_i visits c vertices of K_i. It follows that the length of p is greater than $20c$, contradicting Lemma 7. $\square$

For each $J_i \in \mathcal{J}$, we define a set $\mathcal{J}_i$ of subsets of J_i as follows. First, we pick a vertex $r_i \in J_i$, and we construct a BFS tree T_{J_i} of $G[J_i]$, rooted at r_i. Note that by Lemma 18, $G[J_i]$ is connected, and thus there exists such a BFS tree. We also pick an integer $\alpha_{J_i} \in [0, 100c)$, uniformly at random. Let F_{J_i} be the forest obtained from

T_{J_i} by removing the edges between vertices at levels $100cj + \alpha_{J_i}$ and $100cj + \alpha_{J_i} + 1$, for all j, with $1 \leq j < \left\lfloor \frac{\text{depth}(T_{J_i})}{100c} \right\rfloor - 2$. The set $\mathcal{J}_i$ can now be defined as the set of sets of vertices of the connected components of F_{J_i}. Clearly, $\mathcal{J}_i$ is a partition of J_i.

Lemma 19. *For each $J_i, J_j \in \mathcal{J}$, such that J_i is the parent of J_j in $T_{\mathcal{J}}$, and for each $J_{j,k} \in \mathcal{J}_j$, there exist $u \in J_i$, and $v \in J_{j,k}$, such that $\{u, v\} \in E(G)$.*

Proof. It is easy to verify by the construction of $\mathcal{K}_G$ that J_j is a subset of the vertices of at least $21c$, and at most $42c$ consecutive levels of a BFS tree of G. Let $l_1, \ldots, l_t$ be these levels, where l_1 is the level closest to the root of the BFS tree of G. For every vertex $x \in J_j$, there exists a vertex $y \in J_i$, such that $\{x, y\} \in E(G)$, iff $x \in l_1$. Thus, it suffices to show that for every $J_{j,k} \in \mathcal{J}_j$, $J_{j,k} \cap l_1 = \emptyset$.

It is easy to verify that for every $v \in J_j$, there exists $u \in l_1$, such that $D_{J_j}(v, u) < 42c$. In the construction of $\mathcal{J}_j$, we pick a vertex $r_j \in J_j$, and we compute a BFS tree T' of G_{J_j}. Every $J_{j,k} \in \mathcal{J}_j$ is a subtree $T_{j,k}$ of T' rooted at a vertex $r_{j,k}$. $T_{j,k}$ contains all the predecessors of $r_{j,k}$ that are at distance at most $\delta_{j,k}$, for some $100c \leq \delta_{j,k} \leq 200c$. Assume now that there is no vertex of l_1 in the $42c$ first levels of $T_{j,k}$. Pick a vertex of $T_{j,k}$ at level $42c + 1$. By the above argument, there exists a vertex $u \in l_1$ that is at distance at most $42c$ from v. This implies that u is contained within the $84c + 1$ first levels of $T_{j,k}$. Thus, $T_{j,k} \cap l_1 \neq \emptyset$, and $J_{j,k} \cap l_1 \neq \emptyset$. $\square$

Lemma 20. *For each $J_i, J_j \in \mathcal{J}$, such that J_i is the parent of J_j in $T_{\mathcal{J}}$, and for each $u, v \in J_i$, and $u', v' \in J_j$, such that $\{u, u'\} \in E(G)$, and $\{v, v'\} \in E(G)$, $D_{J_i}(u, v) \leq 90c$.*

Proof. Note that the partition $\mathcal{K}_G$ is obtained on a BFS tree of G with root some $r \in V(G)$. If $r \in J_i$, then $D_{J_i}(u, v) \leq D_{J_i}(u, r) + D_{J_i}(r, v) \leq 84c$.

It remains to consider the case $r \notin V(G)$. This implies that there exists $J_k \in \mathcal{J}$, such that J_k is the parent of J_i in $T_{\mathcal{J}}$. Assume that the assertion is not true. That is, there exist $u, v \in J_i$, and $u', v' \in J_j$, with $\{u, u'\} \in E(G)$, $\{v, v'\} \in E(G)$, and $D_{J_i}(u, v) > 90c$. By the construction of $\mathcal{K}_G$, and since $r \notin J_i$ it follows that there exist $w, z \in J_i$, and $w', z' \in J_k$, with $\{w, w'\} \in E(G)$, and $\{z, z'\} \in E(G)$, and moreover there exists a shortest path p_1 in G from w to u, and a shortest path p_2 from v to z

in G, such that p_1 and p_2 are contained in J_i. It is easy to verify that the length of each of the paths p_1 and p_2 is at least $22c$.

Furthermore, there exists a path p_3 from w' to z', and a path p_4 from u' to v', such that both p_3 and p_4 do not visit J_i. Let p_3' be the path obtained from p_3 by adding the edges $\{w, w'\}$, and $\{z', z\}$. Similarly, let p_4' be the path obtained from p_4 by adding the edges $\{u, u'\}$, and $\{v', v\}$.

Let x_1 be a vertex of p_1 such that $D_G(x_1, u) > 5c$, and $D_G(x_1, w) > 5c$. Similarly, let x_2 be a vertex of p_2 such that $D_G(x_2, v) > 5c$, and $D_G(x_2, z) > 5c$. We need to define the following set of paths:

- Let q_1 be the subpath of p_1 from u to x_1.
- Let q_2 be the path obtained by concatenating the subpath of p_1 from x_1 to w, with p_3.
- Let q_3 be the subpath of p_2 from z to x_2.
- Let q_4 be the path obtained by concatenating the subpath of p_2 from x_2 to v, with p_4.

It is straight-forward to verify that $D_G(q_1, q_3) > 5c$, and $D(q_2, q_4) > 5c$. By applying Lemma 6, we obtain that the optimal distortion for embedding G into an unweighted tree is more than $5c$, a contradiction. $\square$

Theorem 5. *If an unweighted graph G can be embedded into a tree with distortion c, then G can be embedded into a subtree with distortion $O(c \log n)$.*

Proof. We can compute an embedding of G into a subtree T as follows. Initially, we set T equal to the empty subgraph. We pick a vertex $r \in V(G)$, and we compute a (r, c)-partition of G. We compute the partition $\mathcal{J}$, and for each $J_i \in \mathcal{J}$, we compute the partition $\mathcal{J}_i$, as described above. For each $J_i \in \mathcal{J}$, and for each $J_{i,j} \in \mathcal{J}_i$, we add to T a spanning tree of $J_{i,j}$ of radius $O(c)$.

It remains to connect the subtrees by adding edges between the sets $J_{i,j}$. Observe that if $r \in J_i$, then $\mathcal{J}_i$ contains a single set $J_{i,j}$.

Assume now that $r \notin J_j$, and let J_i be the parent of J_j in $T_{\mathcal{J}}$. By Lemma 19, for each $J_{j,k} \in \mathcal{J}_j$, there an edge between $J_{j,k}$ and J_i in G. For each such $J_{j,k}$, we pick one such edge, uniformly at random, and we add it to T.

Consider now two subsets $J_{j,k}, J_{j,l} \in \mathcal{J}_j$. It is easy to see that $J_{j,k}$, and $J_{j,l}$ get connected to the same subset $J_{i,t} \in \mathcal{J}_i$, with probability at least $1 - \frac{90c}{100c} = \Omega(1)$. Thus, the probability that two such subsets have not converged to the same subset in an ancestor after $O(\log n)$ levels is at most 1/poly(n). Since there are at most n^2 pairs of such subsets $J_{i,j}$, it follows that the above procedure results in a tree with distortion $O(c \log n)$ with high probability. $\square$

Chapter 4

Embedding ultrametrics into $\mathbb{R}^d$

In this chapter we present an algorithm for embedding ultrametrics into $\mathbb{R}^d$. More specifically, if the input ultrametric embeds into the Euclidean plane with distortion c, then the embedding produced by the algorithm has distortion $O(c^3)$. In particular, for the case where the input ultrametric is embeddable into the plane with constant distortion, the distortion of the embedding produced by the algorithm is also constant. The running time of our algorithm is linear in the input size, assuming it is given the value of the optimum distortion c (or its approximation). The algorithm generalizes to embeddings into $\mathbb{R}^d$, and the distortion becomes $c^{O(d)}$, where c is the distortion of the optimal embedding of the ultrametric into $\mathbb{R}^d$. We remark that for any fixed $d > 0$, all norms of $\mathbb{R}^d$ are equivalent up to a constant factor in the distortion. Therefore our bounds hold asymptotically for any norm.

We also prove that any ultrametric can be embedded into the plane with distortion $O(\sqrt{n})$. More generally, for any $d \geq 2$, we show how to embed any ultrametric into $\mathbb{R}^d$ with distortion $d^{O(1)}n^{1/d}$. Notice that unlike the first result, this result relates to the absolute version of the distortion minimization problem. The proof is algorithmic, and the embedding can be found in polynomial time. Combining the two results together, we obtain an $O(n^{1/3})$-approximation algorithm for embedding ultrametrics into the plane.

The results presented in this chapter are from [19].

4.1 Overview of techniques

We use the well-known fact that any ultrametric $M = (X, D)$ can be well approximated by hierarchically well-separated trees (HSTs) (see the next Section for definitions). The corresponding HST T has the points of X as its leaves, and each vertex v of T has a label $l(v) \in \mathbb{R}^+$. The distance of any pair of points $p, q \in X$ is exactly the label of their nearest common ancestor.

The hierarchical structure of an HST naturally enables constructing the embedding in a recursive manner. That is, the mapping is constructed by embedding (recursively and independently) the children of the root node, and then combining the embeddings. Implementing this idea, however, requires overcoming a few obstacles, which we discuss now. For simplicity, we focus on embeddings into the plane.

Distortion lower bound. The first issue is how to obtain a good lower bound for the distortion. It is not difficult to see that the distortion depends on both the number of nodes, and the structure of the ultrametric. For example, the full 2-HST of depth t, where all internal nodes have degree 4, requires $\Omega(t)$ distortion; at the same time, the full 4-HST of depth t, where all internal nodes have degree 4, can be embedded with constant distortion.

Our lower bound is obtained as follows. Consider any node v and its children $u_1 \ldots u_k$. Let P_i be the set of leaves in the subtree of the node u_i, $P = P_1 \cup \ldots \cup P_k$. By the definition of ultrametrics, the distances between any pair of points $p \in P_i$ and $q \in P_j$ for $i \neq j$, are equal to the same value, namely $l(v)$. Consider any non-contracting embedding $f : P \to \mathbb{R}^2$. Construct a ball of radius $l(v)/2$ around each point $f(p)$, $p \in P$, and denote this ball by $B(p, l(v)/2)$. It is easy to see that the union of the interiors of the balls around points in P_i and the union of the interiors of the balls around points in P_j must be disjoint if $i \neq j$.

Our lower bound on distortion proceeds by estimating the total volume $C(v)$ of $\cup_{p \in P} B(p, l(v)/2)$. Specifically, by packing argument, one can observe that the distortion of the optimal embedding must be at least $\Omega(\sqrt{C(v)} - O(1))$. Thus, it suffices to have a good lower bound for the volume $C(v)$. It would appear that such lower bounds could be obtained by summing $C(u_i)$'s, since the balls around different sets P_i are disjoint. Unfortunately, $C(u_i)$ is the volume of the union of the balls of radius $l(u_i)/2$, not $l(v)/2$, so the above is not strictly true. However, $\cup_{p \in P_i} B(p, l(v)/2)$ can be expressed as a Minkowski sum of $\cup_{p \in P_i} B(p, l(u_i)/2)$ and a ball of radius $[l(v) - l(u_i)]/2$. Then the volume of that set can be bounded from below by using Brunn-Minkowski inequality, by a function of $C(u_i)$ and $l(v) - l(u_i)$. This enables us to obtain a recursive formula for $C(v)$ as a function of $C(u_i)$'s.

Distortion accumulation. The recursive formula for the lower bound suggests a recursive algorithm. Consider some vertex v of the HST, and let $u_1, \ldots, u_k$ be its children. For each u_i, $1 \le i \le k$, the leaves in the subtree of u_i are mapped into a square $R(u_i)$ whose volume is at most $C(u_i)$. Then the squares are re-arranged to form a square $R(v)$. The main difficulty with this approach is that the optimal way to pack the squares is difficult to find. In fact, the optimal embedding could, in principle, not pack the points into squares. To overcome this problem, we allow some limited stretching of the squares, to fit them into $R(v)$. However, stretching causes distortion, and thus we need to make sure that stretching done over different levels does not accumulate. In order to avoid such accumulation of distortion, we alternate between the horizontal and vertical stretchings of the squares. Specifically, we assign, for each vertex v of the HST, a bit $g(v)$ that determines whether the squares into which the sub-trees of the children of v are embedded will be stretched in the horizontal or the vertical direction before they are packed into the square $R(v)$. We calculate the values of the bits $g(v)$ in a top-down manner, starting with the leaves of the HST, to ensure that the final stretchings are balanced.

It appears that the need to compute a proper choice of stretching directions (which can also be viewed as rotations) at each level is not just an artifact of our algorithm, but it might be necessary to achieve low distortion. In particular, the only constant distortion embedding of a full 2-HST into the plane that we are aware of uses alternating rotations.

Higher dimensions. We show how to generalize the algorithm for embedding ultrametrics into the plane to higher dimensions. We show an algorithm that produces a $c^{O(d)}$-distortion embedding of the input ultrametric into $\mathbb{R}^d$ under the l_2 norm, where c denotes the optimal distortion achievable when embedding the input ultrametric into $\mathbb{R}^d$.

4.2 Preliminaries

A metric $M = (X, D)$ is an *ultrametric*, if it can be represented by a *labeled tree* T whose set of leaves is X, in the following manner. Each non-leaf vertex v of T has a label $l(v) > 0$. If u is a child of v in tree T, then $l(u) \leq l(v)$. For any $x, y \in X$, the distance between x and y is defined to be the label of the nearest common ancestor of x and y, and this distance should be equal to $D(x, y)$.

We now proceed to define hierarchically well-separated trees (HSTs). For any $\alpha \geq 1$, an α-HST is an ultrametric where for each parent-child pair of vertices (u, v), $l(u) = \alpha l(v)$. It is easy to see that for any $\alpha \geq 1$, any ultrametric can be α-approximated by an α-HST (cf. [9]). Moreover, such an HST can be found in time linear in the input size. Therefore, if the input ultrametric M embeds into $\mathbb{R}^d$ with distortion c, then the metric M' defined by the corresponding 2-HST embeds into $\mathbb{R}^d$ with distortion $c' = 2c$. Any non-contracting embedding of M' into $\mathbb{R}^d$ with distortion c'' represents a non-contracting embedding of M with distortion $O(c'')$. Therefore, from now on we will concentrate on embeddings of HSTs into $\mathbb{R}^d$.

Given a 2-HST T, we will use the following additional notation. Let r denote the root of the tree, and let h denote the tree height. We assume that r belongs to the first level of T, and all the leaves belong to level h. By scaling the underlying metric M, we can assume w.l.o.g., that for each vertex v at level $h-1$, $l(v) = 2$. For any non-leaf vertex v, we denote by X_v the set of leaves of the subtree of T rooted at v, and we denote the number of leaves in the subtree n_v.

We will use the Brunn-Minkowski inequality, defined as follows. Given any two sets $A, B \subseteq \mathbb{R}^d$, let $A \oplus B$ denote the Minkowski sum of A and B, i.e., $A \oplus B = \{a + b \mid a \in A, b \in B\}$.

Theorem 6 (Brunn-Minkowski inequality)**.** *For any pair of sets $A, B \subseteq \mathbb{R}^d$,*

$$Vol(A \oplus B)^{1/d} \geq Vol(A)^{1/d} + Vol(B)^{1/d}.$$

4.3 A lower bound on the optimal distortion

In this section we show a lower bound on the distortion of optimal embedding of a metric M' which is defined by a 2-HST denoted by T.

For any $r > 0$, let $B(r)$ denote the ball of radius r in ℓ_2^d centered at the origin. Let $V_d(r)$ denote the volume of a d-dimensional ball of radius r, $V_d(r) = \frac{\pi^{d/2} r^d}{\Gamma(1+d/2)}$. For each vertex v of T, we define a value $C(v)$, which intuitively is a lower-bound on the minimum volume embedding of X_v (the precise statement appears below). The values $C(v)$ are defined recursively, starting from the leaves. For each leaf v, we set $C(v) = V_d(1/2)$.

Consider now vertex v at level $j \in [h-1]$, and let $u_1, \ldots, u_k$ be the children of v in T. We define:

$$C(v) = \sum_{i=1}^{k} \left((C(u_i))^{1/d} + (V_d(l(v)/4))^{1/d} \right)^d$$

Given any embedding $\phi : X \to \ell_2^d$, for any subset $X' \subseteq X$, let $\phi(X')$ denote the image of points in X' under ϕ.

Lemma 21. *Let v be a non-leaf vertex of T, and let ϕ be any non-contracting embedding of X_v into ℓ_2^d. Then the volume of $\phi(X_v) \oplus B\left(\frac{l(v)}{2}\right)$ is at least $C(v)$.*

Proof. Let $u_1, \ldots, u_k$ be the children of v. The proof is by induction. Assume first that v belongs to level $h-1$ of T, and consider $S = \phi(X_v) \oplus B(l(v)/2)$. Recall that $l(v) = 2$. Since the embedding is non-contracting, for any $1 \leq i < j \leq k$, vertices u_i, u_j are embedded at a distance at least 2 from each other. Therefore, set S consists of k balls of disjoint interiors, of radius 1 each, and thus the volume of S is exactly $kV_d(1) = C(v)$.

Assume now that v belongs to some level $j \in [h-2]$. Let $S = \phi(X_v) \oplus B(l(v)/2)$. Equivalently, S is the union of $S_i = \phi(X_{u_i}) \oplus B(l(v)/2)$ for $i \in [k]$. Since the embedding is non-contracting, all the sets S_i have disjoint interiors. For each $i \in [k]$, let us denote $S_i' = \phi(X_{u_i}) \oplus B(l(u_i)/2)$. Recall that $l(v) = 2l(u_i)$. Therefore, for each $i \in [k]$, $S_i = S_i' \oplus B(l(v)/4)$. Using the induction hypothesis, the volume of S_i' is at least $C(u_i)$. From the Brunn-Minkowski inequality, it follows that:

$$\begin{aligned}(\mathrm{Vol}(S_i))^{1/d} &\geq (\mathrm{Vol}(S_i'))^{1/d} + (V_d(l(v)/4))^{1/d} \\ &\geq (C(u_i))^{1/d} + (V_d(l(v)/4))^{1/d}\end{aligned}$$

Therefore, in total,

$$\begin{aligned}\mathrm{Vol}(S) &= \sum_{i=1}^{k} \mathrm{Vol}(S_i) \geq \sum_{i=1}^{k} \left((C(u_i))^{1/d} + (V_d(l(v)/4))^{1/d}\right)^d \\ &= C(v).\end{aligned}$$

□

Suppose we are given some set of points $S \subseteq \mathbb{R}^d$, that has volume V. We define $\rho_d(V) = \left(\frac{V \cdot \Gamma(1+d/2)}{\pi^{d/2}}\right)^{1/d}$, i.e., $\rho_d(V)$ is the radius of the d-dimensional ball in $\mathbb{R}^d$ that has volume V. Observe that S has two points at a distance at least $\rho_d(V)$ from each other (otherwise, S is contained in a ball of radius smaller than $\rho_d(V)$, which is impossible).

Corollary 2. *Let v be some non-leaf vertex of T, and let ϕ be any non-contracting embedding of M' into ℓ_2^d, with distortion at most c'. Then $c' \geq \rho_d(C(v))/l(v) - 1$.*

Proof. Consider $S = \phi(X_v) \oplus B\left(l(v)/2\right)$. By Lemma 21, the volume of S is at least $C(v)$, and thus there are two points $x, y \in S$ within a distance at least $\rho = \rho_d(C(v))$ from each other. By the definition of S, it follows that there are two points $a, b \in X_v$, which are embedded at a distance of at least $\rho - l(v)$ from each other. As the distance between a, b in T is at most $l(v)$, the bound on the distortion follows. □

4.4 Upper bound on the absolute distortion

In this section we show that for any $d \geq 2$, any n-point ultrametric can be embedded into ℓ_2^d with distortion $O(d^{1/2}n^{1/d})$.

Given an ultrametric M, we first compute an α-HST T that α-approximates M, for some constant $\alpha > 16$. Let M' be the metric associated with T. Observe that any embedding of M' into ℓ_2^d with distortion c, is also an embedding of M into ℓ_2^d, with distortion $O(c)$. Thus, it suffices to show that M' can be embedded into ℓ_2^d with distortion $O(d^{1/2}n^{1/d})$.

We will compute an embedding of M' into ℓ_2^d inductively, starting from the leaves of T. For every subtree of T rooted at a vertex u, we compute an embedding f_u of the submetric of M' induced by X_u, into ℓ_2^d. We maintain the following inductive properties of f_u:

- The contraction of f_u is at most 16.
- $f(X_u)$ is contained inside a hypercube of side length $l(u)n_u^{1/d}$.

We assume w.l.o.g. that for each leave v of T, $l(v) = 1$. Thus, we can embed each leave in a center of a hypercube of side 1. The following lemma shows how to compute the recursive embedding of inner vertices of T.

Lemma 22. *Let v be an internal vertex of T, whose children are $u_1, \ldots, u_k$. Assume that for each $i \in [k]$, we are given an embedding $f_{u_i} : X_{u_i} \to \mathbb{R}^d$, with contraction at most 16, such that $f_{u_i}(X_{u_i})$ is contained inside a d-dimensional hypercube S_{u_i}, with side length $l(u_i)n_{u_i}^{1/d}$. Then we can compute in polynomial time an embedding $f_v : X_v \to \mathbb{R}^d$, with contraction at most 16, such that $f_v(X_v)$ is contained inside a d-dimensional hypercube S_v, with side length $l(v)n_v^{1/d}$.*

Proof. For each $i \in [k]$, let $r_i = l(u_i)n_{u_i}^{1/d}$ be the length of the side of the hypercube S_{v_i}. Let also S'_{u_i} be a hypercube of side length $r'_i = r_i + l(v)/16$, having the same center as S_{u_i}. We assume w.l.o.g. that $n_1 \geq n_2 \geq \cdots \geq n_k$ and thus $r'_1 \geq \cdots \geq r'_k$. We note that for each $i : 1 \leq i \leq k$, $r'_i \leq l(v)n_v^{1/d}/4$, since $r'_i = r_i + l(v)/16 = l(u_i)n_{u_i}^{1/d} + l(v)/16 \leq l(v)n_v^{1/d}/4$.

We first define a partition $\mathcal{R} = \{R_j\}_{j=1}^{\lambda}$, of the set $[k]$, which we will use to partition the set of hypercubes $\{S_{u_i}\}_{i=1}^{k}$, as follows. We will define $\lambda + 1$ integers $t_0, t_1, \ldots, t_\lambda$, where $t_0 = 0$, $t_\lambda = k$, and $t_0 < t_1 < \cdots < t_\lambda$, and then set R_j to contain all the indices $i : t_{j-1} + 1 \le i \le t_j$. This defines a partition of the hypercubes into λ sets $\mathcal{S}_1, \ldots, \mathcal{S}_\lambda$, where $\mathcal{S}_j$ contains the hypercubes S_{u_i} with $i \in R_j$. For each $j : 1 \le j \le \lambda$, let $\rho_j = r'_{t_{j-1}+1}$ denote the side of the largest hypercube in $\mathcal{S}_j$, and let $\rho'_j = r_{t_j}$ denote the side of the smallest hypercube in $\mathcal{S}_j$.

We now proceed to define the numbers t_j, for $j : 0 \le j \le \lambda$. Set $t_0 = 0$, and for each $j \ge 1$, if $t_{j-1} < k$, we inductively define t_j as

$$t_j \quad = \quad \min\{k, t_{j-1} + \lfloor l(v) n_v^{1/d} / r'_{t_{j-1}+1} \rfloor^{d-1}\}.$$

If $t_j = k$ then we set $\lambda = j$.

Note that for any $j \in [\lambda - 1]$,

$$|R_j| = \left\lfloor \frac{l(v) n_v^{1/d}}{\rho_j} \right\rfloor^{d-1}$$

We now define the embedding f_v by placing the hypercubes S'_{u_i} inside a hypercube of side length $l(v) n_v^{1/d}$, such that their interiors do not overlap, using the partition $\mathcal{R}$. For each $j \in [\lambda]$, we place the hypercubes in $\mathcal{S}_j$ inside a parallelepiped W_j having $d - 1$ sides of length $l(v) n_v^{1/d}$, and one side of length ρ_j, as follows. It is easy to see that we can pack $|R_j|$ d-dimensional hypercubes of side ρ_j inside W_j. Since each hypercube in $\mathcal{S}_j$ has side at most ρ_j, we can replace each hypercube embedded into W_j by a hypercube from $\mathcal{S}_j$, such that the centers of both hypercubes coincide.

Finally, we place the parallelepipeds W_j inside a parallelepiped W having $d - 1$ sides of length $l(v) n_v^{1/d}$, and one side of length $\sum_{j=1}^{\lambda} \rho_j$. Figure 4-1 depicts such a placement for the case $d = 2$. Observe first that the contraction of this embedding is at most 16: for any pair of vertices $x, y \in X(v)$, if x, y both belong to a subtree of the same child u_i of v, then by induction hypothesis the distance between them is contracted by at most 16. If $x \in X(u_i), y \in X(u_{i'})$ and $i \ne i'$, then the original distance is $D(x, y) = l(v)$. Since we add empty space of width $l(v)/32$ around the hypercubes $S(u_q)$ when they are transformed into hypercubes $S'(u_q)$, it is clear that the distance between the embeddings of x and y is at least $l(v)/16$.

It now only remains to show that $\sum_{j=1}^{\lambda} \rho_j \leq l(v)n_v^{1/d}$. We partition the parallelepipeds W_j into two types. The first type contains all the parallelepipeds W_j, where $\rho_j/\rho'_j \geq 2$. Additionally, the last parallelepiped W_k is also of the first type, regardless of the ratio ρ_k/ρ'_k. Let $\mathcal{T}_1 \subseteq [k]$ contain all the indices j where W_j is of the first type. All the other parallelepipeds belong to the second type, and let $\mathcal{T}_2 = [k] \setminus \mathcal{T}_1$ contain the indices of the parallelepipeds of the second type. Notice that for $j \in \mathcal{T}_1$, the values ρ_j form a geometric series with ratio $1/2$. Since the sides r'_i of the hypercubes S_{u_i} are bounded by $l(v)n_v^{1/d}/4$, it is easy to see that:

$$\sum_{j \in \mathcal{T}_1} \rho_j \leq \frac{l(v)n_v^{1/d}}{4} \left(1 + \frac{1}{2} + \frac{1}{4} + \cdots\right) \leq \frac{l(v)n_v^{1/d}}{2}$$

It now remains to bound $\sum_{j \in \mathcal{T}_2} \rho_j$. Fix some $j \in \mathcal{T}_2$, and consider some hypercube S'_{u_i} where $i \in R_j$. As W_j is of the second type, we know that $r'_i \geq \rho_j/2$. On the other hand,

$$r'_i = r_i + \frac{l(v)}{16} = l(u_i)n_{u_i}^{1/d} + \frac{l(v)}{16} \leq \frac{l(v)}{16}\left(1 + n_{u_i}^{1/d}\right) \leq \frac{l(v)}{4} n_{u_i}^{1/d}$$

Therefore, $n_{u_i} \geq \left(\frac{2\rho_j}{l(v)}\right)^d$. Recall that for $j : 1 \leq j < \lambda$, $|R_j| = \left\lfloor \frac{l(v)n_v^{1/d}}{\rho_j} \right\rfloor^{d-1} \geq \left(\frac{l(v)n_v^{1/d}}{2\rho_j}\right)^{d-1}$. Therefore, we have that

$$\sum_{i \in R_j} n_{u_i} \geq \left(\frac{l(v)n_v^{1/d}}{2\rho_j}\right)^{d-1} \cdot \left(\frac{2\rho_j}{l(v)}\right)^d \geq \frac{2\rho_j}{l(v)} n_v^{1-1/d}$$

Thus, $\rho_j \leq \frac{l(v)\sum_{i \in R_j} n_{u_i}}{2n_v^{1-1/d}}$, and

$$\sum_{j \in \mathcal{T}_2} \rho_j \leq \frac{l(v)n_v}{2n_v^{1-1/d}} \leq \frac{l(v)n_v^{1/d}}{2}$$

We have that in total, $\sum_j \rho_j = \sum_{j \in \mathcal{T}_1} \rho_j + \sum_{j \in \mathcal{T}_2} \rho_j \leq l(v)n_v^{1/d}$. □

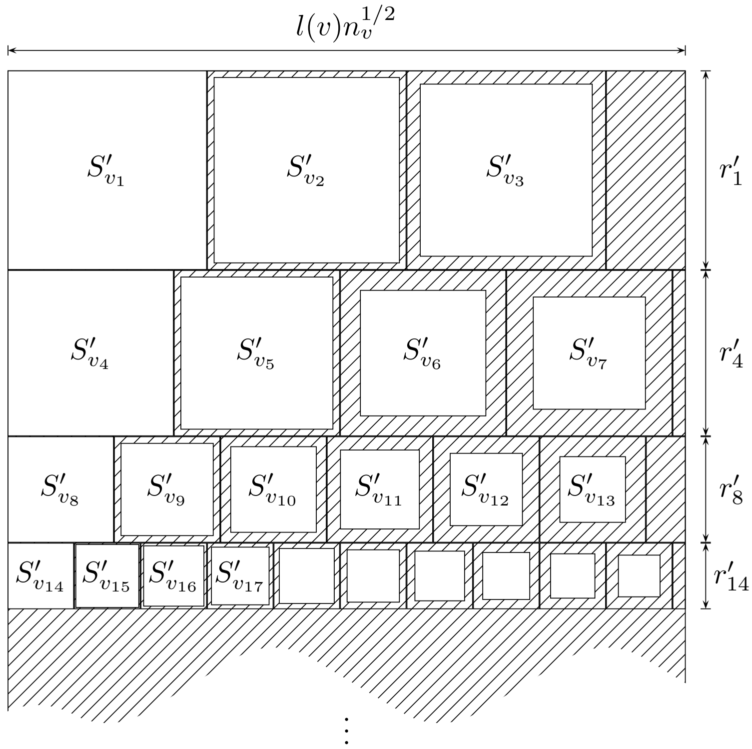


Figure 4-1: The packing computed in the proof of Lemma 22.

We are now ready to prove the main theorem of this section.

Theorem 7. *For any $d \geq 2$, any n-point ultrametric can be embedded into ℓ_2^d with distortion $O(d^{1/2}n^{1/d})$. Moreover, the embedding can be computed in polynomial time.*

Proof. Starting from the leaves of T, we inductively compute for each $v \in V(T)$ the embedding f_v as described above. By recursively applying Lemma 22 we can compute in polynomial time the embedding f_v, that also satisfies the inductive properties. Let f be the resulting embedding f_r.

Consider now two points $x, y \in X$, and let v be the nearest common ancestor of x and y. Since $f_v(X_v)$ is contained inside a hypercube of side length $l(v)n_v^{1/d}$, it follows that $\|f(x) - f(y)\|_2 \leq \left(dn_v^{2/d}l^2(v)\right)^{1/2} = d^{1/2}n^{1/d}D(x, y)$. Since the contraction of f_v is at most 16, it follows that the distortion of f is $O(d^{1/2}n^{1/d})$. $\square$

We remark that Theorem 7 generalizes a result of Gupta [29], who shows that every n-point weighted star metric can be embedded into $\mathbb{R}^d$, with distortion $O(n^{1/d})$. This is a corollary of the following simple observation.

Claim 19. *Every n-point weighted star can be embedded into an ultrametric of size $O(n)$ with distortion at most 2.*

Proof. Consider a star S with root r, and leaves $x_1, \ldots, x_n$, where for each $i \in [n]$, $D_S(r, x_i) = w_i$. Assume w.l.o.g. that $w_1 \leq w_2 \leq \ldots \leq w_n$. We construct a tree T with root r' as follows. T contains a path $z_n, z_{n-1}, \ldots, z_1$, where $z_n = r'$, and for each $i \in [n-1]$, $D_T(r', z_i) = w_n - w_i$. We now embed S into T as follows. For each $i \in [n]$, we add x_i to T, and we connect x_i to z_i with an edge of length w_i. Observe that the shortest-path metric on the leaves of T is an ultrametric, since all the leaves are on the same level. Moreover, for any $i < j \in [n]$, $D_T(x_i, x_j) = 2w_j$, while $D_S(x_i, x_j) = w_i + w_j$, and so the resulting embedding is non-contracting, and has expansion at most 2. □

4.5 Approximation algorithm for embedding ultrametrics into $\mathbb{R}^2$

Let $M = (X, D)$ be the input ultrametric that embeds into the plane with distortion c. Let $M' = (X, D')$ be the metric defined by the 2-HST T which 2-approximates M. Then M' embeds into the plane with distortion $c' \leq 2c$, and any non-contracting embedding of M' into the plane with distortion $O(c'^3)$ is also a non-contracting embedding of M with distortion at most $O(c^3)$. Therefore, from now on we concentrate on embedding M' into the plane.

Consider some non-leaf vertex u. We define $a_u = \sqrt{C(u)}$. If $u \neq r$, let v be its father. We define $b_u = a_u + \frac{\sqrt{\pi} l(v)}{4}$.

Our algorithm works in bottom-up fashion. Let v be some vertex. The goal of the algorithm is to embed all the vertices of X_v into a square Q of side a_v, incurring only small distortion. Let $u_1, \ldots, u_k$ be the children of v, and assume that for all $j : 1 \leq j \leq k$, we have already embedded $X(u_j)$ inside a square Q_j of side a_{u_j}. Recall that for any pair of vertices $x \in X_{u_j}$, $y \in X_{u_{j'}}$, where $1 \leq j \neq j' \leq k$, the distance between x and y in T is $l(v)$. Our first step is to ensure non-contraction (or more precisely small contraction), by adding empty strips of width $\frac{b_{u_j}-a_{u_j}}{2} = \frac{\sqrt{\pi}l(v)}{8}$ around the squares. Thus, we obtain a collection $Q'_1, \ldots, Q'_k$ of squares, of sides $b_{u_1}, \ldots, b_{u_k}$, respectively. Our goal now is to pack these squares into one large square Q of side a_v. Observe that from volume view point, $\mathrm{Vol}(Q) = \mathrm{Vol}(Q'_1) + \ldots + \mathrm{Vol}(Q'_k)$, since $a_v^2 = \sum_{j=1}^{k} b_{u_j}^2$, by the definition of C_v. However, it is not always possible to obtain such tight packing of squares. Instead, we convert each square Q'_j to rectangle R_j whose sides are $b_{u_j}s_{u_j}$, b_{u_j}/s_{u_j} for some $s_{u_j} = O(c')$. Observe that the volume of R_j is the same as that of Q'_j. This will enable us to pack all the rectangles $R_1, \ldots, R_k$ into Q. Recall that inside each square Q'_j, vertices of X_{u_j} are embedded. In order to convert square Q'_j into rectangle R_j, we contract all the distances along one axis, and expand all the distances along the other axis, by the same factor s_{u_j}.

Consider now two vertices u, v, and let z be their least common ancestor. The distance between u and v might thus be contracted or expanded when we calculate the embedding of X_z. However, for each vertex z' on the path from z to r, the distance between u and v might be contracted or expanded again, when calculating the embedding of $X_{z'}$. In order to avoid accumulation of distortion, we would like to alternate the contractions and expansions of this distance in an appropriate way. To this end, we calculate, for each vertex v, a value $g(v) \in \{-1, 1\}$. Let $u_1, \ldots, u_k$ be the children of v, and let $Q'_1, \ldots, Q'_k$ be their corresponding squares. If $g(v) = 1$, then when embedding squares $Q'_1, \ldots, Q'_k$ into square Q of side a_v, we expand them along axis x and contract along axis y. If $g(v) = -1$, we do the opposite. The values of $g(v)$ have to be computed in a top-bottom fashion. They are calculated in such a way that the total distortion of distance between any pair of points in X stays below $poly(c')$.

For any non-root vertex u in T, with parent a vertex v, we define $s_u = a_v/b_u$. Also, for the root r of T, let $s_r = 1$.

Lemma 23. *For each vertex u, $1 \leq s_u \leq 32c'$.*

Proof. If u is the root, then $s_u = 1$. Otherwise, let $u, v \in T$, such that v is the father of u. We have already observed that a_v^2 is the sum of $b_{u_j}^2$, for all children u_j of v. Thus, $s(u) \geq 1$ holds.

Recall now that by the definition of b_u, its value is at least $\frac{l(v)}{4}$. On the other hand, by Corollary 2, $c' \geq \frac{a_v}{l(v)\sqrt{\pi}} - 1$, and thus $a_v \leq (c'+1)\sqrt{\pi}l(v) \leq 8c'l(v)$. Therefore, $s_u = \frac{a_v}{b_u} \leq 32c'$. $\square$

Let v be some non-leaf vertex, and let $u_1, \ldots, u_k$ be its children. Let $Q'_1, \ldots, Q'_k$ be the squares of side $b_{u_1}, \ldots, b_{u_k}$, respectively, corresponding to the children. In order to pack these squares into a square of side a_v, we transform each square Q'_j into a rectangle with sides $b_{u_j}s_j, \frac{b_{u_j}}{s_j}$. The goal of the next lemma is to calculate the values $g(v) \in \{-1, 1\}$ for each $v \in V$, that will determine, along which axis we contract, and along which expand when embedding the subtree of v.

Suppose we have a function $g : V(T) \to \{-1, 1\}$. Consider some vertex $v \in V(T)$, and let $v_1, v_2, \ldots, v_k$ be the vertices on the path from v to r, where $v_1 = r$, $v_k = v$. We define $h(v) = \prod_{j=1}^{k-1} s_{v_{j+1}}^{g(v_j)}$.

Lemma 24. *We can calculate, in linear time, function $g : V(T) \to \{-1, 1\}$, such that for each $v \in V(T)$, $\frac{1}{32c'} \leq h(v) \leq 32c'$.*

Proof. Observe first that in order to be able to calculate $h(v)$ for any $v \in V$, it is enough to know the values of $g(v')$ of all the vertices v' on the path from r to v, not including v.

We traverse the tree in the top-bottom fashion. For root r, we set $g(r) = 1$. Since for all the values s_v, $1 \leq s_v \leq 32c'$ holds, we have that for each level-2 vertex v, $\frac{1}{32c'} \leq h(v) \leq 32c'$ holds, as required.

Consider now some vertex $v \in V$ at level k, where $k \geq 2$. Let $v_1, v_2, \ldots, v_k$ be the vertices on the path from r to v, where $v_1 = r$, and $v_k = v$, and assume we have calculated $g(v_1), \ldots, g(v_{k-1})$, such that for each $j : 2 \leq j \leq k$, $\frac{1}{32c'} \leq h(v_j) \leq 32c'$ holds. We set $g(v) = 1$ if $h(v_k) \leq 1$, and we set $g(v) = -1$ otherwise. Let u be a child of v. Since $h(u) = h_v \cdot s_u^{g(v)}$, and $s_u \leq 32c'$, the inequality $\frac{1}{32c'} \leq h(u) \leq 32c'$ holds.

It is easy to see that the running time of the above algorithm is linear, if the values $h(v)$ of the vertices calculated by the algorithm are stored in a table. The algorithm traverses each vertex only once, and for each vertex v the calculation of $h(v)$ and $g(v)$ takes only constant time. $\square$

4.5.1 Algorithm description

The algorithm consists of two phases. The first phase is pre-processing, and the second phase is computing the embedding itself.

Phase 1: Preprocessing In this phase we translate the input ultrametric M into a 2-HST T, and calculate the values a_v, b_v, s_v, $g(v)$ for each vertex $v \in T$. Each one of these operations takes time linear in the input size.

Phase 2: Computing the embedding The algorithm works in a bottom-up fashion. For any vertex v in tree T, we produce an embedding of vertices X_v inside a square of side a_v. We start from level-h vertices (the leaves). Let v be such vertex. Then $a_v = \sqrt{C(v)} = \sqrt{\pi/4}$. We embed this point in the center of a square with a side of length $\sqrt{\pi/4}$.

Consider some level-i vertex v, for $1 \leq i < h$, and let $u_1, \ldots, u_k$ be its children. We assume that for each $j : 1 \leq j \leq k$, we have calculated the embeddings of u_j into a square Q_j of side a_{u_j}. We convert this square into a rectangle R_j, as follows. First, we add an empty strip of width $\frac{\sqrt{\pi} l(v)}{8}$ along the border of Q_j, so that now we have a new square Q'_j of side b_{u_j}. If $g(v) = 1$, then we expand the square along axis x and contract it along axis y by the factor of s_{u_j}. Otherwise, we expand square Q'_j along axis y and contract it along axis x by the factor of s_{u_j}. Notice that by the definition of s_{u_j}, the length of the longer side of R_j is precisely a_v. As the volume of R_j equals to the volume of Q'_j, and since $a_v^2 = \sum_{j=1}^{k} b_{u_j}^2$, we can pack all the rectangles next to each other inside a square Q of side a_v, with their longer side parallel to the x-axis if $g(v) = 1$, and to y-axis otherwise.

4.5.2 Analysis

The goal of this section is to bound the distortion produced by the algorithm. We first bound the maximum contraction, and then the maximum expansion of distances.

Lemma 25. *For any $u, u' \in X$, the distance between the images of u and u', is at least $\Omega(1/c')D(u,u')$.*

Proof. Let v be the least common ancestor of u, u'.

Let z, z' be the children of v, to whose subtrees vertices u, u' belong, respectively. Let Q, Q' be the squares into which X_z, and $X_{z'}$ are embedded, respectively, and let R, R' be the corresponding rectangles. Recall that we have added a strip of width at least $\frac{\sqrt{\pi}l(v)}{4}$ to squares Q, Q', and then stretched the new squares by a factors of $s(z), s(z')$, respectively. Without loss of generality, we can assume $s(z) \geq s(z')$. Therefore, immediately after computing the embedding for X_v, there is a strip S of width at least $\frac{l(v)}{4s(z)}$ between the rectangles R, R'. The width of strip S in the final embedding is a lower bound on the distance between the images of u and u'. Let $v_1, \ldots, v_k$ be the vertices on the path from r to v, where $v_1 = r, v_k = v$. Let $u_{k+1} = z$. If $g(v) = 1$, then strip S is horizontal, and thus for each $j : 1 \leq j \leq k-1$, if $g(v_j) = 1$ then its width decreases by the factor of $s(v_{j+1})$, and if $g(v_j) = -1$ then its width increases by the same factor. Thus, the final width of S is at least: $\frac{l(v)}{4s(z)^{g(v)}} \prod_{j=1}^{k-1} s(v_{j+1})^{-g(v_j)} = \frac{l(v)}{4} \prod_{j=1}^{k} s(v_{j+1})^{-g(v_j)} \geq \frac{l(v)}{4h(z)} \geq \frac{l(v)}{128c'}$.

If $g(v) = -1$, then strip S is vertical, and thus for each $j : 1 \leq j \leq k-1$, whenever $g(v_j) = 1$, the width of the strip grows by the factor of $s(v_{j+1})$, and whenever $g(v_j) = -1$, this width decreases by the same factor. Thus, in this case, the final width of S is at least: $\frac{l(v)}{4} s(z)^{g(v)} \prod_{j=1}^{k-1} s(v_{j+1})^{g(v_j)} = \frac{l(v)}{4} \prod_{j=1}^{k} s(v_{j+1})^{g(v_j)} \geq \frac{l(v)}{128c'}$.

As $D(u,u') = l(v)$, this concludes the proof of the lemma. □

Lemma 26. *For any $u, u' \in X$, the distance between the images of u and u', is at most $O(c'^2)D(u,u')$.*

Proof. Let v be the least common ancestor of u, u'. Then $D(u,u') = l(v)$. Following Corollary 2, $c' \geq \sqrt{\frac{C(v)}{\pi}}/l(v) - 1$, and thus $a_v \leq (c'+1)\sqrt{\pi}l(v) \leq 4c'l(v)$.

When calculating the embedding of X_v, all the vertices in X_v were embedded inside a square A whose side is $a_v \leq 4c'l(v) = O(c'D(u,u'))$.

After computing the final embedding, A is mapped to a rectangle A', which is obtained from A by expanding by a factor of γ along one axis, and by expanding by a factor of $1/\gamma$ along the other axis. If $v_1, \ldots, v_k$ are all the vertices along the path from the root $r = v_1$ to $v = v_k$, then $\gamma = \prod_{j=1}^{k-1} s(v_{j+1})^{g(v_j)} = h(v)$. Thus, by Lemma 24, γ is at least $\Omega(1/c')$, and at most $O(c')$. It follows that the diameter of A' is at most $O(c'^2 D(u,u'))$. Since the images of u and u' in the final embedding are contained inside A', the assertion follows. $\square$

The following result is now immediate:

Theorem 8. *Given an ultrametric M that c-embeds into the Euclidean plane, we can compute in linear time an embedding of M into the Euclidean plane with distortion $O(c^3)$.*

Observe that for $d = 2$, Theorem 7 provides an $O(\sqrt{n})$-distortion embedding. Combining this with our $O(c^3)$-distortion algorithm we obtain the following result:

Theorem 9. *There is an efficient $O(n^{1/3})$-approximation algorithm for minimum distortion embedding of ultrametrics into the plane.*

Proof. Let c be the optimal distortion achievable by any embedding of the input ultrametric into the plane. If $c > n^{1/6}$ then the above algorithm, which produces an $O(\sqrt{n})$-distortion embedding is an $O(n^{1/3})$-approximation. Otherwise, if $c \leq n^{1/6}$, then the algorithm from Section 4.5 gives $O(c^2) = O(n^{1/3})$-approximation. $\square$

4.6 Approximation algorithm for embedding ultrametrics into higher dimensions

In this section we extend the techniques used in Section 4.5, to obtain an approximation algorithm for embedding ultrametrics into ℓ_2^d.

Given an ultrametric $M = (X, D)$ that embeds into ℓ_2^d with distortion c, we first embed M into a 2-HST $M' = (X, D')$. Let T be the labeled tree associated with M', as in Section 4.5. Then M' embeds into ℓ_2^d with distortion $c' = O(c)$. We now focus on finding an embedding of M' into the ℓ_2^d with distortion at most $c'^{O(d)}$. The same embedding is an $c^{O(d)}$-distortion embedding of M into ℓ_2^d. We compute an embedding of M' into ℓ_2^d by recursively embedding the subtrees of vertices in a bottom-up fashion.

For any vertex u in the tree, let $a_u = (C(u))^{1/d}$. If u is a non-root vertex, let v be the father of u in T. We set $b_u = a_u + (V_d(l(v)/4))^{1/d}$, and $s_u = a_v/b_u$. If u is the root of the tree, we set $s_u = 1$.

Given a vertex v in the tree, we embed the vertices in X_v into a hypercube of side a_v, recursively. Let $u_1, \ldots, u_k$ be the children of v, and assume that for each $i \in [k]$, we are given an embedding of X_{u_i} into a d-dimensional hypercube Q_{u_i} of side length a_{u_i}. We define an additional hypercube Q'_{u_i} of side length b_{u_i} that has the same center as Q_{u_i} (i.e., Q'_{u_i} is obtained from Q_{u_i} by adding a "shell" of width $(V_d(l(v)/4))^{1/d}/2$ around Q_{u_i}). Let Q_v be a d-dimensional hypercube of side length a_v.

Note that the volume of Q_v equals the sum of volumes of Q'_{u_i}, for $1 \le i \le k$. This is since the volume of Q_v is $a_v^d = C(v)$, while the sum of volumes of $Q'_{u_i}, 1 \le i \le k$ is

$$\sum_{i=1}^{k} b_{u_i}^d = \sum_{i=1}^{k} \Big((C(u_i))^{1/d} + (V_d(l(v)/4))^{1/d} \Big)^d = C(v).$$

Fix one coordinate $j \in [d]$. We now show how to embed the hypercubes $Q'_{u_1}, \ldots, Q'_{u_k}$ into Q_v. Consider some hypercube $Q'_{u_i} : 1 \le i \le k$. For each dimension $j' \neq j$, we increase the length of the corresponding side of Q'_{u_i} by the factor of s_{u_i}. Additionally, we decrease the length of the side of Q'_{u_i} corresponding to the dimension j by the factor of $s_{u_i}^{d-1}$. Let R_i denote the resulting parallelepiped. Notice that for each dimension $j' \neq j$, the length of the corresponding side of parallelepiped R_i is exactly a_v. Moreover, the volume of R_i equals the volume of Q'_{u_i}. Therefore, we can easily pack the parallelepipeds R_i, $1 \le i \le k$, inside the hypercube Q_v, where the shortest side of R_i is placed along dimension j.

As in the algorithm for embedding ultrametrics into the plane, we need to ensure that these stretchings do not accumulate as we go up the tree. To ensure this, we calculate, for each vertex v a value $g(v) \in [d]$. When calculating the embedding of the hypercubes $Q'_{u_1}, \ldots, Q'_{u_k}$ into the hypercube Q_v, we contract the hypercubes $Q'_{u_1}, \ldots, Q'_{u_k}$ along the dimension $g(v)$ and expand them along all the other dimensions.

Our next goal is to prove an analogue of Lemma 24, that shows how to calculate the values $g(v)$ so that the total distortion is not accumulated.

We start with the following claim:

Claim 20. *For each vertex u of the tree, $1 \leq s_u \leq 8c'$.*

Proof. If u is the root of the tree, then $s_u = 1$ and the claim is trivially true. Assume now that u is not the root, and let v be its father. We denote the children of v by $u_1, \ldots, u_k$, and we assume that $u = u_i$ for some $i \in [k]$.

Recall that $s_u = a_v/b_u$, and that we have already observed that $a_v^d = \sum_{j=1}^k b_{u_j}^d$, and thus $s_u \geq 1$ clearly holds.

We now prove the second inequality. For the sake of convenience, we denote $V = (V_d(l(v)/4))^{1/d}$. Recall that $b_u = a_u + V \geq V$.

On the other hand, from Corollary 2,

$$c' \geq \rho_d(C(v))/l(v) - 1$$

Therefore, we have that

$$\rho_d(C(v)) = \left(\frac{C(v)\Gamma(1+d/2)}{\pi^{d/2}}\right)^{1/d} \leq 2c'l(v)$$

and thus

$$a_v = C(v)^{1/d} \leq 2c'l(v)\left(\frac{\pi^{d/2}}{\Gamma(1+d/2)}\right)^{1/d} = 8c'V$$

Therefore, $s_u = a_v/b_u \leq 8c'V/V \leq 8c'$. $\square$

For each vertex u of the tree, for each dimension $j \in [d]$, we recursively define a value $h_j(u)$, as follows. If u is the root, then $h_j(u) = 1$ for all $j \in [d]$. Consider now some vertex u which is not the root, and let v be its father. Then we define $h_j(u) = h_j(v) \cdot s_u^{\alpha_j(v)}$, where $\alpha_j(v)$ is defined to be 1 if $j \neq g(v)$, and it is defined to be $-(d-1)$ if $i = g(v)$. Notice that $\prod_{j\in[d]} h_j(u) = 1$.

Fix any vertex $u \in V(T)$ and any dimension $j \in [d]$. Let Q_u be the hypercube of side a_u into which the vertices of X_u have been embedded when u was processed by the algorithm. Then the value $h_j(u)$ is precisely the stretch along the dimension j of Q_u in the final embedding. In other words, if we take a pair of points $x, y \in Q_u$ such that $x_j = y_j - 1$, and for all the other coordinates j', $x_{j'} = y_{j'}$, then $h_j(u)$ is precisely the distance between x and y in the final embedding. We next prove that we can calculate the values $g(v)$ in a way that ensures that that for each vertex u and for each dimension $j \in [d]$, $h_j(u)$ lies between $(O(1/c'))^d$ and $(O(c'))^d$.

Lemma 27. *We can compute in polynomial time values $g(u)$ for all $u \in V(T)$, such that for each $u \in V(T)$, for each dimension $j \in [d]$, $(O(1/c'))^d \leq h_j(u) \leq (O(c'))^d$.*

Proof. If u is the root, then we arbitrarily set $g(u) = 1$.

Consider now some non-root vertex u, and let v be its parent. Let $j \in [d]$ be the dimension for which $h_j(v)$ is maximized. Then we set $g(u) = j$.

Claim 21. *For every vertex u, $\frac{\max_i\{h_i(u)\}}{\min_i\{h_i(u)\}} \leq (8c')^d$.*

Proof. The claim is trivially true for the root r since $\frac{\max_i\{h_i(r)\}}{\min_i\{h_i(r)\}} = 1$. For any non-root vertex u, assume that the claim is true for its parent v. Assume w.l.o.g. that $h_1(v) \geq h_2(v) \geq \cdots \geq h_d(v)$, and $g(u) = 1$. Then $h_1(u) = h_1(v)/s_u^{d-1}$, and for each $i > 1$, $h_i(u) = h_i(v) \cdot s_u$. There are three cases to consider. If $h_1(u)$ equals the maximum value among $\{h_i(u)\}_{i=1}^d$, then clearly $\frac{\max_i\{h_i(u)\}}{\min_i\{h_i(u)\}} \leq \frac{\max_i\{h_i(v)\}}{\min_i\{h_i(v)\}} \leq (8c')^d$ by the induction hypothesis. If $h_1(u)$ equals the minimum value among $\{h_i(u)\}_{i=1}^d$, then $\frac{\max_i\{h_i(u)\}}{\min_i\{h_i(u)\}} = \frac{h_2(u)}{h_1(u)} = \frac{s_u^d h_2(v)}{h_1(v)} \leq s_u^d$. Finally, if neither of the above two cases happens, then $\frac{\max_i\{h_i(u)\}}{\min_i\{h_i(u)\}} = \frac{h_2(u)}{h_d(u)} = \frac{h_2(v)s_u}{h_d(v)s_u} \leq (8c')^d$ by the induction hypothesis. □

Since $\prod_{i=1}^d h_i(u) = 1$, we get that $(O(c'))^{-d} \leq h_i(u) \leq (O(c'))^d$.

It is easy to see that the algorithm for computing the values $g(u)$, runs in polynomial time. □

Let $f : X \to \mathbb{R}^d$ denote the resulting embedding produced by the algorithm. The next two lemmas bound the maximum contraction and the maximum expansion of the distances in this embedding.

Lemma 28. *For any pair $u, u' \in X$ of points, $\|f(u) - f(u')\|_\infty \geq (O(c'))^{-d} D'(u, u')$.*

Proof. Fix any pair $u, u' \in X$ of vertices, and let v be their least common ancestor in the tree T. Thus, $D'(u, u') = l(v)$. Let z, z' be the children of v such that $u \in X_z$ and $u' \in X_{z'}$. Assume w.l.o.g. that $s_z > s_{z'}$. Recall that $Q'_z, Q'_{z'}$ contain empty shell of width $(V_d(l(v)/4))^{1/d}/2$ in which no vertices are embedded. When $Q'_z, Q'_{z'}$ are embedded inside Q_v, they are contracted by the factors $s_z, s_{z'}$ respectively along the ith dimension, where $i = g(v)$. Thus, in the embedding of X_v inside Q_v, the distance between the images of u and u' along the ith dimension is at least:

$$\frac{V_d(l(v)/4)}{s_u^{d-1}} = \frac{\sqrt{\pi} l(v)}{4(\Gamma(1 + d/2))^{1/d} s_u^{d-1}} \geq \frac{l(v)}{2^{O(\log d)} s_u^{d-1}}$$

In the final embedding this distance is multiplied by the factor $h_i(v)$. Thus, the final distance is at least

$$\frac{l(v)}{2^{O(\log d)} s_u^{d-1}} h_i(v) = \frac{l(v)}{2^{O(\log d)}} h_i(u) \geq \frac{l(v)}{(O(c'))^d}$$

□

Lemma 29. *For any pair $u, u' \in X$ of points, $\|f(u) - f(u')\|_\infty \leq (O(c'))^{d+1} D'(u, u')$.*

Proof. Fix any pair $u, u' \in X$ of vertices, and let v be their least common ancestor in the tree T, so that $D'(u, u') = l(v)$.

Recall that Q_v is a hypercube of side a_v, and thus when the embedding of X_v has been computed, the distance between the images of u and u' was at most a_v. In the final embedding this distance increased by the factor of at most $\max_{i \in [d]}\{h_i(v)\} \leq (O(c'))^d$, and thus the final distance is at most $a_v (O(c'))^d$. From Corollary 2, using the same reasoning as in the proof of Claim 20, we have that

$$a_v \leq 2c' l(v) \frac{\sqrt{\pi}}{(\Gamma(1 + d/2))^{1/d}} \leq O(c') l(v)$$

Thus, $\|f(u) - f(u')\|_\infty \leq (O(c'))^{d+1}\, l(v)$. $\square$

Combining the results of Lemma 28 and Lemma 29, we obtain the following.

Theorem 10. *For any $d > 2$, there is a polynomial time algorithm that embeds any input ultrametric M into ℓ_2^d with distortion $c^{O(d)}$, where c is the optimal distortion of embedding M into ℓ_2^d.*

Chapter 5

Improved embeddings of ultrametrics into $\mathbb{R}^d$

In this chapter we give an improved approximation algorithm for embedding ultrametrics into $\mathbb{R}^d$. More precisely, we present an algorithm which for any fixed $d \geq 2$, given an ultrametric M that c-embeds into $\mathbb{R}^d$, computes an embedding of M into $\mathbb{R}^d$ with distortion at most $O(c \cdot \log^{O(1)} \Delta)$. The previous algorithm from Theorem 10 would yield distortion $c^{O(1)}$. Strictly speaking, the two algorithms are incomparable, since the $c^{O(1)}$ bound is better when c is very small (e.g. $c = O(1)$). However, the algorithm presented here is the first one achieving distortion with linear dependence on the optimal.

This new guarantee is obtained using new hierarchical partitioning schemes of the Euclidean space, called *circular partitions*, matching up to a poly-logarithmic factor the lower bound given by Corollary 2. Such a partition consists of a hierarchy of convex polygons, each having small aspect ratio, and satisfying specified volume constraints.

We also apply these partitions to obtain a natural extension of the popular Treemap visualization method. Our proposed algorithm is not constrained in using only rectangles, and can achieve provably better guarantees on the aspect ratio of the constructed polygons.

The results presented in this chapter are from [48].

5.1 The Treemap algorithm

The visualization of hierarchical structures is a fundamental problem in graph drawing, and computer graphics in general. One of the most successful practical algorithms for this problem, that has attracted a lot of attention over the past years, is Treemap [57]. More precisely, one is given a hierarchy of elements represented as a rooted tree with positive weights on its leaves. The weight of each internal vertex is the sum of the weights of the leaves in its subtree. Treemap assigns a rectangle to each vertex such that:

- the area of the rectangle is equal to the weight of the vertex;
- the rectangles of the children of each internal vertex v are disjoint, and are contained inside the rectangle of v.

An extension of Treemap The most important goal of the plane partition computed by Treemap is the minimization of the aspect ratio of each rectangle. However, it is easy to construct instances where the aspect ratio of any such rectangular assignment is unbounded. For example, consider a tree with a root and two leaves, where the first leaf has weight 1, and the second has weight L. The optimal aspect ratio of Treemap in this case is unbounded as $L \to \infty$. This simple observation leads to the following natural question:

> Is there a hierarchical partitioning of the plane into convex polygons that achieves aspect ratio independent of the weights?

We answer this question in the affirmative. More precisely, we present an algorithm that given an n-vertex tree of depth d, outputs a partitioning into convex polygons, each having aspect ratio $O(\mathrm{poly}(d, \log n))$.

We remark that the problem of modifying Treemap so that it uses only sets of small aspect ratio has been considered in [16, 8, 7, 60]. However, our work provides the first provable guarantees on the aspect ratio.

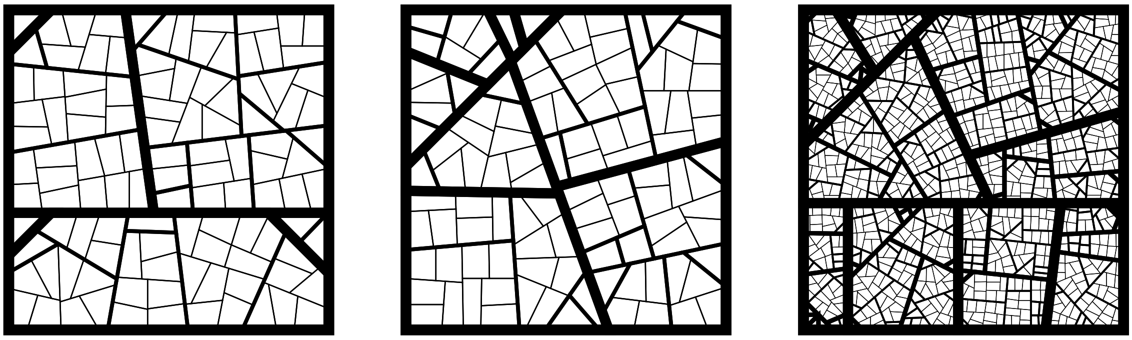


Figure 5-1: Hierarchical partitions computed by the modified Treemap algorithm on synthetic data. Thicker boundaries correspond to higher levels of the partition.

Figure 5-1 depicts partitions computed by our algorithm on synthetic hierarchical data. It would be interesting to compare our algorithm with existing implementations of Treemap, on real data.

Furthermore, if it is required that all polygons assigned to vertices of the tree be rectangles, we show that it is possible to construct a relaxed partition with small aspect ratio, that we call a *rectangular partition with slack*. The difference from the standard partition is that the area of the rectangle assigned to an internal vertex can exceed the sum of the areas of the rectangles assigned to its children by a factor of at most $1 + \varepsilon$.

Previous work on Treemap The *Treemap* algorithm was proposed by Shneiderman [57], and its first efficient implementation was given by Johnson and Shneiderman [34]. There have been several improvements of the original algorithm. Bruls et al. [16] proposed a variant of Treemap that heuristically tries to minimize the aspect ratio of the resulting rectangles. Shneiderman and Wattenberg [58] have proposed a modified algorithm that minimizes the aspect ratio while preserving certain ordering constraints of the rectangles of the children of each vertex. The quality of the representation of a partition has been further improved by van Wijk and van de Wetering [61], who developed a method for displaying the rectangles using more intuitive shading.

Voronoi treemaps [8, 7] are probably the most closely related to ours. The algorithm is not limited to output a partitioning of the plane into rectangles, but is allowed to output arbitrary, even nonconvex objects. Partitioning of an area is done as follows. First a set S of points that correspond to subtrees is placed within the area. Then, each point of the area is assigned to the closest point in S, where the distance function is modified for each point p in S according to the weight of the subtree corresponding to p. An iterative process is used to optimize the placement of points, and the size of an area assigned to a point may slightly differ from the expected. A version of Voronoi treemaps provides a partitioning into polygons. As opposed to the partitioning scheme discussed here, Voronoi treemaps are not known to give any theoretical guarantees on aspect ratios of computed areas.

Another proposed extension of Treemap to non-rectangular objects are *circular treemaps* [66], which use circles instead of rectangles. Circular treemaps are visually appealing, and nicely display nesting, but a lot of space may be wasted in the process of partitioning a circle into smaller circles.

Extensions of Treemap for visualization in 3-dimensional space have been considered by Rekimoto and Green [53], Bladh et al. [14], and Bladh et al. [13]. A variant of Treemap that constructs radial partitions was proposed by Stasko et al. [59].

The Treemap algorithm has been used to visualize a wide range of hierarchical data, including stock portfolios [36], news items [65], blogs [64], business data [63], tennis matches [33], photo collections [12], and file-system usage [57, 66].

Shneiderman maintains a webpage [56] that describes the history of his invention. It gives an overview of applications and proposed extensions to his original idea.

5.1.1 Preliminaries

For a set $A \subset \mathbb{R}^d$, let $\mathrm{Vol}(A)$, and $\mathrm{diam}(A)$ denote the d-dimensional volume, and the diameter of A, respectively. We define the *aspect ratio* of a polygon A to be $\lambda(A) = \frac{\mathrm{diam}(A)^2}{\mathrm{Vol}(A)}$.

For a d-dimensional hyperrectangle R of sides $s_1, s_2, \ldots, s_d \in \mathbb{R}_+$, the *rectangular aspect ratio* $\lambda_{\mathrm{rect}}(R)$ of R equals $\frac{\max_i s_i}{\min_i s_i}$. It can easily be shown that for 2-dimensional rectangles, the aspect ratio and the rectangular aspect ratio are within a constant factor.

5.2 Hierarchical circular partitions of $\mathbb{R}^2$

We show an algorithm that constructs a partition of the plane that reflects properties of a tree with weights $w(\cdot)$ assigned to its vertices. There is a 1-to-1 correspondence between the polygons in the partition and the vertices of the tree, and each polygon has volume equal to the weight of the corresponding vertex.

Throughout this chapter, we will refer to this partition as *hierarchical circular partition.* We call it "hierarchical" because if a vertex v is a descendant of another vertex u, then the polygon corresponding to v is contained inside the polygon corresponding to u. Furthermore, if two vertices are not in the ancestor-descendant relation in the tree, the interiors of the polygons corresponding to these two vertices are disjoint. The term "circular" is used because we require all the polygons to have small aspect ratio. Intuitively, if a polygon has small aspect ratio, it is close to a circle. The main technical difficulty that we face is showing that the aspect ratios of all polygons in our partition are small.

A formal specification of all the desired properties of such a partition follows. We write $\mathcal{P}(S)$ to denote the power set of S, i.e., the set of all subsets of S.

Definition 2 (γ-Hierarchical Circular Partition)**.** *Let $T = (V, E)$ be a rooted tree with n leaves, and depth d. Let $w : V \to \mathbb{R}_{\geq 0}$ be a function such that for any internal vertex $v \in V(T)$, with children $u_1, \ldots, u_k$, $w(v) \geq \sum_{i=1}^{k} w(u_i)$. Then, for some $\gamma > 0$, a* γ-hierarchical circular partition *for (T, w) is a mapping $f : V(T) \to \mathcal{P}(\mathbb{R}^2)$, such that:*

- *For each $v \in V(T)$, $f(v)$ is a convex polygon in $\mathbb{R}^2$ with $\lambda(f(v)) \leq \gamma$.*
- *For each $v \in V(T)$, $\mathrm{Vol}(f(v)) = w(v)$.*
- *For each $u, v \in V(T)$, such that u is the parent of v in T, $f(v) \subseteq f(u)$.*
- *For each $u, v \in V(T)$, such that u is not an ancestor of v, and v is not an ancestor of u, int$(f(u)) \cap$ int$(f(v)) = \emptyset$.*

5.2.1 Existence of a good cut

The main component of a proof that hierarchical circular partitions with good properties exist will be the following lemma. It shows that there is always a way to cut a polygon into two smaller polygons of required volumes so that the aspect ratios of the new polygons are bounded. The proof of the lemma is long and consists of a case analysis.

Lemma 30 (Circular Cut)**.** *Let $P \subset \mathbb{R}^2$ be a convex polygon with k vertices, and aspect ratio $\lambda(P)$, and let $a \in (0, 1/2]$. Then, P can be partitioned into two convex polygons P_1, and P_2, such that*

- *Each of the P_1, and P_2 has at most $k+1$ vertices.*
- $\mathrm{Vol}(P_1) = a \cdot \mathrm{Vol}(P)$, *and* $\mathrm{Vol}(P_2) = (1-a) \cdot \mathrm{Vol}(P)$.
- *The aspect ratio of each of the P_1, P_2 is at most*

$$\max\{\lambda(P_1), \lambda(P_2)\} \leq \max\left\{\lambda(P)\left(1 + \frac{6}{k}\right), k^8\right\}$$

Proof. We distinguish between the following two cases.

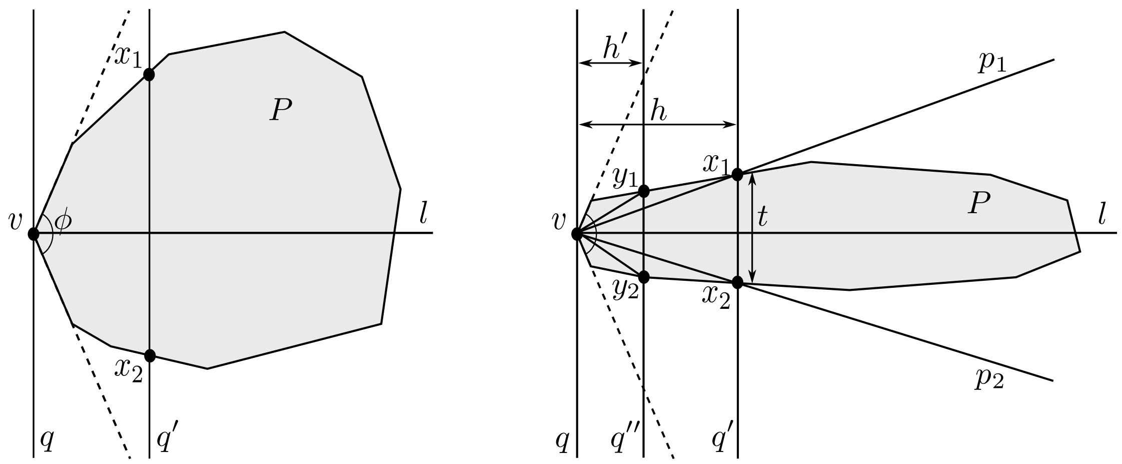
Case 1: $a \leq 1/k^2$. Let ϕ be the smallest angle of P, and let v be a vertex of P, incident to an angle ϕ. Since P has k vertices, we have

$$\phi \leq \pi\left(1 - \frac{2}{k}\right)$$

Let l be the bisector of ϕ, and let q be the line normal to l. Let S be the halfplane with boundary q, such that $S \cap P = v$. Consider the translation S' of S, such that

$$\mathrm{Vol}(S' \cap P) = a \cdot \mathrm{Vol}(P)$$

Let also q' be the boundary of S'. We define $P_1 = S' \cap P$, and $P_2 = \mathrm{cl}(P \setminus S')$. Clearly, P_1, and P_2 are convex polygons with at most $k+1$ vertices each, such



(a) Case 1.

(b) Case 1.2.

Figure 5-2: Partitioning P into P_1, and P_2, when $\alpha \leq 1/k^2$.

that $\mathrm{Vol}(P_1) = a \cdot \mathrm{Vol}(P)$, and $\mathrm{Vol}(P_2) = (1-a)\cdot \mathrm{Vol}(P)$. Therefore, it remains to bound the aspect ratios of P_1, and P_2.

Since $P_2 \subset P$, we have

$$\begin{aligned}
\lambda(P_2) &= \frac{\mathrm{diam}(P_2)^2}{\mathrm{Vol}(P_2)} \leq \frac{\mathrm{diam}(P)^2}{(1-a)\cdot \mathrm{Vol}(P)} = \frac{\lambda(P)}{1-a} \\
&< \lambda(P)\,(1+2a) < \lambda(P)\left(1+\frac{2}{k^2}\right) \\
&< \lambda(P)\left(1+\frac{1}{k}\right).
\end{aligned}$$

We next bound $\lambda(P_1)$. Let x_1, x_2 be the two points where q' intersects ∂P, and let t be the distance between x_1, and x_2. Let h be the distance between the lines q and q'. Figure 5-2(a) depicts the arrangement. We distinguish between the following cases.

Case 1.1: $t \geq h/k^2$. Since P is convex, the triangle vx_1x_2 is contained in P_1. Therefore, $\mathrm{Vol}(P_1) \geq h \cdot t/2 \geq h^2/(2k^2)$. On the other hand, since S' is normal to the bisector of the angle of v, it follows that P_1 is contained inside a rectangle of width h, and height H, with

$$\begin{aligned} H &\leq 2 \cdot h \cdot \tan(\phi/2) \leq 2 \cdot h \cdot \tan\left(\frac{\pi(1-2/k)}{2}\right) \\ &\leq 2 \cdot h / \tan(\pi/k) \leq 2 \cdot h \cdot k/\pi \end{aligned}$$

Thus, $\operatorname{diam}(P_1) < h(1 + 2 \cdot k/\pi)$. It follows that

$$\lambda(P_1) = \frac{\operatorname{diam}(P_1)^2}{\operatorname{Vol}(P_1)} < \frac{(h + 2 \cdot h \cdot k/\pi)^2}{h^2/(2k^2)} < k^5$$

Case 1.2: $t < h/k^2$. Let p_1 be the line passing through v, and x_1, and let p_2 be the line passing through v, and x_2. Let γ be the angle between p_1, and p_2. Observe that P_2 is contained between p_1 and p_2. Therefore, there exist a point $u \in P_2$, such that

$$\frac{\gamma}{2\pi} \pi \|u - v\|_2^2 \geq \operatorname{Vol}(P_2)$$

It follows that $\operatorname{diam}(P)^2 \geq \|u - v\|_2^2 \geq \frac{2}{\gamma}(1-a)\operatorname{Vol}(P)$. Therefore,

$$\lambda(P) = \frac{\operatorname{diam}(P)^2}{\operatorname{Vol}(P)} \geq \frac{2}{\gamma}(1-a) \geq \frac{2}{\gamma}\left(1 - \frac{1}{k^2}\right)$$

We now give an upper bound on the diameter of P_1. Assume w.l.o.g. that $\|v - x_2\|_2 \geq \|v - x_1\|_2$, and let $R = \|v - x_2\|_2$. Consider a line q'', parallel to q, that lies between q and q'. Let h' be the distance between q and q''. The line q'' intersects ∂P_1 on two points y_1, y_2 (see Figure 5-2(b)). We will show that $\|y_1 - y_2\|_2 \leq 2t$. Assume for the sake of contradiction, that $\|y_1 - y_2\|_2 > 2t$. Let g_1 be the line passing through y_1, and x_1, and let g_2 be the line passing through y_2, and x_2. Observe that since $\|y_1 - y_2\|_2 > \|x_1 - x_2\|_2$, it follows that g_1, and g_2 intersect at a point w, such that P_2 is contained in the triangle x_1x_2w. Observe that the polygon $vy_1x_1x_2y_2$ is contained in P_1. If $h' \geq h/2$, then the volume of the triangle vy_1y_2 is greater or equal to the volume of the triangle x_1x_2w. Therefore,

$\mathrm{Vol}(P_1) \geq \mathrm{Vol}(P_2)$, contradicting the fact that $a \leq 1/k^2$. If on the other hand $h' < h/2$, then the volume of the quadrilateral $y_1x_1x_2y_2$, is greater than the volume of the triangle x_1x_2w, implying that $\mathrm{Vol}(P_1) \geq \mathrm{Vol}(P_2)$, a contradiction. Therefore, we obtain that $\|y_1 - y_2\|_2 \leq 2t$.

It now follows that any point $u \in P_1$ is at distance at most $2t$ from the line segment vx_2. Thus,

$$\begin{aligned} \mathrm{diam}(P_1) &= \max_{u,u' \in P_1} \|u - u'\|_2 \\ &\leq \max_{u,u' \in P_1} \{2t + \|v - x_2\|_2 + 2t\} \\ &\leq R + 4t \leq R\left(1 + \frac{4}{k^2}\right). \end{aligned}$$

Let x^* be the point on the line segment x_1x_2, that is closest to v. Since $R \geq h$, we have

$$\mathrm{Vol}(P_1) \geq \frac{\gamma}{2\pi}\pi\|v - x^*\|_2^2 \geq \frac{\gamma}{2}(R - t)^2 \geq \frac{\gamma}{2}R^2\left(1 - \frac{1}{k^2}\right)$$

Therefore,

$$\begin{aligned} \lambda(P_1) &= \frac{\mathrm{diam}(P_1)^2}{\mathrm{Vol}(P_1)} \leq \frac{2}{\gamma} \cdot \frac{(1 + 4/k^2)^2}{1 - 1/k^2} \\ &\leq \lambda(P)\frac{(1 + 4/k^2)^2}{(1 - 1/k^2)^2} \leq \lambda(P) \cdot (1 + 6/k^2)^2 \\ &\leq \lambda(P) \cdot (1 + 2/k)^2 \leq \lambda(P) \cdot (1 + 6/k) \end{aligned}$$

Case 2: $a > 1/k^2$.

Case 2.1: $\lambda(P) \leq k^6$. We pick an arbitrary half-plane H, such that $\mathrm{Vol}(P \cap H) = a \cdot \mathrm{Vol}(P)$. We set $P_1 = P \cap H$, and $P_2 = \mathrm{cl}(P \setminus H)$. Clearly, we have

$$\lambda(P_1) = \frac{\mathrm{diam}(P_1)^2}{\mathrm{Vol}(P_1)} \leq \frac{\mathrm{diam}(P)^2}{a \cdot \mathrm{Vol}(P)} \leq k^2 \cdot \lambda(P) \leq k^8$$

and

$$\lambda(P_2) = \frac{\text{diam}(P_2)^2}{\text{Vol}(P_2)} \leq \frac{\text{diam}(P)^2}{(1-a)\cdot\text{Vol}(P)} \leq 2\cdot\lambda(P) \leq 2\cdot k^6 < k^7$$

Case 2.2: $\lambda(P) > k^6$. Pick points $v_1, v_2 \in P$, such that $\|v_1 - v_2\|_2 = \text{diam}(P)$. Let ρ be the line passing through v_1, and v_2. Let also ν_1, and ν_2, be the lines normal to ρ, passing through v_1, and v_2 respectively. Note that P is contained between ν_1, and ν_2.

For each $z \in [0, \text{diam}(P)]$, let $\nu(z)$ be a line normal to ρ that is at distance z from ν_1, and at distance $\text{diam}(P) - z$ from ν_2. Define $f(z)$ to be the length of the intersection of P with $\nu(z)$. Observe that

$$\text{Vol}(P) = \int_{z=0}^{\text{diam}(P)} f(z)dz$$

Pick $s_1, s_2 \in [0, \text{diam}(P)]$, so that

$$a\cdot\text{Vol}(P) = \int_{z=0}^{s_1} f(z)dz = \int_{z=\text{diam}(P)-s_2}^{\text{diam}(P)} f(z)dz$$

Let Q_1 be the part of P that is contained between ν_1, and $\nu(s_1)$. Similarly, let Q_2 be the part of P that is contained between $\nu(\text{diam}(P)-s_2)$, and ν_2. Clearly, both Q_1, and Q_2 are convex polygons with at most $k+1$ vertices.

First, we will show that

$$\min\left\{\frac{\text{Vol}(Q_1)}{s_1}, \frac{\text{Vol}(Q_2)}{s_2}\right\} \leq \frac{\text{Vol}(P)}{\text{diam}(P)}$$

Assume for the sake of contradiction that $\frac{\text{Vol}(Q_1)}{s_1} > \frac{\text{Vol}(P)}{\text{diam}(P)}$, and $\frac{\text{Vol}(Q_2)}{s_2} > \frac{\text{Vol}(P)}{\text{diam}(P)}$. It follows that there exist $z_1 \in [0, s_1]$, and $z_2 \in [\text{diam}(P) - s_2]$, such that $f(z_1) > \frac{\text{Vol}(P)}{\text{diam}(P)}$, and $f(z_2) > \frac{\text{Vol}(P)}{\text{diam}(P)}$. Since P is convex, f is a bitonic function. Therefore, for each $z \in [z_1, z_2]$, $f(z) > \frac{\text{Vol}(P)}{\text{diam}(P)}$. It follows that

$$\mathrm{Vol}(P) = \mathrm{Vol}(Q_1) + \mathrm{Vol}(Q_2) + \mathrm{Vol}(P \setminus (Q_1 \cup Q_2)) > \frac{\mathrm{Vol}(P)}{\mathrm{diam}(P)} \cdot \mathrm{diam}(P),$$

a contradiction.

We can therefore assume w.l.o.g. that

$$\frac{\mathrm{Vol}(Q_1)}{s_1} \leq \frac{\mathrm{Vol}(P)}{\mathrm{diam}(P)}$$

Note that this implies

$$s_1 \geq a \cdot \mathrm{diam}(P)$$

We set $P_1 = Q_1$, and $P_2 = P \setminus Q_1$. It remains to bound $\lambda(P_1)$, and $\lambda(P_2)$.

By the convexity of P, $\mathrm{Vol}(P) \geq \max_{z\in[0,\mathrm{diam}(P)]} f(z) \cdot \mathrm{diam}(P)/2$. Since $\lambda(P) > k^6$, it follows that

$$\max_{z\in[0,\mathrm{diam}(P)]} f(z) < \frac{2}{k^6} \cdot \mathrm{diam}(P).$$

This implies that P is contained inside a rectangle with one edge of length $\mathrm{diam}(P)$ parallel to ρ, and one edge of length $\frac{4}{k^6} \cdot \mathrm{diam}(P)$ normal to ρ. Thus,

$$\mathrm{diam}(P_1) \leq s_1 + \frac{4}{k^6} \cdot \mathrm{diam}(P).$$

Let σ_1, σ_2 be the two points where $\nu(s_1)$ intersects ∂P. Let ζ_1, ζ_2, be the lines passing through v_1, and σ_1, σ_2 respectively. Let also σ_1', and σ_2', be the points where ζ_1, and ζ_2 respectively intersect ν_2 (see Figure 5-3). By the convexity of P and P_1, we have

$$\mathrm{Vol}(P_1) \geq \mathrm{Vol}(v_1\sigma_1\sigma_2) = \left(\frac{s_1}{\mathrm{diam}(P)}\right)^2 \cdot \mathrm{Vol}(v_1\sigma_1'\sigma_2') \geq \left(\frac{s_1}{\mathrm{diam}(P)}\right)^2 \cdot \mathrm{Vol}(P).$$

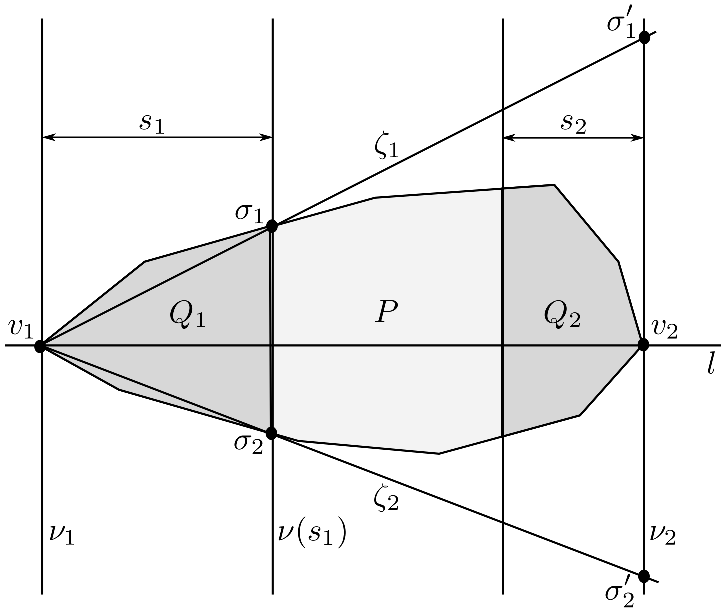


Figure 5-3: Partitioning P into P_1, and P_2, when $\alpha > 1/k^2$: Case 2.2.

Since $\mathrm{Vol}(P_1) = \alpha \cdot \mathrm{Vol}(P)$, it follows that $s_1 \leq \sqrt{\alpha} \cdot \mathrm{diam}(P)$. Therefore,

$$\begin{aligned}
\lambda(P_1) &= \frac{\mathrm{diam}(P_1)^2}{\mathrm{Vol}(P_1)} \\
&\leq \frac{(s_1 + 4 \cdot \mathrm{diam}(P)/k^6)^2}{\mathrm{Vol}(P_1)} \\
&\leq \frac{(\sqrt{\alpha} \cdot \mathrm{diam}(P) + 4 \cdot \mathrm{diam}(P)/k^6)^2}{\alpha \cdot \mathrm{Vol}(P)} \\
&< \frac{\mathrm{diam}(P)}{\sqrt{\mathrm{Vol}(P)}} \cdot (1 + 4/k^4)^2 \\
&\leq \lambda(P) \cdot (1 + 8/k^4 + 16/k^{16}) \\
&\leq \lambda(P) \cdot (1 + 1/k)
\end{aligned}$$

Since f is bitonic, it follows that

$$\min_{z \in [s_1, \mathrm{diam}(P) - s_2]} f(z) \geq \min\{ \max_{z \in [0, s_1]} f(z), \max_{z \in [\mathrm{diam}(P) - s_2, \mathrm{diam}(P)]} f(z)\}$$

Therefore, $\frac{\mathrm{Vol}(P_2)}{\mathrm{diam}(P)-s_1} \geq \frac{\mathrm{Vol}(P_1)}{s_1}$. We have

$$\mathrm{diam}(P_2) \leq \mathrm{diam}(P) - s_1 + \frac{4}{k^6} \cdot \mathrm{diam}(P)$$

Thus,

$$\begin{aligned}
\lambda(P_2) &= \frac{\mathrm{diam}(P_2)^2}{\mathrm{Vol}(P_2)} \\
&\leq \frac{(\mathrm{diam}(P)(1+4/k^6) - s_1)^2}{(1-a)\cdot \mathrm{Vol}(P)} \\
&\leq \lambda(P) \cdot \left(\frac{1+4/k^6 - a}{\sqrt{1-a}}\right)^2 \\
&\leq \lambda(P) \cdot (1 + 4\cdot\sqrt{2}/k^6)^2 \\
&\leq \lambda(P) \cdot (1 + 1/k^2)^2 \\
&\leq \lambda(P) \cdot (1 + 3/k^2) \\
&\leq \lambda(P) \cdot (1 + 1/k)
\end{aligned}$$

This concludes the proof. □

5.2.2 Circular partitions

Now we have all the necessary tools to prove that for any tree T, there exists a γ-hierarchical circular partition with γ polynomial in the depth of T and the logarithm of the number of leaves in T. Initially, we transform T into an equivalent balanced binary tree. For a binary tree, at each internal vertex we can split the polygon corresponding to it into two polygons corresponding to its children with a single cut. To determine the cut, we use Lemma 30, which yields that the aspect ratios of all the polygons will be bounded.

Lemma 31 (Existence of Hierarchical Circular Partitions)**.** *Let $T = (V, E)$ be a rooted tree with n leaves, and depth d. Let $w : V \to \mathbb{R}_{\geq 0}$ be a function such that for any interval vertex $v \in V(T)$, with children $u_1, \ldots, u_k$, $w(v) \geq \sum_{i=1}^{k} w(u_i)$. Then, there exists an $O\left((d \cdot \lg n)^{17}\right)$-hierarchical circular partition for (T, w).*

Proof. Let r be the root of T. We first construct a *binary* tree $T' = (V, E)$, such that $V(T) \subseteq V(T')$, and for each $u, v \in V(T)$, if u is an ancestor of v in T, then u is also an ancestor of v in T'. Clearly, this can be done as follows: For each non-leaf vertex $v \in V(T)$, we replace the set of edges connecting u with its children by a balanced binary tree of depth at most $\lceil \lg n \rceil$. The resulting tree has depth $d' \leq d \cdot \lceil \lg n \rceil$. We define weights w' of nodes in T' as follows. For each node $v \in V(T)$, we set $w'(v) = w(v)$. For each other node $v \in V(T') \setminus V(T)$, that was added to T' as a result of replacing the edges adjacent to a vertex u by a balanced binary tree, we set the value $w'(v)$ to be the sum of the weights of the children of u that are below v in T'. Note that for any node $v \in V(T')$, the sum of the weights of its children in T' is at most $w'(v)$.

We will define inductively a hierarchical circular partition f, starting from r. We set $f(r)$ to be a square in $\mathbb{R}^2$ of volume $w(r)$. Consider now a non-leaf vertex $v \in V(T')$ such that $f(v)$ has already been defined. The volume of the polygon $f(v)$ is $w'(v)$. Let t be the sum of the weights of the children of v in T'. Let P be the polygon obtained by uniform shrinking of $f(v)$ by a factor of $\sqrt{t/w'(v)}$ with any point inside $f(v)$ being a fixed point of the transformation. The volume of P equals t. If v has exactly one child u in T', then we simply set $f(u) = P$. Otherwise, let u_1, u_2 be the children of v in T'. Let $a = \frac{w'(u_1)}{w'(u_1)+w'(u_2)}$. Applying Lemma 30, we partition $f(v)$ into two convex polygons P_1, and P_2, such that $\mathrm{Vol}(P_1) = a \cdot \mathrm{Vol}(f(v)) = w'(u_1)$, and $\mathrm{Vol}(P_2) = (1-a) \cdot \mathrm{Vol}(f(v)) = w'(u_2)$. Moreover, we have $\max\{\lambda(P_1), \lambda(P_2)\} \leq \max\left\{\lambda(f(v))\left(1+\frac{6}{k}\right), k^8\right\}$. We set $f(u_1) = P_1$, and $f(u_2) = P_2$.

We would like to bound $\lambda(f(v))$, for each $v \in V(T)$. Since $f(r)$ is a square, we have that $\lambda(f(r)) = 2$. Consider now $v \in V(T')$. Let t be the distance between r and v in T'. Let p be the path from r to v in T', with $p = v_0, v_2, \ldots, v_t$, where $v_0 = r$, and $v_t = v$. Observe that for each $i \in \{0, \ldots, t\}$, $f(v_i)$ is a convex polygon with at most $i+4$ vertices. It follows by Lemma 30, that for each $i \in \{1, \ldots, t\}$,

$$\lambda(f(v_i)) \leq \max\left\{(i+3)^8, \lambda(f(v_{i-1})) \cdot \left(1 + \frac{6}{i+3}\right)\right\}.$$

Hence, we have

$$\begin{aligned}
\lambda(f(v_i)) &\leq (t+3)^8 \cdot \prod_{j=3}^{t+3} \left(1 + \frac{6}{j}\right) \\
&= (t+3)^8 \cdot \frac{\prod_{j=3}^{t+3} (j+6)}{\prod_{j=3}^{t+3} j} \\
&\leq (t+3)^8 \cdot (t+9)^6 \leq (t+9)^{14}.
\end{aligned}$$

□

Remark 1 (Implementation remark)**.** *The proof of Lemma 30 is constructive and shows how to efficiently compute a good cut. Nevertheless, from the practical perspective, a natural heuristic to consider is to always compute the best cut. This is how the circular partitions in Picture 5-1 were computed.*

5.3 Partitions with slack

In this section, we show that if we allow small distortion of the volumes at each level of the tree, then there exists a partition of a hypercube into hyperrectangles (d-dimensional rectangles) of small aspect ratio. For each internal node, the hyperrectangles assigned to its children, may have volumes shrunken by a factor in the range $[1-\varepsilon, 1]$ with respect to the volume assigned to their parent.

In the algorithm, we always use cuts perpendicular to the longest side of a hyperrectangle. We try to balance the weights of the children assigned to each resulting hyperrectangle. If this is possible, the two resulting hyperrectangles also have small aspect ratios. Otherwise, one child must have large weight. Therefore, we can maintain small aspect ratios by slightly shrinking the volume of its hyperrectangle, and using the resulting empty space to improve the aspect ratio of the other, small hyperrectangle.

Definition 3 (Hierarchical Hyperrectangular Partition with Slack)**.** *Let $T = (V, E)$ be a rooted tree with n leaves, and depth d. Let $w : V \to \mathbb{R}_{\geq 0}$ be a function such* *that for any internal vertex $v \in V(T)$, with children $u_1, \ldots, u_k$, $w(v) \geq \sum_{i=1}^{k} w(u_i)$. Then a γ-*hierarchical hyperrectangular partition with ε-slack *for (T, w) is a mapping $f : V(T) \to \mathcal{P}(\mathbb{R}^d)$, for some $d \geq 2$, such that:*

- *For each $v \in V(T)$, $f(v)$ is a d-dimensional hyperrectangle with $\lambda_{\mathrm{rect}}(f(v)) \leq \gamma$.*
- *For the root r of T, $\mathrm{Vol}(f(r)) = w(r)$.*
- *For each $u, v \in V(T)$, such that u is the parent of v in T, $f(v) \subseteq f(u)$, and*

$$(1-\varepsilon)\frac{\mathrm{Vol}(f(u))}{w(u)} \leq \frac{\mathrm{Vol}(f(v))}{w(v)} \leq \frac{\mathrm{Vol}(f(u))}{w(u)}.$$

- *For each $u, v \in V(T)$, such that u is not an ancestor of v, and v is not an ancestor of u, $int(f(u)) \cap int(f(v)) = \emptyset$.*

Lemma 32. *Let $\varepsilon \in (0, 1/3)$, and let $d \geq 2$. Let $T = (V, E)$ be a rooted tree of depth t. Let $w : V \to \mathbb{R}_{\geq 0}$ be a function such that for any interval vertex $v \in V(T)$, with children $u_1, \ldots, u_k$, $w(v) \geq \sum_{i=1}^{k} w(u_i)$. Then, there exists a $1/\varepsilon$-hierarchical hyperrectangular partition $f : V \to \mathcal{P}(\mathbb{R}^d)$ for (T, w) with ε-slack.*

Proof. We create a mapping f such that for each $u \in V$, $f(u)$ is a hyperrectangle. We start from a hypercube of volume $w(r)$, where r is the root of the the tree. We fix $f(x)$ to be this hypercube. Its rectangular aspect ratio is 1.

We show by induction how to construct f and prove that the rectangular aspect ratio of each $f(u)$ is at most $1/\varepsilon$. This implies that the (standard) aspect ratio of each $f(u)$ is at most $\sqrt{d}/\varepsilon$.

For each $f(u)$, we define $w'_v = \frac{\mathrm{Vol}(f(u))}{w(u)} \cdot w(v)$ for each child v of u in T. Then we shrink $f(u)$ so that the volume of the shrunken hyperrectangle R is exactly equal to the sum of w'_v over the children v of u.

Whenever we want to subdivide a hyperrectangle R of rectangular aspect ratio at most $1/\varepsilon$ among a subset S of at least two children of u, we do what follows. We split S with a cut which is perpendicular to the longest side of R. Let $s \in S$ be the child in S of the largest w'_s. There are two cases.

- If $w'_s / \sum_{v \in S} w'_v \leq 1 - \varepsilon$, then we can split S into two sets S_1 and S_2 each of weight which is at most an $1-\varepsilon$ fraction of the total weight of S. Then we split R with a cut which is perpendicular to the longest cut, so that we create two hyperrectangles R_1 and R_2 of volume proportional to the total weight of S_1 and S_2, respectively. All sides but the longest are preserved in the new hyperrectangles, and the length of the initially longest side becomes an at least ε fraction of the original value. This implies that if the rectangular aspect ratio of R_1 or R_2 increases with respect to the ratio of R, then it cannot be greater than $1/\varepsilon$.

- The second case is when $w'_s / \sum_{v \in S} w'_v > 1 - \varepsilon$, i.e., there is a very heavy element in S. In this case, we must be more careful to avoid assigning a bad hyperrectangle. We first split R into two hyperrectangles R_1 and R_2 with a cut perpendicular to the longest side, so that $\mathrm{Vol}(R_1) = (1-\varepsilon)\,\mathrm{Vol}(R)$ and $\mathrm{Vol}(R_2) = \varepsilon\,\mathrm{Vol}(R)$. The rectangular aspect ratio of both R_1 and R_2 is at most $1/\varepsilon$. We set $f(s)$ to be R_1. This means that we assign to s a hyperrectangle of volume smaller by a factor of at most $1-\varepsilon$ than what is implied by the weight of s. To the other elements we assign R_2 uniformly shrunken so that its volume equals $\sum_{x \in S \setminus \{s\}} w'_x$. The shrunken R_2 is a subset of the initial R_2. We proceed with it recursively, until S has only one element.

□

5.4 Improved embeddings of ultrametrics into $\mathbb{R}^d$

In this section, we give an approximation algorithm for embedding ultrametrics into $\mathbb{R}^d$. Let $M = (X, D)$ be the given ultrametric. After scaling M, we can assume that the minimum distance is 1, and the diameter is Δ. It is known, and easy to see that for any $\alpha > 1$, M can be embedded into an α-HST, with distortion α (cf. [10]). Given M, we initially compute an embedding of M into a 2-HST T, with distortion 2. Let $M' = (X, D')$ be the metric space corresponding to T. Any embedding of M' into $\mathbb{R}^d$

with distortion c', is clearly also an embedding of M into $\mathbb{R}^d$ with distortion at most $c = O(c')$. It therefore suffices to embed of M' into $\mathbb{R}^d$.

The intuition behind our algorithm is as follows. We first compute a hierarchical partition of $\mathbb{R}^d$ into sets with small aspect ratio. The sets in the lower level of the partition would roughly correspond to balls around the images of the points in our embedding. Therefore, given the hierarchical partition we will be able to easily obtain the embedding.

More precisely, the algorithm works as follows. Initially, we compute the values $C(v)$, for each vertex v of the HST T. Then, using Lemma 32, we compute a $(\log \Delta)$-hierarchical hyperrectangular partition g for (T, C) (i.e. with weight assignment $w(v) = C(v)$). We further define a mapping $g' : V(T) \to \mathcal{P}(\mathbb{R}^d)$ by slightly modifying g as follows. Starting from the root of T, we traverse all the vertices of T. When we visit a vertex u, and we shrink uniformly all the hyperrectangles of the vertices in the subtree rooted at u, by a factor of $1 - 1/\log \Delta$, with the center of the hyperrectangle of u being the fixed point in the transformation. Let $g' : V(T) \to \mathcal{P}(\mathbb{R}^d)$ be the resulting mapping. Observe that for each $v \in V(T)$, $\mathrm{Vol}(g'(v)) \geq (1 - 1/\log \Delta)^{\log \Delta} \mathrm{Vol}(g(v)) = \Omega(\mathrm{Vol}(g(v))$, and that $\lambda_{\mathrm{rect}}(g(v')) = \lambda_{\mathrm{rect}}(g(v))$. For each point $x \in X$, let v_x be the leaf of T corresponding to x. Having computed g', we simply set $f(x)$ to be the center of the hyperrectangle $g'(v_x)$. It remains to bound the distortion of f.

Lemma 33. *The expansion of f is $O(\log \Delta \cdot c')$.*

Proof. Consider points $x, y \in X'$, and let v_x, v_y, be the leafs of T that correspond to x, and y respectively. Let v be the nearest common ancestor of v_x, and v_y, in T. We have $D'(x, y) = l(v)$. By Lemma 32, it follows that in the partition g' computed by the algorithm, v is mapped to a hyperrectangle $g(v') \subset \mathbb{R}^d$, with $\lambda_{\mathrm{rect}}(g'(v)) \leq \log \Delta$. Note that $f(x) \in g'(v_x)$, $f(y) \in g'(v_y)$, and also $g'(v_x) \subseteq g'(v)$, $g'(v_y) \subseteq g'(v)$. Since $\mathrm{Vol}(g'(v)) \leq \mathrm{Vol}(g(v)) \leq C(v)$, we have $\|f(x) - f(y)\|_2 \leq \mathrm{diam}(g'(v)) \leq \mathrm{diam}(g(v)) \leq d \cdot \log \Delta \cdot (C(v))^{1/d}$. Therefore, by Corollary 2, we obtain that $\|f(x) - f(y)\|_2 = O(c' \cdot l(v) \cdot \log \Delta) = O(\log \Delta \cdot c' \cdot D'(x, y))$. $\square$

Lemma 34. *The contraction of f is $O(\log^{O(1)} \Delta)$.*

Proof. Since the depth of T is $\log \Delta$, it follows that for each vertex $u \in V(T)$, $\mathrm{Vol}(g'(u)) = \Omega(\mathrm{Vol}(g(u))) = \Omega((1 - 1/\log \Delta)^{\log \Delta} C(u)) = \Omega(C(u))$. Consider points $x, y \in X'$, and let $v_x, v_y \in V(T)$ be the leafs of T corresponding to x, y respectively. Let v be the nearest common ancestor of v_x, and v_y in T. We will consider the following two cases for v:

Case 1: v is the parent of v_x, and v_y in T. Since the minimum distance in M' is 1, it follows that $D'(x, y) = 1$. By the construction, $f(x)$ is the center of $g'(v_x)$. Let t be the distance between $f(x)$, and $\partial g'(v_x)$. Since $\lambda_{\mathrm{rect}}(g'(v_x)) \leq \log \Delta$, we have

$$t \geq \frac{(\mathrm{Vol}(g'(v_x)))^{1/d}}{\log \Delta} = \frac{\Omega((C(v_x)^{1/d})}{\log \Delta} = \Omega(1/\log \Delta).$$

Thus, $\|f(x) - f(y)\|_2 \geq t = \Omega(D(x, y)/\log \Delta)$.

Case 2: v is not the parent of v_x, and v_y in T. Let u_x be the child of v, that lies on the path from v to v_x, in T. Let γ be the distance between x, and $\partial g'(u_x)$. By the construction of g' we have $\|f(x) - f(y)\|_2 \geq \gamma = \Omega((C(u_x))^{1/d}/\log^{O(1)} \Delta) = \Omega(l(u_x)/\log^{O(1)} \Delta) = \Omega(D(x, y)/\log^{O(1)} \Delta)$. □

Combining lemmas 34, and 33, we obtain the main result of the section.

Theorem 11. *For any fixed $d \geq 2$, there exists a polynomial-time, polylog(Δ)-approximation algorithm, for the problem of embedding ultrametrics into $\mathbb{R}^d$ with minimum distortion.*

Chapter 6

NP-hardness of embedding ultrametrics into $\mathbb{R}^2$

In this chapter we show that the problem of computing a minimum distortion embedding of an ultrametric into the plane under the ℓ_∞ norm is NP-hard.

The results presented in this chapter are from [19].

6.1 Preliminaries

We say that a square $S \subset \mathbb{R}^2$ is *orthogonal* if the sides of S are parallel to the axes. We perform a reduction from the following NP-complete problem (see [40]): Given a packing square S and a set of packed squares $L = \{s_1, \ldots, s_n\}$, is there an orthogonal packing of L into S? We call this problem SquarePacking.

For a square s, let $a(s)$ denote the length of its side. Assume w.l.o.g. for each $i \in [n]$, $a(s_i) \in \mathbb{N}$, $a(S) \in \mathbb{N}$, and that $a(s_1) \leq a(s_2) \leq \ldots \leq a(s_n)$. The SquarePacking problem is strongly NP-complete. Thus we can assume w.l.o.g. that there exists $N = \mathrm{poly}(n)$, such that $1 \leq a(s_1) \leq \ldots \leq a(s_n) \leq a(S) < N$.

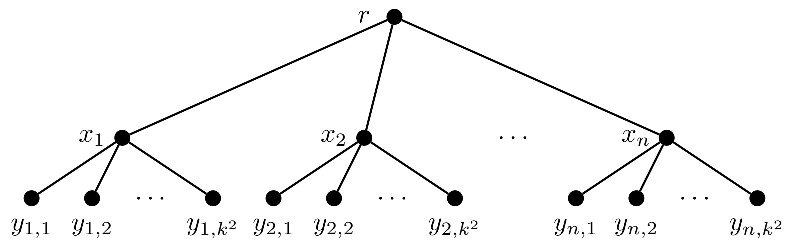


Figure 6-1: The constructed tree T. The labels of the vertices are: $l(r) = a(S)$ and $l(x_i) = a(s_i) - a(S)/(k-1)$.

6.2 The construction

Consider an instance of the SquarePacking problem, where S is the packing square, and $L = \{s_1, \dots s_n\}$ is the set of packed squares. We will define an ultrametric $M = (X, D)$ and an integer k, such that M embeds into the plane with distortion at most $k-1$ iff there exists an orthogonal packing of L into S. It is convenient to define M by constructing its associated labeled tree T, where each $v \in V(T)$ has a label $l(v) \in \mathbb{Q}$.

Let $k = N^{10}$. For each square $s_i \in L$, we introduce a set of k^2 leaves $y_{i,1}, \dots y_{i,k^2}$ in T. We connect all of these leaves to a vertex x_i, and we set $l(x_i) = a(s_i) - a(S)/(k-1)$. Note that $l(x_i)$ is very close to $a(s_i)$. Next, we introduce a root vertex $r \in V(T)$, and for each $i \in [n]$, we connect x_i to r. We set $l(r) = a(S)$.

For a vertex $v \in V(T)$, we denote by X_v the set of leaves of T having v as an ancestor. Figure 6-1 depicts the described construction.

6.2.1 Satisfiable instances

Assume that there exists an orthogonal packing of L into S. We will show that there exists an embedding $f : X \to \mathbb{R}^2$ with distortion $k-1$.

As a first step, for each vertex $x_i : 1 \le i \le n$, we embed all the vertices of X_{x_i} in a square Q_i of side $(k-1)l(x_i)$. This is done by simply placing a $k \times k$ orthogonal grid with step $l(x_i)$ inside Q_i and embedding the vertices of X_{x_i} on the grid points. Next, we transform the squares Q_i into squares Q'_i by adding empty strips of width $a(S)/2$

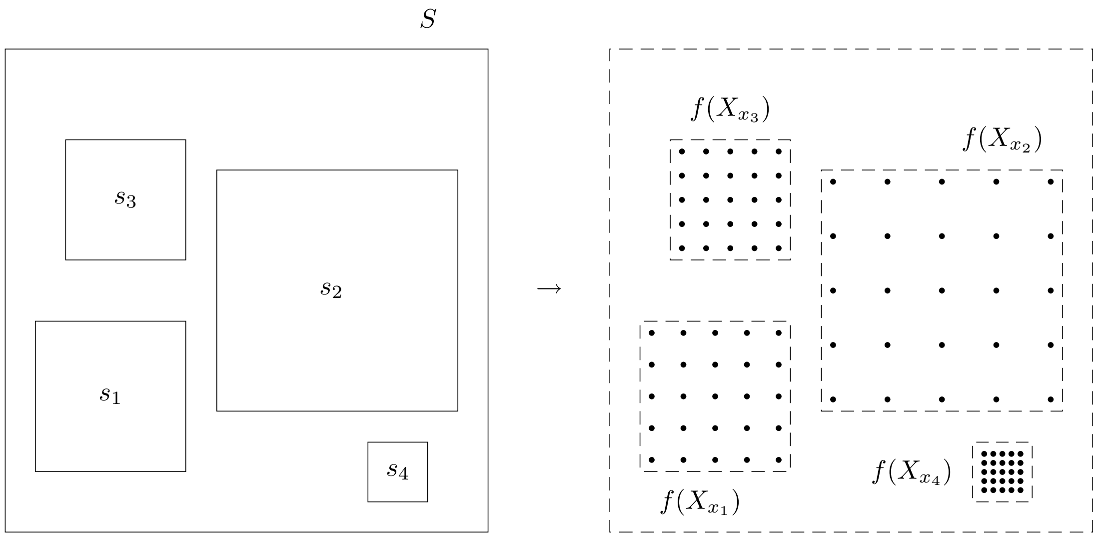


Figure 6-2: The embedding constructed for the YES instance.

around Q_i. Notice that the side of Q_i' is exactly $(k-1)l(x_i) + a(S) = (k-1)a(s_i)$. Finally, we embed the squares Q_i' into a square $\mathcal{S}$ of side $(k-1)a(S)$ according to the packing of the input squares in S. Figure 6-2 depicts the resulting embedding f.

We now show that the distortion of the embedding f is at most $k-1$.

Let $u, v \in X$. We have to consider the following cases for u, v:

Case 1: $u, v \in X_{x_i}$ for some $i \in [n]$. Since the vertices of X_{x_i} are embedded on a grid of step $l(x_i)$, it follows that $\|f(u) - f(v)\|_\infty \geq l(x_i) = D(u, v)$. Thus, the contraction is at most 1. Moreover, since all the vertices of X_{x_i} are embedded inside a square Q_i of side $l(x_i)(k-1)$, the expansion is at most $k-1$.

Case 2: $u \in X_{x_i}$ and $v \in X_{x_j}$, for some $i \neq j$. Since we add empty strips of width $a(S)/2$ around the squares Q_i, Q_j, we have that $\|f(u) - f(v)\|_\infty \geq a(S) = l(r) = D(u, v)$. Thus, the contraction is 1. On the other hand, all the vertices are embedded inside a square $\mathcal{S}$ of side $l(r)(k-1) = a(S)(k-1)$, and therefore the expansion is at most $k-1$.

Thus, we have shown that the distortion is at most $k-1$.

6.2.2 Unsatisfiable instances

Assume that there is no orthogonal packing of L inside S. We show that the minimum distortion required to embed M into the plane is greater than $k-1$. Assume that there exists an embedding $f : X \to \mathbb{R}^2$, with distortion at most $k-1$. W.l.o.g. we can assume that f is non-contracting.

The following lemma will be useful in the analysis.

Lemma 35. *Let $M = (X, D)$ be a uniform metric on k^2 points, for some integer $k > 0$. Then, the minimum distortion for embedding M into the plane is $k-1$. Moreover, an embedding f has distortion $k-1$ iff $f(X)$ is an orthogonal grid.*

Proof. By scaling M, we can assume w.l.o.g. that for any $u, v \in X$, $D(u, v) = 1$. Consider an non-contracting embedding $f : X \to \mathbb{R}^2$. For any $v \in X$, let A_v be square of side length 1, centered at $f(v)$. Clearly, for any $u, v \in X$, with $u \neq v$, the interiors of squares A_u and A_v are disjoint. Let $A = \bigcup_{v \in X} A_v$. It follows that $\mathrm{Vol}(A) = |X|$. Thus, there exist $p_1, p_2 \in A$, such that $\|p_1 - p_2\|_\infty \geq |X|^{1/2} = k$. Let $v_1, v_2 \in X$ be the centers of the squares A_{v_1}, A_{v_2} to which p_1 and p_2 belong, respectively. Then $\|f(v_1) - p_1\|_\infty \leq 1/2$, and $\|f(v_2) - p_2\|_\infty \leq 1/2$. It follows that $\|f(v_1) - f(v_2)\|_\infty \geq k - 1$. Thus the distortion is at least $k-1$.

Clearly, if f maps X onto a $k \times k$ orthogonal grid, the distortion of f is $k-1$. It remains to show that this is the only possible optimal embedding.

Assume that an embedding f has distortion $k-1$, and let f be non-contracting. Observe that since the diameter of $f(X)$ is at most $k-1$, $f(X)$ must be contained inside a square K of side length $k-1$. Let $\{A_v\}_{v \in X}$ be defined as above. It follows that A is contained inside a square K' of side length k. Since $\mathrm{Vol}(A) = \mathrm{Vol}(K')$, it easily follows that $f(X)$ is an orthogonal $k \times k$ grid. □

Corollary 3. *For each $i \in [n]$, $f(X_{x_i})$ is an orthogonal $k \times k$ grid of side length $(k-1)l(x_i) = (k-1)a(s_i) - a(S)$.*

For each $i \in [n]$, let Q'_i be the square of side length $(k-1)a(s_i)$, that has the same center of mass as $f(X_{x_i})$.

Claim 22. *For each $i, j \in [n]$, $i \neq j$, the interiors of the squares Q'_i, Q'_j are disjoint.*

Proof. Assume that the assertion is not true. That is, there exist $i, j \in [n]$, with $i \neq j$, and $p \in \mathbb{R}^2$, such that p belongs to the interiors of both squares Q'_i, Q'_j. By the definition of Q'_i and Q'_j, there are points $v_1 \in X_{x_i}$, $v_2 \in X_{x_j}$ which are embedded within distance smaller than $a(S)/2$ from p. But then $\|f(v_1) - f(v_2)\|_\infty < a(S)$, contradicting the fact that the embedding is non-contracting. $\square$

Claim 23. *$\bigcup_{i=1}^n Q'_i$ is contained inside a square of side length $ka(S)$.*

Proof. Since f has expansion at most $k - 1$, $f(X)$ is contained inside an orthogonal square $\mathcal{S}$ of side length $(k-1)l(r) = (k-1)a(S)$. Observe that for each $i \in [n]$, for each point $p \in Q_i$, there exists $v \in X_{x_i}$, such that $\|p - f(v)\|_\infty \leq a(S)/2$. Let $\mathcal{S}'$ be the square of side length $ka(S)$ that has the same center as $\mathcal{S}$. It follows that $\mathcal{S}'$ contains $\bigcup_{i=1}^n Q'_i$. $\square$

Lemma 36. *If M can be embedded into the plane with distortion at most $k-1$, then there exists an orthogonal packing of L inside S.*

Proof. If there exists an embedding $f : X \to \mathbb{R}^2$ with distortion $k - 1$, by Claim 23 we obtain that $\bigcup_{i=1}^n Q_i$ is contained inside a square of side length $ka(S)$. Moreover, by Claim 22, the embeddings of squares Q'_i defines a feasible packing of these squares into the square $\mathcal{S}'$. Note that for each $i : 1 \leq i \leq n$, Q_i has side length $(k-1)a(s_i)$. That is, the squares $Q_1, \ldots, Q_n$ are just scaled copies of the squares $s_1, \ldots, s_n$. Thus, we obtain that there exists an orthogonal packing of L inside a square S' of side length $a(S)\frac{k}{k-1}$. Recall that $k = N^{10} > a(S)^{10}$. Thus, S' has side length less than $a(S) + 1/2$.

Since $a(S)$ and $a(s_i)$ for each $i \in [n]$ are integers, it follows that there is also an orthogonal packing of L into a square of side length $a(S)$. $\square$

The following theorem is now immediate.

Theorem 12. *The problem of minimum-distortion embedding of ultrametrics into the plane under the ℓ_∞ norm is NP-hard.*

Chapter 7

Inapproximability of embedding into $\mathbb{R}^d$

It has been shown in [43] that for any $d \geq 1$, any n-point metric can be embedded into $\mathbb{R}^d$ with distortion $\tilde{O}(n^{2/d})$ via a random projection, and that in the worst case this bound is essentially optimal. This clearly also implies an $\tilde{O}(n^{2/d})$-approximation algorithm for minimizing the distortion. We show that for any fixed $d \geq 2$, there is no polynomial-time algorithm for embedding into $\mathbb{R}^d$, with approximation ratio better than $\Omega(n^{1/(17d)})$, unless P = NP. Our result establishes that random projection is not too far, concerning the dependence on d, from the best possible approximation algorithm for this problem. Note that since for fixed d all norms on $\mathbb{R}^d$ are equivalent up to a constant factor, the same result holds for all norms.

We obtain our hardness result via a reduction from the problem 3-SAT. We encode a SAT formula using geometric gadgets, that are subsets of d-dimensional grids. The main technical difficulty is to characterize the structural properties of these gadgets, under any low-distortion embedding into $\mathbb{R}^d$. Our approach for obtaining such characterizations is as follows. We first construct d-dimensional simplicial complexes that can be viewed as continuous analogs of our discrete gadgets. Intuitively, a low-distortion embedding of a discrete object, corresponds to a continuous mapping of a simplicial complex, satisfying certain non-intersection conditions. This correspondence allows us to translate desired geometric properties, to purely topological counterparts.

In the heart of our topological analysis lies the following lemma. Consider a unit ball B^d in $\mathbb{R}^d$, under the ℓ_2 norm. Assume that there exists a continuous mapping $f : B^d \to \mathbb{R}^d$, such that the image of the origin $f(0)$ lies in the unbounded connected component of $\mathbb{R}^d \setminus f(\partial B^d)$. That is, the origin moves "outside" the boundary of the ball. Then, there exist two points in B^d that are far apart from each other, and have the same image under f.

The above statement is derived via a careful application of Sarkaria's Coloring-Embedding theorem [54, 55, 44], The formulation of Sarkaria's theorem that we are using is due to Matoušek [44]. It gives sufficient conditions for the embeddability of a simplicial complex in terms of the chromatic number of a certain Kneser graph.

The results presented in this chapter are from [46].

7.1 A topological prelude

Before we describe our hardness reduction, we prove the main topological lemma (lemma 37), that we will use later in our analysis.

The *system of minimal nonfaces* of a simplicial complex T is the set of all minimal subsets of vertices of T, that are not contained in the same simplex in T. The *Kneser graph* of a family of sets $\mathcal{F}$, denoted by $KG(\mathcal{F})$, is the graph with vertex set $\mathcal{F}$, and edge set $\{\{s,t\} \in \binom{\mathcal{F}}{2} : s \cap t = \emptyset\}$. Finally, for a graph J, let $\chi(J)$ be its chromatic number.

The following theorem, which is due to Sarkaria [54, 55], gives a necessary condition for the existence of a continuous mapping from a simplicial complex into $\mathbb{R}^d$, in which the images of each pair of disjoint simplexes are disjoint. A detailed exposition of this theorem can be found in [44].

Theorem 13 (Sarkaria's Coloring/Embedding Theorem, [54, 55])**.** *Let T be a simplicial complex on n vertices, and let $\mathcal{F}$ be the system of minimal nonfaces of T. Then, if $d \leq n - \chi(KG(\mathcal{F})) - 2$, then for any continuous mapping $f : |T| \to \mathbb{R}^d$, the images of some two disjoint faces of T intersect.*

We define a simplicial complex K as follows. Let B^d be the ℓ_2 unit ball in $\mathbb{R}^d$. Let X be the boundary of the d-simplex, mapped on ∂B^d. Erecting a cone over X, with apex $a = 0$, results in a triangulation of B^d. Add a new vertex b, and an edge between a and b. Let K be the resulting simplicial complex. For example, for $d = 1$, we get a star with 3 leaves, and for $d = 2$, we get a disk with an edge attached to its center.

Let ξ_d be the minimum distance between any pair of points in B^d, that are mapped to disjoint simplices in K.

Lemma 37 (Main Topological Lemma)**.** *Let $d \geq 2$, and let B^d be the unit ball in $\mathbb{R}^d$. Let $f : B^d \to \mathbb{R}^d$ be a continuous map, such that $f(0)$ is in the closure of the unbounded connected component of $\mathbb{R}^d \setminus f(\partial B^d)$. Then, there exist $x, x' \in B^d$, with $\|x - x'\|_2 \geq \xi_d$, and $f(x) = f(x')$.*

Proof. Let K be the simplicial complex defined as above, and let $\mathcal{F}$ be the set of minimal nonfaces of K. Let V be the set of vertices of the original d-simplex X. Observe that $\mathcal{F}$ consists of all the sets $\{b, c\}$, $c \in V$, and $V \cup \{b\}$, $V \cup \{a\}$. Therefore, any two sets in $\mathcal{F}$ have non-empty intersection, and $KG(\mathcal{F})$ does not contain any edges. It follows that $\chi(KG(\mathcal{F}) = 1$. By theorem 13, we have that for any continuous mapping $G : |K| \to \mathbb{R}^d$, the images of two disjoint faces of K intersect.

Let $f : B^d \to \mathbb{R}^d$ be a continuous mapping, such that $f(0)$ is in the closure of the unbounded connected component of $\mathbb{R}^d \setminus f(\partial B^d)$. If $f(0) \in f(\partial B^d)$, then there is clearly a point $x \in B^d$, with $\|x\|_2 = 1$, such that $f(x) = f(0)$, and the assertion follows. Thus, we can assume that $f(0)$ is in the interior of the unbounded connected component of $\mathbb{R}^d \setminus f(\partial B^d)$. Fix a path P connecting $f(a)$ to a point outside $f(B^d)$ and avoiding $f(S^{d-1})$. Extend f to a mapping $\bar{f} : |K| \to R^d$ by mapping the part of K corresponding to B^d by f, and the edge between a and b to P. We know that there exist simplices $s_1, s_2 \in K$, with $s_1 \cap s_2 = \emptyset$, such that $\bar{f}(s_1) \cap \bar{f}(s_2) \neq \emptyset$. Observe that s_1 cannot be the edge ab, since then s_2 would have to be in the boundary of B^d, but $\bar{f}(ab)$ avoids $\bar{f}(S^{d-1})$ by construction. We also cannot have $s_1 = \{a\}$ since $\bar{f}(a)$ does not intersect $\bar{f}(b) \cup \bar{f}(S^{d-1})$. Thus, $s_1, s_2 \in B^d$. So we have that

$f(s_1) \cap f(s_2) \neq \emptyset$. Therefore, there exist points $p_1 \in s_1$, $p_2 \in s_2$, with $\|p_1 - p_2\|_2 \geq \xi_d$, and $f(p_1) = f(p_2)$. $\square$

7.2 The reduction

In this section we describe our NP-hardness reduction. We will reduce the problem 3-SAT(5) to our problem. Recall that an instance of 3-SAT(5) is a CNF formula $\phi = C_1 \wedge \ldots \wedge C_M$, on N variables $\chi_1, \ldots, \chi_N$, with each variable appearing in at most five clauses. Given a formula ϕ, we will construct a weighted undirected graph $G = (V, E)$. The shortest-path metric of G will be the instance of the problem of embedding into $\mathbb{R}^d$.

7.2.1 An informal description

Before we give the technical details, we discuss the high-level idea of the reduction. The graph G contains a main part H that we call the *wall*. The wall is a very large d-grid, with edges of length $1/\Gamma$, where Γ is a sufficiently large parameter, to be specified later. The purpose of the wall is to enforce some kind of structure in any low-distortion embedding of the rest of the graph. In particular, we chose the edges of H to be sufficiently small so that in any low-distortion embedding, the image of H induces a fine net on $\mathbb{R}^d$. We formalize this intuition in section 7.4, where we prove that c-embeddings of d-grids into $\mathbb{R}^d$, induce $O(c)$-nets in $\mathbb{R}^d$. At the same time, the edges of H are sufficiently large, so that given a satisfiable instance, we can construct a low-distortion embedding of G, by interleaving H with the rest of the gadgets. We also chose certain regions of the wall as *literal-gadgets*, encoding the literals in the 3-SAT formula.

The remaining parts of G are gadgets that encode the variables, and the clauses of the 3-SAT formula. A *variable-gadget*, encoding a variable, is a path with edges of length ε, where ε is a small parameter (much smaller than $1/\Gamma$), to be specified later. We connect a variable-gadget B_i to the wall H by adding edges between B_i and two paths of H. The two paths of H are sufficiently far from each other. Using the fact that the image of H induces a net in $\mathbb{R}^d$, we can show that the image of B_i under any low-distortion embedding has to be close to the image of one of the two paths to which it is attached. This is done by a careful argument that relates the image of B_i with that of the literal-gadgets. This way we encode the two possible true/false values that the i-th variable can attain in a satisfying assignment.

A *clause-gadget*, encoding a clause, is the boundary of d-grid, with edges of length ε. We similarly attach each clause gadget to parts of H that are isomorphic to boundaries of d-cubes, and correspond to the literals appearing in the clause. We can again show that the image of the clause-gadget under any low-distortion embedding, has to be close to the image of exactly one of the literal-gadgets that it is attached to. This way we encode the fact that in a satisfying assignment, each clause has to be satisfied by some literal.

Having established that in any low-distortion embedding, the images of variable-gadgets and the clause-gadgets are close to the images of certain parts of the image of H, it remains to show that given such an embedding, we can obtain a satisfying assignment for the 3-SAT formula. To that extend, we need to show that the images of a clause-gadget and a variable-gadget cannot be both near the same literal-gadget. This is done by showing that in such a scenario, one end-point of the variable-gadget is "inside" the clause-gadget, while the other one is "outside". This, implies that the images of these two gadgets have to "intersect".

However, since the gadgets are discrete objects, we can only state a continuous analog of the above intersection argument. This is done by extending the embedding linearly to a continuous map of appropriate d-dimensional simplicial complexes. We define a complex $\tilde{X}$ for each gadget X, so that X can be viewed as a discretization of $\tilde{X}$. After obtaining the above continuous formulation, we can apply topological techniques to prove the desired intersections, concluding the analysis.

7.2.2 The gadgets

We now proceed with the formal description of the graph G. We split the construction into certain parts of G, that encode different parts of the 3-SAT formula. Throughout our analysis we use the parameters $\Gamma = 6400 \cdot d^4 M^4/\xi_d$, $\varepsilon = 1/\Gamma^3$, and $L = 200 \cdot d \cdot \Gamma^4 \cdot M$.

The wall We start with a graph H with vertex set

$$A = \{x_i : i \in \{-L, -L+1, ..., L-1, L\}^d\},$$

interconnected as a d-dimensional cubic grid; that is, $\{x_i, x_j\}$ forms an edge if $\|i - j\|_1 = 1$. All the edges in H have length $1/\Gamma^2$. We will refer to H as the *wall*.

For $i, j \in \mathbb{Z}^d$, we denote by $A[i \ldots j]$ the rectangular part of A between x_i and x_j, and by $A'[i \ldots j]$ its boundary. Formally, we have

$$A[i \ldots j] = \{x_k \in A : i_1 \leq k_1 \leq j_1, \ldots, i_d \leq k_d \leq j_d\},$$

and

$$A'[i \ldots j] = A[i \ldots j] \setminus A[i + \mathbf{1}_d \ldots j - \mathbf{1}_d].$$

Literal-gadgets For every literal we define a region of the wall called *literal-gadget* and defined as follows: For a variable χ_i, we have literal-gadgets $\Lambda_{i,0}$ and $\Lambda_{i,1}$ for the literals $\neg\chi_i$ and χ_i respectively. Each literal-gadget corresponds to a rectangular region of the wall. More precisely, we set

$$\Lambda_{i,j} = H\left[A\left[\Gamma^2\lambda_{i,j} - 2\Gamma^{3/2}\mathbf{1}_d \ldots \Gamma^2\lambda'_{i,j} + 2\Gamma^{3/2}\mathbf{1}_d\right]\right],$$

where

$$\lambda_{i,j} = (0, (4i + 2j - 4), 0, 0, \ldots, 0),$$

and

$$\lambda'_{i,j} = (10, (4i + 2j - 3), 1, 1, \ldots, 1).$$

We also define the *frontier* of a literal-gadget, denoted by $\Phi_{i,j}$ to be the boundary of slightly larger region of H, containing the literal-gadget. Formally, we set

$$\Phi_{i,j} = H\left[A'\left[\Gamma^2\lambda_{i,j} - \frac{\Gamma^2}{2}\mathbf{1}_d \ldots \Gamma^2\lambda'_{i,j} + \frac{\Gamma^2}{2}\mathbf{1}_d\right]\right].$$

For each occurrence of a literal in a clause we define a rectangular *sub-literal-gadget* to be a region of a literal gadget. We chose these regions such that all sub-literal-gadgets are disjoint, and sufficiently far from each other. That is, for each $l \in [5]$, the sub-literal-gadget $\Lambda_{i,j,l}$ is

$$\Lambda_{i,j,l} = H\left[A\left[\Gamma^2\lambda_{i,j,l} - 2\Gamma^{3/2}\mathbf{1}_d \ldots \Gamma^2\lambda'_{i,j,l} + 2\Gamma^{3/2}\mathbf{1}_d\right]\right],$$

where $\lambda_{i,j,l} = (2l-1, (4i+2j-4), 0, 0, \ldots, 0)$, and $\lambda'_{i,j,l} = (2l, (4i+2j-3), 1, 1, \ldots, 1)$. Figure 7-1(a) depicts the placement of the literal-gadgets, the sub-literal-gadgets, and the frontiers in the wall.

Variable-gadgets For each variable χ_i, we introduce a graph called *variable-gadget*, denoted by B_i. B_i is a path $b_{i,0}, b_{i,1}, \ldots, b_{i,9/\varepsilon}$, with each edge having length ε. We connect a variable-gadget with the wall, by adding edges between to two paths in H. These two paths lie in the middle of the two literal-gadgets for the variable χ_i. Formally, for each $j \in \{0,1\}$, for each $b_{i,l} \in V(B_i)$, we add an edge of length $1/\Gamma$ between $b_{i,l}$ and x_w, where $w = ((\lfloor l \cdot \varepsilon \rfloor + 1/2)\Gamma^2, (4i-4+2j+1/2)\Gamma^2, \Gamma^2/2, \ldots, \Gamma^2/2)$. Figure 7-1(b) depicts how a variable-gadget is connected to the wall.

Clause-gadgets For each clause C_i, we introduce a graph called *clause-gadget*, denoted by K_i. Each clause-gadget is the boundary of a d-dimensional grid, and has

$$V(K_i) = \{\kappa_{i,j} : j \in \{0, \ldots, 1/\varepsilon\}^{d-1}, \|j - \mathbf{1}_d \cdot 1/(2\varepsilon)\|_\infty = 1/(2\varepsilon)\}.$$

We have an edge of length ε between each pair $\{\kappa_{i,l}, \kappa_{i,l'}\}$, with $\|l - l'\|_1 = 1$. We connect each clause-gadget K_i with the wall by adding edges between K_i and the boundaries of three sub-grids of the wall. Each such sub-grid is contained in the sub-literal-gadget corresponding to a literal appearing in the clause C_i. Formally, let C_i be the r-th clause in which the variable χ_t appears. Assume further that χ_t appears as the literal $y_{\cdot\cdot}$. We add an edge of length $1/\Gamma$ between each vertex $\kappa_{i,w}$ of K_i and the vertex $x_{\Gamma^2 \cdot \lfloor w \cdot \varepsilon \rfloor + w'}$ of the wall, where $w' = ((2r-1)\Gamma^2, (4t-4)\Gamma^2, 0, \ldots, 0)$ if $y_{i,j} = \chi_t$, and $w' = ((2r-1)\Gamma^2, (4t-3)\Gamma^2, 0, \ldots, 0)$ if $y_{i,j} = \neg\chi_t$. Observe that multiple vertices of K_i get attached to the same vertex of H. Figure 7-1(c) depicts how a clause-gadget is connected to the wall. This concludes the construction.

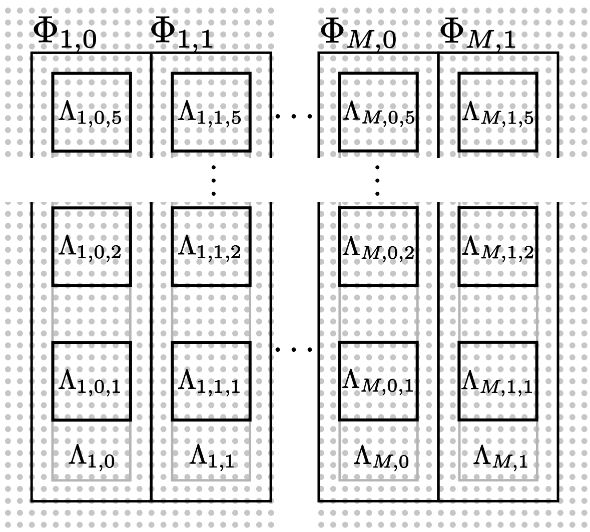
7.3 Satisfiable instances

We now show that if the formula ϕ is satisfiable, then G embeds into $\mathbb{R}^d$ with small distortion.

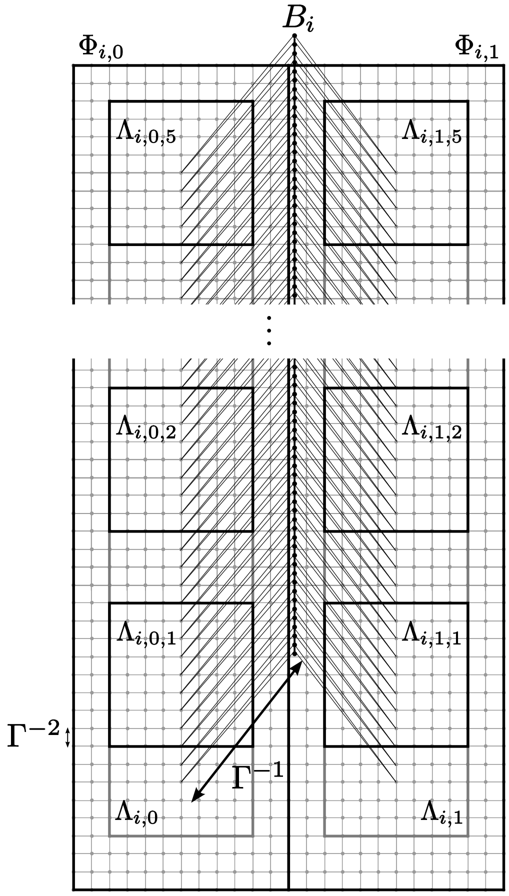
Lemma 38. *If ϕ is satisfiable, then G embeds into $\mathbb{R}^d$ with distortion at most $4\sqrt{d}M\Gamma$.*

Proof. Assume that ϕ is satisfiable, and fix a satisfying truth assignment T. We will define an embedding $f : V(G) \to \mathbb{R}^d$. We first define f on the vertices of the wall H. A natural embedding of the wall into $\mathbb{R}^d$ would map each vertex x_i to the point $(i_1 \cdot \Gamma^{-2}, \ldots, i_d \cdot \Gamma^{-2})$. This natural embedding is almost an isometry (ignoring possible short-cuts through the gadgets that might incur only an extra factor of $O(N)$). However, it is not appropriate here because it does not leave enough space for the remaining gadgets. We resolve this problem with the following modification: Along each dimension, after placing $\Gamma^2/2$ vertices of H, we leave a gap of length Γ^{-1}. Since the edges of the wall have length Γ^2, this can be done with distortion roughly $O(\Gamma)$. Recall that the gadgets are connected to the wall with edges of length Γ^{-1}. Therefore, we can place the gadgets inside these gaps, without contracting the distances between the gadgets and the wall by too much. Formally, for a vertex x_i of the wall, we set $f(x_i) = (g(i_1), g(i_2), \ldots, g(i_d))$, where $g(j) = \left\lceil \frac{2j}{\Gamma^2} \right\rceil \cdot \frac{1}{\Gamma} + j \cdot \frac{1}{\Gamma^2}$.

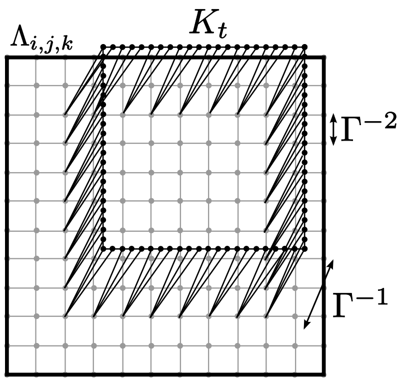
We next define f on the vertices of the variable-gadgets. Note that each gadget is connected to different parts of the wall, that are within $O(1)$ distance from each



(a) The part of the wall containing the literal-gadgets.



(b) Connecting a variable-gadget B_j with the literal-gadgets $\Lambda_{i,0}$, and $\Lambda_{i,1}$.



(c) Connecting a clause-gadget K_t with a sub-literal-gadget $\Lambda_{i,j,k}$.

Figure 7-1: The reduction for dimension $d = 2$.

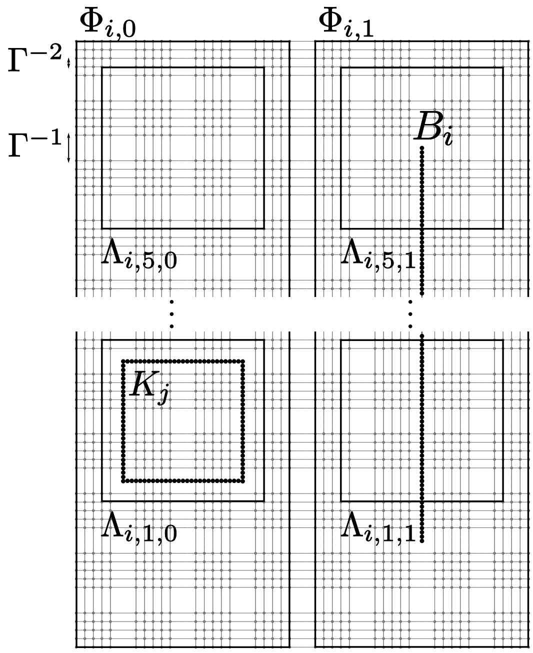


Figure 7-2: Example embedding for a satisfiable instance, for $d = 2$. In the satisfying assignment the variable χ_i is set to true, and the clause C_j is satisfied via the positive literal χ_i.

other. Since the distance between the gadgets and the wall is Γ^{-1}, by placing a gadget near a certain literal, we do not expand the distance to the remaining literals by too much. Formally, for each $i \in [N]$, let $\tau_i = 1$ if the variable χ_i is set to true in T, and $\tau_i = 0$ otherwise. For each $i \in [N]$, and for each $b_l \in V(B_i)$, we set $f(b_l) = \left(1 + \frac{2}{\Gamma}\right) \cdot \left(\frac{1}{2} + l \cdot \varepsilon, 4i - 4 + 2\tau_i, \frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right) + \frac{1}{2\Gamma} \cdot \mathbf{1}_d$. Finally, we define f on the vertices of the clause-gadgets. For each $i \in [M]$, let $v = v(i)$ be such that the clause C_i contains a literal y of the variable χ_v, and $T(y) = \text{true}$. Let K_i be the r-th clause in which the variable χ_v appears, for some $r \in [5]$. For each $\kappa_l \in V(K_i)$, we set $f(\kappa_l) = \left(1 + \frac{2}{\Gamma}\right) \cdot \left(\frac{1}{2} + 2r - 2, 4v - 4 + 2(1 - \tau_v), \frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right) + \frac{1}{2\Gamma} \cdot \mathbf{1}_d + \left(1 + \frac{2}{\Gamma}\right) \cdot \left(l \cdot \varepsilon - \frac{1}{2} \cdot \mathbf{1}_d\right)$. The resulting embedding is depicted in figure 7-2. It is straight-forward to verify that f has expansion at most $2M\Gamma$, and contraction at most $2\sqrt{d}$. $\square$

7.4 A structural property of embeddings of d-grids into $\mathbb{R}^d$.

In order to analyze the reduction for the case of unsatisfiable instances, we need to gain some understanding of the structure of low-distortion embeddings of d-grids into $\mathbb{R}^d$. To that extend, we show that in any embedding of sufficiently small distortion of a d-grid into $\mathbb{R}^d$, the image of the grid induces a net on a large ball around the image of the center of the grid. This basic property will be later used in our analysis to show that in any low-distortion embedding, the image of the wall induces a net in $\mathbb{R}^d$.

Since after adding the gadgets in G, the shortest-path metric on the wall is not anymore isometric to that of a d-grid, we need to prove the property for a slightly more general class of graphs, that we call *central contractions of grids*. Intuitively, a central contraction is obtained from a grid, by adding an arbitrary set of edges between vertices that are close to the center of the grid.

Definition 4 (Central Contraction of a Grid)**.** *Let J be a d-dimensional grid, with $V(J) = \{v_i : \forall j \in [d], 0 \leq i_j \leq k_j\}$, and $E(J) = \{\{v_i, v_{i'}\} : \|i - i'\|_1 = 1\}$, with each edge having unit length. Let J' be a graph obtained from J by adding a finite set of edges $\{v_i, v_j\}$, each having an arbitrary positive length, and such that for each $t \in [d]$, $i_t, j_t \in \{k_t(\frac{1}{2} - \frac{\xi_d}{8d^2}), \ldots, k_t(\frac{1}{2} + \frac{\xi_d}{8d^2})\}$. Then, J' is called a* central contraction *of J.*

We remark that the proof of the following statement (lemma 39) is the first place where we need to apply the topological property given by lemma 37. This might come as a surprise since at a first glance, the two statements seem unrelated. Informally, the argument is as follows. Consider a low distortion non-contracting embedding f of a central contraction of a grid, in which there is a large empty ball close to the image of the centroid of the grid. Let B_1 be the largest such ball. We can find a vertex u^* which is close to the centroid, and its image lies on the boundary of B_1. Since u^* is close to the centroid, there is a sufficiently large sub-grid Q centered at u^*. We extend f linearly to a continuous mapping $\tilde{g}$ of an appropriate d-dimensional simplicial complex $\tilde{Q}$ for which Q is a net. The complex $\tilde{Q}$ is chosen to be homeomorphic to a solid d-cube, which is in turn homeomorphic to the unit ball. Since the expansion is small, the image of each simplex of $\tilde{Q}$ is small, relative to the radius of B_1. Therefore, we can slightly modify the mapping $\tilde{g}$, so that the image of the complex avoids the interior of B_1. By applying a suitable homeomorphism on a subset of $\mathbb{R}^d$, we obtain a continuous map of $\tilde{Q}$ into $\mathbb{R}^d$, such that $\tilde{g}(u^*)$, lies on the boundary the unbounded connected component of $\mathbb{R}^d \setminus \tilde{g}(\partial\tilde{Q})$. Since $\tilde{Q}$ is homeomorphic to the unit ball, we can apply lemma 37, to obtain two points in $\tilde{Q}$ that are far from each other, and have the same image under $\tilde{g}$. Since Q is a net on $\tilde{Q}$, we can find vertices that are far from each other in Q, and their images are very close under f, contradicting the non-contraction hypothesis.

Lemma 39 (From Grids to Nets). *Let $d \geq 2$, and let $J = (V, E)$ be a d-dimensional grid with $V(J) = \{v_i : \forall j \in [d], 0 \leq i_j \leq k_j\}$, and $E(J) = \{\{v_i, v_{i'}\} : \|i - i'\|_1 = 1\}$, such that for each $j \in [d]$, $k_j \geq R$, for some $R \geq c \cdot \frac{128 \cdot d}{\xi_d}$. Assume that each edge of J has unit length. Let J' be a central contraction of J. Let $f : V(J') \to \mathbb{R}^d$ be a non-contracting embedding of J' into $\mathbb{R}^d$ with expansion c. Then, for any $p \in \mathbb{R}^d$, with $\|p - f(v_{k_1/2,\ldots,k_d/2})\|_2 \leq R/16$, there exists $u \in V(J')$, with $\|p - f(u)\|_2 \leq 2 \cdot c$.*

Proof. Let f be a non-contracting embedding of J' into $\mathbb{R}^d$ with expansion c. Assume for the sake of contradiction that there exists a point $p \in \mathbb{R}^d$, with $\|p - f(v_{k_1/2,\ldots,k_d/2})\|_2 \leq R/16$, such that for any $u \in V(J)$, $\|p - f(u)\|_2 > 2 \cdot c$.

Let u^* be the vertex of J which is nearest to p under f. That is, $u^* = \mathrm{argmin}_{u \in V(J)} \|p - f(u^*)\|_2$. Since f is non-contracting, we have

$$\begin{aligned}
D_J(u^*, v_{k_1/2,\ldots,k_d/2}) &\leq \|f(u^*) - f(v_{k_1/2,\ldots,k_d/2})\|_2 \\
&\leq \|f(u^*) - f(p)\|_2 + \|f(p) - f(v_{k_1/2,\ldots,k_d/2})\|_2 \\
&\leq 2 \cdot \|f(p) - f(v_{k_1/2,\ldots,k_d/2})\|_2 \\
&\leq R/8
\end{aligned}$$

Therefore, there exist $i^* \in \mathbb{Z}^d$, such that for each $j \in [d]$, $i^*_j \in \{k_j/8, \ldots, 7k_j/8\}$, with

$u^* = v_{i^*}$.

Let $A = \{v_{i_1,\ldots,i_d} \in V(J) : \|i - i^*\|_\infty \leq 3R/8\}$, and define the vertex-induced subgraph $Q = J[A]$. We construct a d-dimensional simplicial complex $\tilde{Q}$ corresponding to the graph Q as follows. The set of 0-simplices of $\tilde{Q}$ is A. For each $i \in \mathbb{Z}^d$, such that for each $j \in [d]$, $i_j \in \{i^*_j - 3R/8, i^*_j + 3R/8 - 1\}$, let T_i be the triangulation of the hypercube $A_i = Q[\{v_j : j \in \{i_1, i_1+1\} \times \ldots \times \{i_d, i_d+1\}\}]$. We add to $\tilde{Q}$ all the simplices in T_i.

Let g be the restriction of f on $V(Q)$. Recall that for a simplicial complex K, $|K|$ denotes the union of all its simplices. Let $\tilde{g}$ be the linear extension of g on $|\tilde{Q}|$.

Let $B_1 = B(p, \|p - f(u^*)\|_2)$. Note that $f(V(J)) \cap \mathrm{Int}(B_1) = \emptyset$. We proceed to define a map $\tilde{h} : |\tilde{Q}| \to \mathbb{R}^d \setminus \mathrm{Int}(B_1)$. For each point $x \in |\tilde{Q}|$ with $\tilde{g}(x) \notin \mathrm{Int}(B_1)$, we set $\tilde{h}(x) = \tilde{g}(x)$. For each point $x \in |\tilde{Q}|$ with $\tilde{g}(x) \in \mathrm{Int}(B_1)$, let r_x be the ray starting at p and passing through $\tilde{g}(x)$. We set $\tilde{h}(x)$ to be the point where r_x intersects ∂B_1.

Define $\phi : |\tilde{Q}| \to \mathbb{R}^d$ where for each 0-simplex $v_{i_1,\ldots,i_d} \in \tilde{Q}$, $\phi(v_i) = (i_1 - i^*_1, \ldots, i_d - i^*_d)$, and for all other points $x \in \tilde{Q}$, $\phi(x)$ is defined via a linear extension.

Let C denote the unit ball in $\mathbb{R}^d$ under the ℓ_2 norm. We define a map $\mu : C \to |\tilde{Q}|$ as follows. Let $\mu(0) = u^*$, and for each $x \in \mathbb{R}^d \setminus \{0\}$, let $\mu(x) = \phi^{-1}(\frac{3R}{8} \cdot x \cdot \frac{\|x\|_2}{\|x\|_\infty})$.

Consider the map $\psi : C \to \mathbb{R}^d \setminus \mathrm{Int}(B_1)$ defined by $\psi(x) = \tilde{h}(\mu(x))$. The map ψ is clearly continuous. Furthermore, $\psi(C)$ is homeomorphic to a subset of the unit ball in $\mathbb{R}^d$, under a homeomorphism that sends 0 to the boundary of the unit ball. We can thus apply lemma 37 and obtain points $y, y' \in C$, such that $\|y - y'\|_2 \geq \xi_d$, and $\psi(y) = \psi(y')$. Let $\mu(y) = x$, and $\mu(y') = x'$, for some $x, x' \in |\tilde{Q}|$. Let σ, σ' be simplices of $\tilde{Q}$ such that $x \in \sigma$, $x' \in \sigma'$. Pick vertices $w, w' \in \tilde{Q}$, with $w \in \sigma$, $w' \in \sigma'$.

Observe that for each $p \in |\tilde{Q}|$, $\|\tilde{h}(p) - \tilde{g}(p)\|_2 < c$. Thus,

$$\begin{aligned}
\|f(w) - f(w')\|_2 &= \|\tilde{h}(w) - \tilde{h}(w')\|_2 \\
&\leq \|\tilde{h}(w) - \tilde{h}(x)\|_2 + \|\tilde{h}(x') - \tilde{h}(w')\|_2 \\
&< \|\tilde{g}(w) - \tilde{g}(x)\|_2 + \|\tilde{g}(x') - \tilde{g}(w')\|_2 + 4 \cdot c \\
&\leq 6 \cdot c \qquad (7.1)
\end{aligned}$$

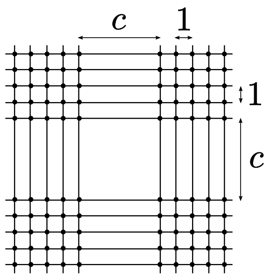


Figure 7-3: A tight example for lemma 39.

Since J' is a central-contraction of J, it follows that for each $u, v \in V(J')$, $D_{J'}(u,v) \geq D_J(u,v) - R \cdot \frac{\xi_d}{4d}$. Thus,

$$
\begin{aligned}
D_{J'}(w,w') &\geq \|\phi(w) - \phi(w')\|_1 - R \cdot \frac{\xi_d}{4d} \\
&\geq \|x - x'\|_1 - \|\phi(w) - x\|_1 - \|\phi(w') - x'\|_2 - R \cdot \frac{\xi_d}{4d} \\
&\geq \frac{1}{d} \cdot \frac{3R}{8} \cdot \|y - y'\|_2 - 2 \cdot d - R \cdot \frac{\xi_d}{4d} \\
&\geq \frac{1}{d} \cdot \frac{3R}{8} \cdot \xi_d - 2 \cdot d - R\frac{\xi_d}{4d} \\
&> \frac{\xi_d}{8d} \cdot R - 2 \cdot c \\
&\geq 14 \cdot c
\end{aligned}
\tag{7.2}
$$

Combining (7.1) and (7.2) we obtain a contradiction of the fact that f is non-contracting. $\square$

Remark 2. *For any fixed $d \geq 2$, the bound on $\|p - f(u)\|_2$ given in lemma 39 is tight up to a constant factor. Figure 7-3 depicts an embedding of a 2-dimensional grid into $\mathbb{R}^2$ with distortion $O(c)$, such that $\|p - f(u)\|_2 = \Omega(c)$.*

7.5 Unsatisfiable instances

We will now show that if there exists an embedding of G into $\mathbb{R}^d$ with small enough distortion, then ϕ is satisfiable. Let $f : V(G) \to \mathbb{R}^d$ be an embedding with distortion at most $\Gamma^{3/2}$. After scaling f, we can assume w.l.o.g. that it is non-contracting and has expansion at most $\Gamma^{3/2}$.

For each clause-gadget K_i, we define a simplicial complex $\tilde{K}_i$ corresponding to K_i as follows. We set $V(K_i)$ to be the set of 0-simplices of $\tilde{K}_i$. We also add simplices in $\tilde{K}_i$ so that each hypercube in K_i corresponds to a subdivision of a solid hypercube in $\tilde{K}_i$.

Similarly, we define a simplicial complex $\tilde{H}$ corresponding to the wall H as follows. We add all the vertices of H as 0-simplices in $\tilde{H}$. We will add simplices in $\tilde{H}$, so that each hypercube of H corresponds to a subdivision of a solid hypercube in $\tilde{H}$. This way, each literal-gadget $\Lambda_{i,j}$, sub-literal-gadget $\Lambda_{i,j,k}$, and frontier $\Phi_{i,j}$ induces naturally a subcomplex $\tilde{\Lambda}_{i,j}$, $\tilde{\Lambda}_{i,j,k}$, and $\tilde{\Phi}_{i,j}$ of $\tilde{H}$ respectively. Recall that each $\Lambda_{i,j,k}$ is a grid of side-length $s_1 = \Gamma^2 + 4\Gamma^{3/2}$, and each K_l is a grid of side-length $s_2 = 1/\varepsilon$. We can assume w.l.o.g. that s_2 is a multiple of s_1. In this case, by adding extra vertices on each $\partial\tilde{\Lambda}_{i,j,k}$, we can pick a triangulation of $\tilde{H}$ such that each $\tilde{\Lambda}_{i,j,k}$ is combinatorially isomorphic to each $\tilde{K}_l$.

We extend f to a map $\tilde{f}$ defined on all of the above complexes, via linear extension over the simplices.

Lemma 40 shows via an application of lemma 37 that the image of each literal-gadget has to be "inside" the image of its frontier.

Lemma 40. *For each $i \in [N]$, for each $j \in \{0,1\}$, and for each $v \in V(\Lambda_{i,j})$, $f(v)$ is contained in the interior of a bounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{\Phi}_{i,j})$.*

Proof. Let X be the subcomplex of $\tilde{H}$ induced on the vertex set

$$A\left[\Gamma^2\lambda_{i,j} - \frac{\Gamma^2}{2}\mathbf{1}_d \dots \Gamma^2\lambda'_{i,j} + \frac{\Gamma^2}{2}\mathbf{1}_d\right].$$

Observe that $\partial X = \tilde{\Phi}_{i,j}$. Let Y be the unit ball in $\mathbb{R}^d$ under the ℓ_2 norm. It is easy to see that there exists a homeomorphism $\phi : X \to Y$ with $\phi(v) = 0$, and such that for each $x, x' \in X$, $\|\phi(x) - \phi(x)\|_2 \geq \|x - x'\|_2/(20 \cdot d)$.

Assume that the assertion is true. Then, by lemma 37 we obtain that there exist $z, z' \in X$, such that $\tilde{f}(z) = \tilde{f}(z')$, and $\|z - z'\|_2 \geq \xi_d/(20 \cdot d)$. Let w and w' be vertices from simplices of X that contain z and z' respectively. We have

$$\|w - w'\|_2 \geq \|z - z'\|_2 - \frac{2 \cdot \sqrt{d}}{\Gamma^2} \geq \frac{\xi_d}{20 \cdot d} - \frac{2 \cdot \sqrt{d}}{\Gamma^2} > \frac{\xi_d}{40 \cdot d}.$$

On the other hand,

$$\|f(w) - f(w')\|_2 \leq \|\tilde{f}(w) - \tilde{f}(z)\|_2 + \|\tilde{f}(w') - \tilde{f}(z')\|_2 \leq \Gamma^{3/2} \cdot \frac{2 \cdot d}{\Gamma^2} < \|w - w'\|_2,$$

contradicting the non-contraction of f. □

Lemma 41. *For each $i, i' \in [N]$, and for each $j, j' \in \{0, 1\}$, with either $i \neq i$, or $j \neq j'$, for each $v \in V(\Lambda_{i,j})$, $f(v)$ is contained in the closure of the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{\Phi}_{i',j'})$.*

Proof. Assume for the sake of contradiction that the assertion is not true. Pick $i, i' \in [N]$, and $j, j' \in \{0, 1\}$, with $(i, j) \neq (i', j')$, and $v \in \Lambda_{i,j}$, such that $f(v)$ is contained in the interior of a bounded connected component X of $\mathbb{R}^d \setminus \tilde{f}(\tilde{\Phi}_{i',j'})$.

Let u be a vertex in $\partial\tilde{H}$. Observe that there exists a path P in the 1-skeleton of $\tilde{H}$ between v and u, such that for any $p \in P$, and any $x \in X$, $\|p - x\|_2 \geq 1$. Note that the diameter of X is at most $\Gamma^{3/2} \cdot 8 \cdot d$, while the distance between X and $f(u)$ is at least $L/2$. Thus, $f(u)$ is not contained in X. It follows that $\tilde{f}(P) \cap \tilde{f}(\tilde{\Phi}_{i',j'}) \neq \emptyset$. Pick $z \in P$, $z' \in \tilde{\Phi}_{i',j'}$, such that $\tilde{f}(z) = \tilde{f}(z')$. Pick vertices w and w' from the simplices that contain z and z' respectively. Similarly to the proof of lemma 40, we obtain that $\|f(w) - f(w')\|_2 < \|w - w'\|_2$, contradicting the fact that f is non-contracting. □

Definition 5 (Variable Gadget Near a Literal Gadget)**.** *For $j \in [M]$, $l \in \{0, 1\}$, we say that* the variable gadget B_j is near the literal gadget $\Lambda_{j,l}$ *if for each $v \in V(B_i)$ there exists $u \in V(\Lambda_{i,j})$ such that $\|f(v) - f(u)\|_2 \leq 2/\Gamma^{1/2}$.*

Definition 6 (Clause Gadget Near a Literal Gadget)**.** *For $i \in [N]$, $j \in [M]$, $l \in \{0, 1\}$, we say that* the clause gadget K_i is near the literal gadget $\Lambda_{j,l}$ *if for each $v \in V(K_i)$ there exists $u \in V(\Lambda_{i,j})$ such that $\|f(v) - f(u)\|_2 \leq 2/\Gamma^{1/2}$.*

Lemma 42. *For each vertex v in either a variable-gadget, or a clause-gadget, there exists $u \in V(H)$, with $\|f(u) - f(v)\|_2 \leq 2/\Gamma^{1/2}$.*

Proof. Observe that the shortest-path metric of G restricted on the wall H, is the shortest-path metric of a central contraction of $G[V(H)]$. Note that for each vertex v in either a variable-gadget, or a clause-gadget,

$$\|f(x_{(0,\ldots,0)}) - f(v)\|_2 \leq \Gamma^{3/2} D_G(x_{0,\ldots,0}, v) < 10\Gamma^{3/2} d\Gamma^2 < L/16.$$

Thus, by lemma 39 if follows that there exists $u \in V(H)$, with $\|f(u) - f(v)\|_2 \leq 2\Gamma^{3/2}/\Gamma^2 = 2/\Gamma^{1/2}$. ☐

Lemma 43. *For each variable χ_i, $i \in [N]$, there exists unique $j \in \{0, 1\}$, such that the variable-gadget B_i is near the literal gadget $\Lambda_{i,j}$.*

Proof. Let $v \in V(B_i)$. By lemma 42 there exists $u \in V(H)$, with $\|f(u) - f(v)\|_2 \leq 2/\Gamma^{1/2}$. Since f is non-contracting, $D_G(u, v) \leq 2/\Gamma^{1/2}$. Let w be the neighbor of v in H which is closest to u. We have $D_G(w, u) \leq 2/\Gamma^{1/2}$. By the construction w is contained in a literal-gadget $\Lambda_{i,j}$, for some $j \in \{0, 1\}$. Moreover, the ball of radius $2/\Gamma^{1/2}$ around w in G, is contained in $\Lambda_{i,j}$. Thus, $u \in \Lambda_{i,j}$.

It remains to show that for any $v' \in V(B_i)$, $u' \in V(H)$, such that $\|f(u') - f(v')\|_2 \leq 2/\Gamma^{1/2}$, $u' \in V(\Lambda_{i,j})$. Assume for the sake of contradiction that $u' \in V(\Lambda_{i,j'})$, $j' \neq j$. It follows that there exists a path P in H from u' to $w' = x_{(L,\ldots,L)}$, such that $D_G(P, \Phi_{i,j}) \geq 1/2$. Let $\tilde{P}$ be the polygonal curve in $\mathbb{R}^d$ connecting the images under f of consecutive vertices in P. Since $D_G(w', \Phi_{i,j}) > L/3$, it follows that $f(w')$ is in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{\Phi}_{i,j})$. By lemma 40, $f(u)$ is contained in the closure of a bounded connected component X of $\mathbb{R}^d \setminus \tilde{f}(\tilde{\Phi}_{i,j})$. Therefore, if $f(u') \in X$, then $\tilde{P} \cap \tilde{f}(\tilde{\Phi}_{i,j}) \neq \emptyset$. This however implies that there exist $z \in V(\Phi_{i,j})$, and $z' \in V(P)$ such that $\|f(z) - f(z')\|_2 \leq 2/\Gamma^2$, and $D_G(z, z') > 1/2$, contradicting the non-contraction of f. We thus obtain that $f(u') \notin X$.

The distance between $f(u)$ and $\tilde{f}(\tilde{\Phi}_{i,j})$ is at least $1/(2d) > 2/\Gamma^{1/2}$. Thus, $f(v) \in X$. Similarly, it follows that $f(v') \notin X$. Thus, $\tilde{f}(B_i) \cap \tilde{f}(\tilde{\Phi}_{i,j}) \neq \emptyset$, contradicting the non-contraction of f. This concludes the proof. ☐

Lemma 44. *For each $i \in [N]$, there exists a unique variable χ_j appearing in C_i, $j \in [M]$, such that the clause-gadget K_i is near the literal-gadget $\Lambda_{j,l}$, where $l = 0$ if χ_i appears as a positive literal in C_i, and $l = 1$ otherwise.*

Proof. The proof is analogous to the proof of lemma 43. $\square$

The following lemma shows that main structural property satisfied by the variable-gadgets, and the clause-gadgets; they cannot be both near the same literal-gadget. This is essentially the most technically involved part of this section.

Lemma 45. *For each $i \in [N]$, $j \in [M]$, $l \in \{0, 1\}$, if the variable-gadget B_i is near the literal-gadget $\Lambda_{i,l}$, then the clause-gadget K_j is not near the literal-gadget $\Lambda_{i,l}$.*

Proof. Assume for the sake of contradiction that B_i and K_j are both near the literal-gadget $\Lambda_{i,l}$. Assume that C_j is the t-th clause in which the variable χ_i appears. By the construction of the complex $\tilde{H}$, there exists a simplicial map $\phi : \tilde{K}_j \to \tilde{\Lambda}_{i,l,t}$, such that $\phi(\tilde{K}_j) = \partial\tilde{\Lambda}_{i,l,t}$.

Since K_j is near $\Lambda_{i,l}$, it follows that for each $v \in V(K_j)$, there exists $r(v) \in V(\Lambda_{i,l,t})$, such that $\|f(v) - f(r(v))\|_2 \leq 2/\Gamma^{1/2}$. Pick a shortest path P_v between $\phi_2(v)$ and $\phi_1(r(v))$ in the 1-skeleton of $\tilde{\Lambda}_{i,l,t}$.

Let $\tilde{Y}$ be the subcomplex of $\tilde{\Lambda}_{i,l,t}$ obtained by contracting each P_v into $r(v)$, and removing any simplices that contain the same vertex at least twice. Let $s : \tilde{\Lambda}_{i,l,t} \to \tilde{Y}$ be the resulting map. Observe that s is simplicial, and that $s(\phi(\tilde{K}_j)) = \partial\tilde{Y}$. Note that by contracting an edge of a path P_v, with one end-point on the boundary, and removing the simplices with multiply occurrences of a vertex, the resulting space is homeomorphic to $\tilde{\Lambda}_{i,l,t}$. Since $\tilde{Y}$ is obtained after a finite number of such contractions, it follows that it is homeomorphic to $\tilde{\Lambda}_{i,l,t}$.

Let p be the centroid of $\Lambda_{i,l,t}$. That is, $p = x_{\Gamma^2(\lambda_{i,l,t}+\lambda'_{i,l,t})/2}$. Let $z = b_{i,(2t-1)/\varepsilon}$. Note that $\{p, z\} \in E(G)$, is an edge of length $1/\Gamma$. Since B_i is near $\Lambda_{i,j}$, it follows that there exists $w \in V(\Lambda_{i,l,t})$, such that $\|f(z) - f(w)\|_2 \leq 2/\Gamma^{1/2}$, and $D_G(w, p) \leq 2/\Gamma^{1/2}$.

We will first show that $f(z)$ is contained in a bounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$. Suppose that this is not true. We define a map $\tilde{g}(\tilde{Y}) \to \mathbb{R}^d$ as follows. For each 0-simplex $v \in \partial\tilde{Y}$, we set $\tilde{g}(v) = \tilde{f}(\phi^{-1}(s^{-1}(v)))$. For each 0-simplex $v \in \mathrm{Int}(\tilde{Y})$, we set $\tilde{g}(v) = \tilde{f}(s^{-1}(v))$. For each other point of $\tilde{Y}$, $\tilde{g}$ is defined via linear extension on each simplex. Let B^d denote the ℓ_2 unit ball in $\mathbb{R}^d$. Note that $\tilde{X}$ is homeomorphic to B^d, via a homeomorphism h, with $h(\phi_1(w)) = 0$, and such that for each $q, q' \in \tilde{\Lambda}_{i,j,t}$, $\|q - q'\| \geq \|h(\phi_1(q)) - h(\phi_1(q'))\|_2/(2d)$. Observe that if $f(w)$ is contained in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$, then $\tilde{g}(h(0))$ is contained in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{g}(h(B^d))$. Thus, by applying lemma 37 on the map $\tilde{g} \circ h$, we obtain that there exist $a, a' \in B^d$, with $\tilde{g}(h(a)) = \tilde{g}(h(a'))$, and $\|a - a'\|_2 \geq \xi_d$. Let $b, b' \in V(\Lambda_{i,j,l})$ be the nearest vertices to $h(a)$, $h(a')$ respectively in $\tilde{\Lambda}_{i,j,l}$. We have $\|\tilde{g}(b) - \tilde{g}(b')\|_2 \leq 2\Gamma^{3/2}/\Gamma^2 = 2/\Gamma^{1/2}$. Since f is non-contracting, it follows each P_v has length at most $2/\Gamma^{1/2}$. Therefore, $D_G(b, b') \geq \xi_d/d - 6/\Gamma^2$. Note that by the definition of $\tilde{g}$, we have that for any $v \in \tilde{Y}$, $\|\tilde{f}(v) - \tilde{g}(v)\|_2 \leq 2/\Gamma^{1/2}$. Thus, $\|\tilde{f}(b) - \tilde{f}(b')\|_2 \leq 6/\Gamma^{1/2} < D_G(b, b')$, contradicting the non-contraction of f. We thus obtain that $f(w)$ is contained in a bounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$. Since $\|f(w) - f(z)\|_2 \leq 2/\Gamma^2$, and the distance between $f(w)$ and $\tilde{f}(\partial\tilde{K}_j)$ is at least $1/3$, it follows that $f(z)$ is also in a bounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$.

Let, $p' = x_{\Gamma^2(3\lambda_{i,l,t} + \lambda'_{i,l,t})/2}$. Let $z' = b_{i,(2t-2)/\varepsilon}$. Note that $\{p', z'\} \in E(G)$, is an edge of length $1/\Gamma$. Since B_i is near $\Lambda_{i,j}$, it follows that there exists $w' \in V(\Lambda_{i,l,t})$, such that $\|f(z') - f(w')\|_2 \leq 2/\Gamma^{1/2}$, and $D_G(w', p') \leq 2/\Gamma^{1/2}$.

We now claim that $\tilde{f}(z')$ is in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$. To see that, let Q be a path between w and $x' = x_{(L,\ldots,L)}$ in G, such that $D_G(Q, K_j) > 1/4$. Observe that the distance between $\tilde{f}(x')$ and $\tilde{f}(\tilde{K}_j)$ is greater than the diameter of $\tilde{f}(\tilde{K}_j)$. Thus, $\tilde{f}(x')$ is contained in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$. By the non-contraction of f, it follows that the polygonal curve connecting consecutive vertices in Q, cannot intersect $\tilde{f}(\tilde{K}_j)$. Thus, $\tilde{f}(w')$ is also in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$. Also, since $\|\tilde{f}(w') - \tilde{f}(z')\|_2$ is less that the distance between $\tilde{f}(w')$ and $\tilde{f}(\tilde{K}_j)$, it follows that $\tilde{f}(z')$ is also in the unbounded connected component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$.

Since $\tilde{f}(z)$ is in the bounded component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$, and $\tilde{f}(z')$ is in the unbounded component of $\mathbb{R}^d \setminus \tilde{f}(\tilde{K}_j)$, it follows that there exist $c \in V(K_j)$, $c' \in V(B_i)$, such that $\|f(c) - f(c')\|_2 \leq 2\Gamma^{3/2}\varepsilon$. On the other hand, $D_G(c, c') \geq D_G(K_j, B_i) = 2/\Gamma > \|f(c) - f(c')\|_2$, contradicting the non-contraction of f. $\square$

Lemma 46. *If there exists an embedding of G into $\mathbb{R}^d$ with distortion at most $\Gamma^{3/2}$, then ϕ is satisfiable.*

Proof. We define a truth assignment T as follows. By lemma 43 we have that for each variable χ_i, there exists unique $j_i \in \{0, 1\}$ such that the variable-gadget B_i is near the literal-gadget $\Lambda_{i,j}$. We set $T(\chi_i) = \text{true}$ if $j_i = 1$, and $T(\chi_i) = \text{false}$ otherwise.

By lemma 44, we have that for each clause C_i, there exists unique $j \in [M]$, and unique $l \in \{0, 1\}$, such that the variable χ_j appears in C_i, K_i is near $\Lambda_{j,l}$, and $l = 0$ iff χ_j appears as a positive literal in C_i. Let y be the literal of χ_j in C_i. By lemma 45, B_j is not near $\Lambda_{j,l}$, and therefore $T(y) = \text{true}$. It follows that T satisfies ϕ. $\square$

Theorem 14. *For any fixed $d \geq 2$, the problem of computing a minimum distortion embedding of an n-point metric space into $\mathbb{R}^d$, is NP-hard to approximate within $\Omega(n^{1/(17d)})$.*

Proof. We have $n = |V(G)| = |V(H)| + \sum_{i \in [M]} |V(K_i)| + \sum_{j \in [N]} |V(B_j)| = O(L^d) + O(Md(1/\varepsilon)^{d-1}) + O(N/\varepsilon) = O(L^d) = O(M^{17d})$. By lemma 38, if ϕ is satisfiable, then G embeds into $\mathbb{R}^d$ with distortion at most $O(M\Gamma) = O(M^5)$. Also, by lemma 46, if ϕ is unsatisfiable, then any embedding of G into $\mathbb{R}^d$ has distortion at least $\Omega(\Gamma^{3/2}) = \Omega(M^6)$. Therefore, it is NP-hard to approximate the optimal distortion within a factor better than $\Omega(M) = \Omega(n^{1/(17d)})$ $\square$

Bibliography

[1] R. Agarwala, V. Bafna, M. Farach-Colton, B. Narayanan, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy: (fitting distances by tree metrics). *7th Symposium on Discrete Algorithms*, 1996.

[2] N. Ailon and M. Charikar. Fitting tree metrics: Hierarchical clustering and phylogeny. In *Proceedings of the Symposium on Foundations of Computer Science*, 2005.

[3] N. Alon. Problems and results in extremal combinatorics, II. *Discrete Math.*, 2008. To appear.

[4] N. Alon, M. Bădoiu, E. Demaine, M. Farach-Colton, M. T. Hajiaghayi, and A. Sidiropoulos. Ordinal embeddings of minimum relaxation: General properties, trees and ultrametrics. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, 2005.

[5] D. Avis and M. Deza. The cut cone, l_1 embedability, complexity and multicommodity flows. *Networks*, 21:595–617, 1991.

[6] R. Babilon, J. Matoušek, J. Maxová, and P. Valtr. Low-distortion embeddings of trees. *J. Graph Algorithms Appl.*, 7:399–409, 2003.

[7] M. Balzer and O. Deussen. Voronoi treemaps. In *INFOVIS*, page 7, 2005.

[8] M. Balzer, O. Deussen, and C. Lewerentz. Voronoi treemaps for the visualization of software metrics. In *SOFTVIS*, pages 165–172, 2005.

[9] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. *Proceedings of the Symposium on Foundations of Computer Science*, 1996.

[10] Y. Bartal and M. Mendel. Dimension reduction for ultrametrics. In *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 664–665, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics.

[11] M. Bateni, M. Hajiaghayi, , E. D. Demaine, and M. Moharrami. Plane embeddings of planar graph metrics. *Discr. Comput. Geometry*, 38:615–637, 2007.

[12] B. B. Bederson, B. Shneiderman, and M. Wattenberg. Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Trans. Graph.*, 21(4):833–854, 2002.

[13] T. Bladh, D. A. Carr, and M. Kljun. The effect of animated transitions on user navigation in 3D tree-maps. In *IV*, pages 297–305, 2005.

[14] T. Bladh, D. A. Carr, and J. Scholl. Extending tree-maps to three dimensions: A comparative study. In *APCHI*, pages 50–59, 2004.

[15] J. Bourgain. On Lipschitz embedding of finite metric spaces into Hilbert space. *Isreal Journal of Mathematics*, 52:46–52, 1985.

[16] D. M. Bruls, C. Huizing, and J. J. van Wijk. Squarified treemaps. In *W. de Leeuw, R. van Liere (eds.), Data Visualization 2000, Proceedings of the joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42. Springer, 2000.

[17] M. Bădoiu, , P. Indyk, and A. Sidiropoulos. Approximation algorithms for embedding general metrics into trees. In *Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[18] M. Bădoiu, J. Chuzhoy, P. Indyk, and A. Sidiropoulos. Low-distortion embeddings of general metrics into the line. In *Proc. 37th ACM Symposium on Theory of Computing*, 2005.

[19] M. Bădoiu, J. Chuzhoy, P. Indyk, and A. Sidiropoulos. Embedding ultrametrics into low-dimensional spaces. In *22nd Annual ACM Symposium on Computational Geometry*, 2006.

[20] M. Bădoiu, K. Dhamdhere, A. Gupta, Y. Rabinovich, H. Raecke, R. Ravi, and A. Sidiropoulos. Approximation algorithms for low-distortion embeddings into low-dimensional spaces. In *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms*, 2005.

[21] L. Cai and D. G. Corneil. Tree spanners. *SIAM J. Discrete Math*, 8(3):359–387, 1995.

[22] M. Cary, A. Rudra, and A. Sabharwal. On the hardness of embeddings between two finite metrics. In *ICALP*, pages 1412–1423, 2005.

[23] M. M. Deza and M. Laurent. *Geometry of Cuts and Metrics.* Algorithms and Combinatorics 15. Springer-Verlag, Berlin etc., 1997.

[24] J. Edmonds. Embedding into ℓ_∞^2 is easy embedding into ℓ_∞^3 is NP-complete. In *Proc. 18th ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[25] Y. Emek and D. Peleg. Approximating minimum max-stretch spanning trees on unweighted graphs. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[26] D. Eppstein. Spanning trees and spanners. *Handbook of Computational Geometry (Ed. J.-R. Sack and J. Urrutia)*, pages 425–461, 2000.

[27] M. Farach-Colton, S. Kannan, and T. Warnow. A robust model for finding optimal evolutionary tree. *Proceedings of the Symposium on Theory of Computing*, 1993.

[28] S. P. Fekete and J. Kremer. Tree spanners in planar graphs. *Discrete Applied Mathematics*, 108:85–103, 2001.

[29] A. Gupta. Embedding tree metrics into low dimensional Euclidean spaces. *Discrete Comput. Geom.*, 24:105–116, 2000.

[30] A. Gupta. Steiner nodes in trees don't (really) help. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, 2001.

[31] A. Hall and C. H. Papadimitriou. Approximating the distortion. In *APPROX-RANDOM*, 2005.

[32] B. Harb, S. Kannan, and A. McGregor. Approximating the best-fit tree under l_p norms. In *Proceedings of Approx*, 2005.

[33] L. Jin and D. C. Banks. Tennisviewer: A browser for competition trees. *IEEE Computer Graphics and Applications*, 17(4):63–65, 1997.

[34] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Proc., IEEE Conference on Visualization*, pages 284–291, 1991.

[35] W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mapping into Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

[36] W.-A. Jungmeister and D. Turo. Adapting treemaps to stock portfolio visualization. Technical Report UMCP-CSD CS-TR-2996, University of Maryland, College Park, Maryland 20742, USA, 1992.

[37] A. Kearsley, R. Tapia, and M. Trosset. The solution of the metric STRESS and SSTRESS problems in multidimensional scaling using Newton's method. *Computational Statistics*, 1998.

[38] C. Kenyon, Y. Rabani, and A. Sinclair. Low distortion maps between point sets. *Proceedings of the Symposium on Theory of Computing*, 2004.

[39] J. R. Lee, A. Naor, and Y. Peres. Trees and Markov convexity. In *Proc. 17th ACM-SIAM Symposium on Discrete Algorithms*, 2006.

[40] J. Y.-T. Leung, T. W. Tam, C. S. Wong, G. H. Young, and F. Y. L. Chin. Packing squares into a square. *J. Parallel Distrib. Comput.*, 10(3):271–275, 1990.

[41] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some its algorithmic applications. *Combinatorica*, 15:215–245, 1995.

[42] S. Malitz and J. Malitz. A bounded compactness theorem for l_1-embeddings of metric spaces in the plane. *In Discrete Comput. Geom.*, 8:373–385, 1992.

[43] J. Matoušek. Bi-Lipschitz embeddings into low-dimensional Euclidean spaces. *Comment. Math. Univ. Carolinae*, 31:589–600, 1990.

[44] J. Matoušek. *Using the Borsuk-Ulam Theorem.* Springer, 2003.

[45] J. Matoušek and A. Sidiropoulos. Inapproximability for metric embeddings into $\mathbb{R}^d$. *Manuscript*, 2008.

[46] J. Matoušek and A. Sidiropoulos. On the computational near-optimality of random projection. *Manuscript*, 2008.

[47] MDS: Working Group on Algorithms for Multidimensional Scaling. Algorithms for multidimensional scaling. DIMACS Web Page.

[48] K. Onak and A. Sidiropoulos. Circular partitions with applications to visualization and embeddings. In *Proceedings of the 24th ACM Symposium on Computational Geometry, to appear*, 2008.

[49] C. Papadimitriou and S. Safra. The complexity of low-distortion embeddings between point sets. *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pages 112–118, 2005.

[50] D. Peleg and E. Reshef. A variant of the arrow distributed directory protocol with low average case complexity. In *Proc. 25th Int. Colloq. on Automata, Language and Programming*, pages 670–681, 1998.

[51] D. Peleg and D. Tendler. Low stretch spanning trees for planar graphs. *Technical Report MCS01-14, The Weizmann Institute of Science*, 2001.

[52] D. Peleg and J. D. Ullman. An optimal synchronizer for the hypercube. In *Proc. 6th ACM Symposium on Principles of Distributed Computing*, pages 77–85, 1987.

[53] J. Rekimoto and M. Green. The information cube: Using transparency in 3d information visualization. In *Proceedings of the Third Annual Workshop on Information Technologies and Systems (WITS'93)*, pages 125–132, 1993.

[54] K. S. Sarkaria. A generalized Kneser conjecture. *J. Combinatorial Theory, Set. B*, 49:236–240, 1990.

[55] K. S. Sarkaria. A generalized van Kampen-Flores theorem. *Proc. Amer. Math. Soc.*, 111:559–565, 1991.

[56] B. Shneiderman. Treemaps for space-constrained visualization of hierarchies. Available at `http://www.cs.umd.edu/hcil/treemap-history/index.shtml`.

[57] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.

[58] B. Shneiderman and M. Wattenberg. Ordered treemap layouts. In *IEEE Symposium on Information Visualization, INFOVIS '01*, pages 73–78, 2001.

[59] J. Stasko, R. Catrambone, M. Guzdial, and K. McDonald. An evaluation of space-filling information visualizations for depicting hierarchical structures. *International Journal of Human-Computer Studies*, 53(5):663–694, 2000.

[60] D. Turo and B. Johnson. Improving the visualization of hierarchies with treemaps: Design issues and experimentation. *IEEE Visualization*, pages 124–131, 1992.

[61] J. J. van Wijk and H. van de Wetering. Cushion treemaps: Visualization of hierarchical information. In *INFOVIS*, pages 73–78, 1999.

[62] G. Venkatesan, U. Rotics, M. S. Madanlal, J. A. Makowsky, and C. P. Rangan. Restrictions of minimum spanner problems. *Information and Computation*, 136(2):143–164, 1997.

[63] R. Vliegen, J. J. van Wijk, and E.-J. van der Linden. Visualizing business data with generalized treemaps. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):789–796, 2006.

[64] S. Wan. Blog treemap visualizer (`http://www.samuelwan.com/information/archives/000159.html`).

[65] M. Weskamp. Newsmap. http://marumushi.com/apps/newsmap/newsmap.cfm.

[66] K. Wetzel. Using Circular Treemaps to visualize disk usage. Available at `http://lip.sourceforge.net/ctreemap.html`.