Fat polygonal partitions with applications to visualization and embeddings

FAT POLYGONAL PARTITIONS WITH APPLICATIONS TO VISUALIZATION AND EMBEDDINGS[*]

Mark de Berg,[†] *Krzysztof Onak,*[‡] *and Anastasios Sidiropoulos*[§]

ABSTRACT. Let $\mathcal{T}$ be a rooted and weighted tree, where the weight of any node is equal to the sum of the weights of its children. The popular Treemap algorithm visualizes such a tree as a hierarchical partition of a square into rectangles, where the area of the rectangle corresponding to any node in $\mathcal{T}$ is equal to the weight of that node. The aspect ratio of the rectangles in such a rectangular partition necessarily depends on the weights and can become arbitrarily high.

We introduce a new hierarchical partition scheme, called a *polygonal partition*, which uses convex polygons rather than just rectangles. We present two methods for constructing polygonal partitions, both having guarantees on the worst-case aspect ratio of the constructed polygons; in particular, both methods guarantee a bound on the aspect ratio that is independent of the weights of the nodes.

We also consider *rectangular partitions with slack*, where the areas of the rectangles may differ slightly from the weights of the corresponding nodes. We show that this makes it possible to obtain partitions with constant aspect ratio. This result generalizes to hyper-rectangular partitions in $\mathbb{R}^d$. We use these partitions with slack for embedding ultrametrics into d-dimensional Euclidean space: we give a polylog(Δ)-approximation algorithm for embedding n-point ultrametrics into $\mathbb{R}^d$ with minimum distortion, where Δ denotes the spread of the metric, i.e., the ratio between the largest and the smallest distance between two points. The previously best-known approximation ratio for this problem was polynomial in n. This is the first algorithm for embedding a non-trivial family of weighted-graph metrics into a space of constant dimension that achieves polylogarithmic approximation ratio.

1 Introduction

Hierarchical structures are commonplace in many areas. It is not surprising, therefore, that the visualization of hierarchical structures—in other words, of rooted trees—is one of the most widely studied problems in information visualization and graph drawing. In the weighted variant of the problem, we are given a rooted tree in which each leaf has a positive

[†]*TU Eindhoven*, `mdberg@win.tue.nl`

[‡]*IBM T.J. Watson Research Center*, `krzysztof@onak.pl`

[§]*Dept. of Computer Science & Engineering, and Dept. of Mathematics, The Ohio State University*, `sidiropoulos.1@osu.edu`

weight and the weight of each internal node is the sum of the weights of the leaves in its subtree. One of the most successful practical algorithms for visualizing such weighted trees is the so-called Treemap algorithm. Treemap visualizes the given tree by constructing a hierarchical *rectangular partition* of a square, as illustrated in Figure 1(a). More precisely, Treemap assigns a rectangle to each node in the tree such that

- the area of the rectangle is equal to the weight of the node;

- the rectangles of the children of each internal node ν form a partition of the rectangle of ν.

The Treemap algorithm was proposed by Shneiderman [21] and its first efficient implementation was given by Johnson and Shneiderman [14]. Treemap has been used to visualize a wide range of hierarchical data, including stock portfolios [15], news items [26], blogs [25], business data [24], tennis matches [13], photo collections [6], and file-system usage [21, 27]. Shneiderman maintains a webpage [20] that describes the history of his invention and gives an overview of applications and proposed extensions to his original idea. Below we only discuss the results that are directly related to our work.

In general, there are many different rectangular partitions corresponding to a given tree. To obtain an effective visualization it is desirable that the aspect ratio of the rectangles be kept as small as possible; this way the individual rectangles are easier to distinguish and the areas of the rectangles are easier to estimate. Various heuristics have been proposed for minimizing the aspect ratio of the rectangles in the partition [8, 22, 23]. Unfortunately, the aspect ratio can become arbitrarily bad if the weights have unfavorable values. For example, consider a tree with a root and two leaves, where the first leaf has weight 1 and the second has weight W. Then the optimal aspect ratio of any rectangular partition is unbounded as $W \to \infty$. Hence, in order to obtain guarantees on the aspect ratio we cannot restrict ourselves to rectangles. This lead Balzer et al. [2, 1] to introduce Voronoi treemaps, which use more general regions in the partition. However, their approach is heuristic and it does not come with any guarantees on the aspect ratio of the produced regions. Thus the following natural question is still open:

> Suppose we are allowed to use arbitrary convex polygons in the partition, rather than just rectangles. Is it then always possible to obtain a partition that achieves aspect ratio independent of the weights of the nodes in the input tree? (The aspect ratio of a convex region A is defined as $\mathrm{diam}(A)^2 / \mathrm{area}(A)$, where $\mathrm{diam}(A)$ is its diameter and $\mathrm{area}(A)$ is its area.[1])

Our results on hierarchical partitions. Our main result is an affirmative answer to the question above: we present two algorithms that, given an n-node tree of height h, construct

---

[1]Another common definition of the aspect ratio of a convex region A is the ratio between the radius R of the smallest circumscribing circle and the radius r of the largest inscribed circle. The aspect ratio defined in this manner is sometimes referred to as the *fatness* of the region. There are several other definitions of fatness, all of which are equivalent up to constant factors for convex planar objects [11]. In particular, in our case it is easy to show that $\mathrm{diam}(A) = \Theta(R)$ and $\mathrm{area}(A) = \Theta(R \cdot r)$, which implies that $\mathrm{diam}(A)^2 / \mathrm{area}(A) = \Theta(R/r)$.

(a) A rectangular partition   (b) A greedy polygonal partition

(c) An angular polygonal partition
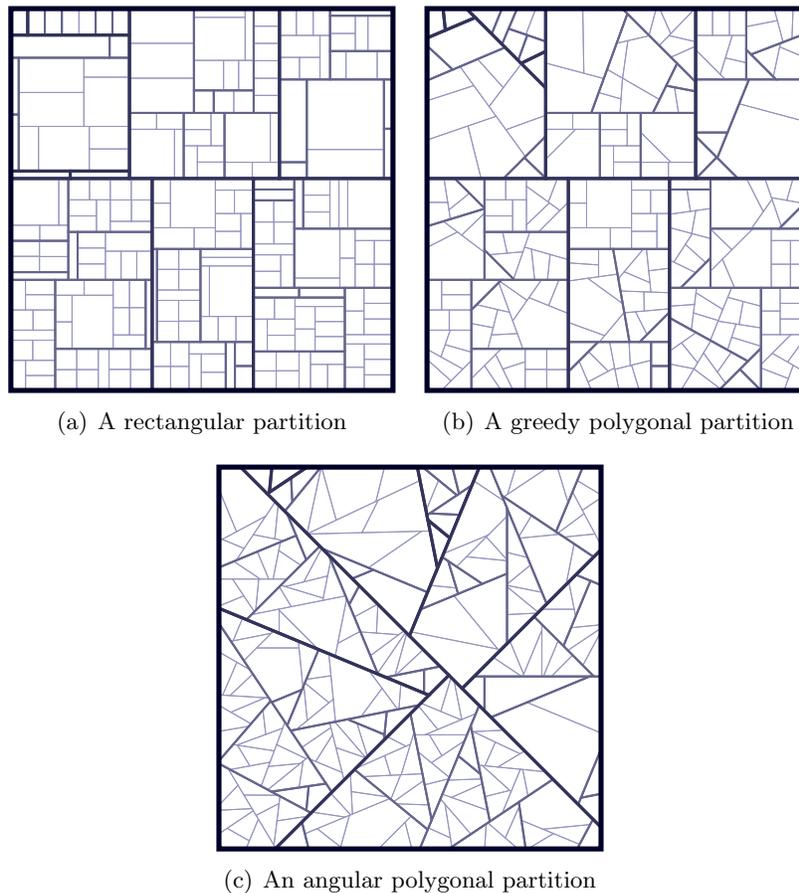
Figure 1: Sample polygonal partitions

a partition into convex polygons with aspect ratio $O(\mathrm{poly}(h, \log n))$. Our algorithms, which are described in Section 2, are very simple. They first convert the input tree into a binary tree, and then recursively partition the initial square region using straight-line cuts. The methods differ in the way in which the orientation of the cutting line is chosen at each step. The *greedy method* minimizes the maximum aspect ratio of two subpolygons resulting from the cut, while the *angular method* maximizes the angle that the splitting line makes with any of the edges of the polygon being cut. Figures 1(b) and 1(c) depict partitions computed by our algorithms.

The main challenge lies in the analysis of the aspect ratio achieved by the algorithms, which is given in Sections 3 and 4. We prove that the angular method produces a partition with aspect ratio $O(h + \log n)$. For the greedy method we can only prove an aspect ratio of $O((h + \log n)^8)$. Since the greedy method is the most natural one, we believe this result is still interesting. Moreover, in the (limited) experiments we have done—see Section 2—the greedy method always outperforms the angular method. Besides these two algorithms, we also prove a lower bound: we show in Section 5 that for certain trees and weights, any

partition into convex polygons must have polygons with aspect ratio $\Omega(h)$.[2]

After having studied the problem of constructing polygonal partitions, we return to rectangular partitions. As observed, it is in general not possible to obtain any guarantees on the aspect ratio of the rectangles in the partition. If, however, we are willing to let the area of a rectangle deviate slightly from the weight of its corresponding node then we can obtain bounded aspect ratio, as we show in Section 6. More precisely, we obtain the following partition. Let $\varepsilon \in (0, 1/3)$. We allow that the area A of the rectangle assigned to every non-root node v is shrunken by a factor of at most $1 - \varepsilon$ compared to its share of the area A' of the rectangle of the parent v'. That is, we only require that

$$(1 - \varepsilon) \cdot \frac{A'}{w_{v'}} \leq \frac{A}{w_v} \leq \frac{A'}{w_{v'}},$$

where w_v and $w_{v'}$ are the weights of v and v', respectively. Then we show that the aspect ratio of every rectangle can be bounded by $1/\varepsilon$. We call this kind of partition a *rectangular partition with slack*.

Application to embedding ultrametrics. The work of Bǎdoiu *et al.* [9] establishes a lower bound for the distortion of the best embedding of an ultrametric into $\mathbb{R}^d$. Our hierarchical partitions with low aspect ratio can be used for efficiently constructing an embedding that closely matches the lower bound of Bǎdoiu *et al.* More details, including a brief history of relevant embedding results, follow.

Let us first recall a few standard definitions. A metric space $M = (X, D)$ is a set X together with a symmetric distance function $D \colon X \times X \to \mathbb{R}_{\geq 0}$ that satisfies the triangle inequality and $D(x_1, x_2) = 0$ if and only if $x_1 = x_2$. An *embedding* of a metric space $M = (X, D)$ into a *host* metric space $M' = (X', D')$ is an injective mapping $f \colon X \to X'$. The *distortion* of an embedding f is defined as

$$\max_{x,y \in X} \frac{D'(f(x), f(y))}{D(x, y)} \cdot \max_{x,y \in X} \frac{D(x, y)}{D'(f(x), f(y))}.$$

Over the past few decades, low-distortion embeddings of metric spaces into various host spaces have been the subject of extensive study [12]. Embeddings into Euclidean space are of particular importance in applications, and have received a lot of attention. Bourgain's theorem [7] asserts that any n-point metric space admits an embedding into high-dimensional Euclidean space with distortion $O(\log n)$. Matoušek [16] has shown that the minimum distortion for embedding into d-dimensional Euclidean space is $n^{\Theta(1/d)} \cdot \log^{O(1)} n$. Since the distortion for embedding into constant-dimensional Euclidean space can be polynomially large in the worst case, it is natural to ask whether we can *approximate* the best possible distortion for a given input metric. Matoušek and Sidiropoulos [17] have shown that minimum-distortion embeddings of general metrics into $\mathbb{R}^d$ (with $d \geq 2$) are hard to approximate to within a factor of roughly $n^{1/(22d-10)}$, unless P=NP [17]. In other words, it is unlikely that there exists a polynomial-time algorithm with significantly better performance than the worst case guarantee.

---

[2]Recently de Berg, Speckmann, and van der Weele [10] refined our angular method to get rid of the additive $O(\log n)$ factor in the upper bound, thus obtaining a polygonal partition of aspect ratio $O(h)$.

In light of the above inapproximability result, it is natural to ask whether there exist interesting families of metrics, for which we can obtain better than polynomial approximation factors for embedding into constant-dimensional Euclidean space. In this paper we present the first result of this type, for embedding *ultrametrics* into $\mathbb{R}^d$. An ultrametric is a metric satisfying the following strengthened version of the triangle inequality: for any $x, y, z \in X$ we have $D(x, z) \leq \max\{D(x, y), D(y, z)\}$. Equivalently, $M = (X, D)$ is an ultrametric if it can be realized as the shortest-path metric over the leaves of a rooted edge-weighted tree such that the distance between the root and any leaf is the same. Ultrametrics have received a lot of attention in the embeddings literature, and play a central role in many algorithmic applications (see e.g. [3]).

Bădoiu *et al.* [9] showed that finding a minimum-distortion embedding of an ultrametric into $\mathbb{R}^2$ is NP-complete and presented an $O(n^{1/3})$-approximation algorithm for the problem. They extended the algorithm to embedding ultrametrics into $\mathbb{R}^d$, obtaining an $(n^{\frac{1}{d} - \Theta(\frac{1}{d^2})})$-approximation. This result is obtained using a lower bound on the amount of space required in a non-contracting embedding of every subtree of an ultrametric into $\mathbb{R}^d$. We apply our results on rectangular partitions with slack to a hierarchical structure corresponding to the ultrametric with weights given by the lower bound. Bounded aspect ratios in our partition imply relatively low distortion and good approximation to the best embedding of the ultrametric into $\mathbb{R}^d$. Moreover, we show a connection between embedding ultrametrics into $\mathbb{R}^d$ and (hyper-)rectangular partitions with slack and we use this connection to obtain a significant improvement over the result of Bădoiu *et al.* [9]. More precisely, using our results on (hyper-)rectangular partitions with slack, we obtain a polynomial-time polylog(Δ)-approximation algorithm for the problem of embedding ultrametrics into $(\mathbb{R}^d, \ell_2)$ with minimum distortion, where Δ is the *spread* of X. (The spread of X is defined as $\Delta = \mathrm{diam}(X)/\min_{x,y \in X} D(x, y)$.) As long as the spread is sub-exponential in n, this is an exponential improvement over [9].

2   The two algorithms

Before we present our algorithms, we define the problem more formally and introduce some notation. Let $\mathcal{T}$ be a rooted tree. We say that $\mathcal{T}$ is *properly weighted* if each node $\nu \in \mathcal{T}$ has a positive weight $w(\nu)$ that equals the sum of the weights of the children of ν. We assume without loss of generality that $w(\mathrm{root}(\mathcal{T})) = 1$. A *polygonal partition* for a properly weighted tree assigns a convex polygon $P(\nu)$ to each node $\nu \in \mathcal{T}$ such that

- the polygon $P(\mathrm{root}(\mathcal{T}))$ is the unit square;

- for any node ν we have $\mathrm{area}(P(\nu)) = w(\nu)$;

- for any node ν, the polygons assigned to the children of ν form a disjoint partition of $P(\nu)$.

Recall that the aspect ratio of a planar convex region A, denoted by $\mathrm{asp}(A)$, is defined as $\mathrm{asp}(A) := \mathrm{diam}(A)^2 / \mathrm{area}(A)$. The aspect ratio of a polygonal partition is the maximum aspect ratio of any of the polygons in the partition. Our goal is to show that any properly
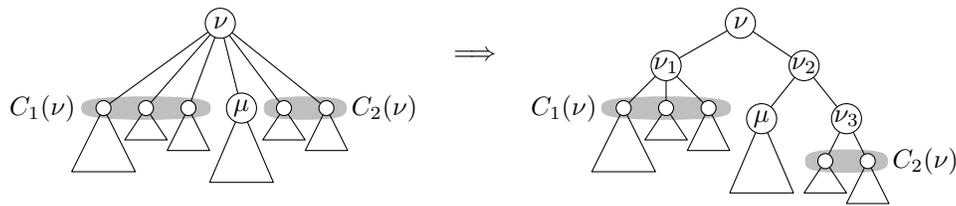
Figure 2: Transforming a tree to a binary tree.

weighted tree admits a *fat polygon partition*, that is, a polygonal partition with small aspect ratio.

We propose two methods for constructing fat polygonal partitions. They both start with transforming the input tree $\mathcal{T}$ into a *binary* tree $\mathcal{T}'$. The nodes of $\mathcal{T}$ are a subset of nodes of $\mathcal{T}'$, and two nodes are in the ancestor-descendant relation in $\mathcal{T}$ if and only if they are in the same relation in $\mathcal{T}'$. The weights assigned to nodes of $\mathcal{T}$ are preserved in $\mathcal{T}'$. Any polygonal partition for $\mathcal{T}'$ restricted to nodes from $\mathcal{T}$ is a polygonal partition for $\mathcal{T}$. Then, for the binary tree $\mathcal{T}'$, it suffices to design a method that cuts the polygon $P(\nu)$ corresponding to a node ν into two polygons of prespecified areas that correspond to ν's children. We propose two such methods: the *angular* method and the *greedy* method. To achieve a polygonal partition for $\mathcal{T}'$, it suffices to recursively apply one of the cutting methods.

The transformation into a binary tree. We transform the input n-node tree $\mathcal{T}$ into a binary tree $\mathcal{T}'$ by replacing every internal node ν of degree greater than two by a collection of nodes whose subtrees together are exactly the subtrees of ν. This can be done in such a way that height($\mathcal{T}'$) = O(height($\mathcal{T}$) + log n) [19]. The number of nodes in $\mathcal{T}'$ is $O(n)$. For completeness we sketch how this transformation is done.

For a node ν, we use $\mathcal{T}_\nu$ to denote the subtree rooted at ν, and $|\mathcal{T}_\nu|$ to denote the number of nodes in $\mathcal{T}_\nu$. The transformation is a recursive process, starting at the root of $\mathcal{T}$. Suppose we reach a node ν. If ν has degree two or less, we just recurse on the at most two children of ν. If ν has degree $k \geq 3$, we proceed as follows. Let $C(\nu)$ be the set of children of ν, and let $\mu \in C(\nu)$ be the child with the largest number of nodes in its subtree. We partition $C(\nu) \setminus \{\mu\}$ into two non-empty subsets $C_1(\nu)$ and $C_2(\nu)$ such that $\sum_{\mu \in C_1(\nu)} |\mathcal{T}_\mu| < |\mathcal{T}_\nu|/2$ and $\sum_{\mu \in C_2(\nu)} |\mathcal{T}_\mu| < |\mathcal{T}_\nu|/2$. We create three new nodes ν_1, ν_2, ν_3 and modify the tree as shown in Figure 2. The weights $w(\nu_1), w(\nu_2), w(\nu_3)$ are set to the sum of the weights of the leaves in their respective subtrees. Finally, we recurse on ν_1, μ, and ν_3. After the procedure has finished we have a (properly weighted) tree in which every node has degree at most two. We remove all degree-1 nodes to obtain our binary tree $\mathcal{T}'$. The height of $\mathcal{T}'$ is at most 2(height($\mathcal{T}$) + log n), because every time we go down two levels in $\mathcal{T}'$ we either pass through an original node from $\mathcal{T}$ or the number of nodes in the subtree halves.
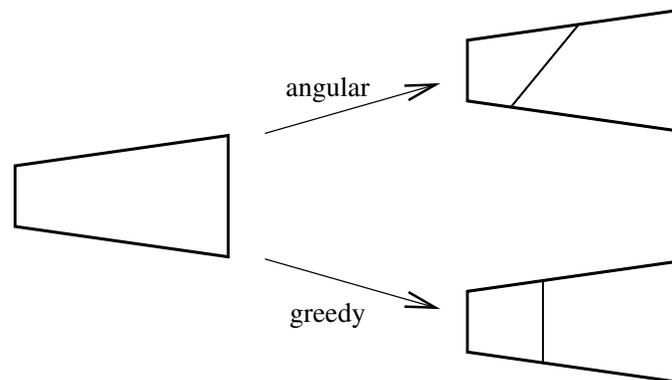
Figure 3: Sample executions of our cutting methods.

Methods for cutting a polygon.   Suppose we have to cut a convex polygon $P(\nu)$ into two subpolygons. Note that if we fix an orientation for the cut, then there are only two choices left for the cut because the areas of the subpolygons are prespecified. (The two choices correspond to having the smaller of the two areas to the left or to the right of the cut.) Our cutting methods, depicted in Figure 3 are the following.

- *Angular*: Let c denote the cut, that is, the line segment separating the two subpolygons of $P(\nu)$. We select the orientation of c such that we maximize $\min\{\text{angle}(c, e) : e$ is an edge of the input polygon$\}$, where $\text{angle}(c, e)$ is the smaller of the angles between the lines $\ell(c)$ and $\ell(e)$ containing c and e, respectively. In other words, we cut in a direction as different as possible from all the orientations determined by the edges of the polygon. We then take any of the two cuts of the selected orientation.

- *Greedy*: The greedy method selects the cut that minimizes the maximum of the aspect ratios of the two subpolygons.

Experiments.   To get an idea of the relative performance of the two methods we implemented them and performed some experiments. Table 1 shows the results on two hierarchies. One is synthetic and was generated using a random process, the other is the home directory with all subfolders of one of the authors. Leaves in the latter hierarchy are the files in any of the folders, and the weight of a leaf is the size of the corresponding file. For comparison, we also ran two additional partitioning methods. Both of these methods use the transformation of the input into a binary tree. The *random* method always makes a cut in a random direction. In the *greedy rectangular* method, all polygons are rectangles and all cuts are parallel to the sides of the original rectangle. The method always greedily chooses the cut that is perpendicular to the longer side, which maximizes the aspect ratio of the two subpolygons resulting from the cut.

      In all our tests, the methods partitioned a square. The greedy method performs best, closely followed by the angular method. Interestingly, the greedy rectangular method performs even worse than the random method—apparently restricting to rectangles is a

Method	Synthetic Data		Home Folder	
	Average	Maximum	Average	Maximum
Angular	3.79	13.19	3.87	20.11
Greedy	2.56	6.79	2.57	8.39
Random	5.79	355.14	6.26	1609.66
Greedy Rectangular	3.52	1445.99	24.49	230308.30

Table 1: Aspect ratios of partitions generated by various methods

very bad idea as far as aspect ratio is concerned.

In the next two sections we prove that both the angular method and the greedy method construct a partition in which the aspect ratios are $O(\text{poly}(\text{height } \mathcal{T} + \log n))$. For the angular method, the proof is simpler and gives a better bound on the worst-case aspect ratio. We present the more complicated proof for the greedy method because it is the most natural method and it has the best performance in practice.

3 Analysis of angular partitions

The idea behind the angular partitioning method is that a polygon with large aspect ratio must have two edges that are almost parallel. Hence, if we avoid using partition lines whose orientations are too close to each other, then we can control the aspect ratio of our subpolygons. Next we make this idea precise.

Let U be the initial unit square that we partition, and let $\phi > 0$ be a parameter. Recall that for two line segments e and e', we use angle(e, e') to denote the smaller angle defined by the lines $\ell(e)$ and $\ell(e')$ containing e and e', respectively. We define a convex polygon $P \subset U$ to be a *ϕ-separated polygon* if it satisfies the following condition. For any two distinct edges e and e', we have:

(i) angle$(e, e') \geq \phi$; or

(ii) e is contained in U's top edge and e' is contained in U's bottom edge (or vice versa); or

(iii) e is contained in U's left edge and e' is contained in U's right edge (or vice versa).

Lemma 1. *The aspect ratio of a ϕ-separated polygon P is $O(1/\phi)$.*

Proof. Let $d := \text{diam}(P)$ and let uv be a diagonal of P that has length d. Consider the bounding box B of P that has two edges parallel to uv. We call the edge of B parallel to and above uv its *top edge*, and the edge of B parallel to and below uv its *bottom edge*. Let r be a vertex of P on the top edge of B and let s be a vertex on its bottom edge—see Fig. 4. Let e_1 and e_2 be the edges of P incident to r, and let e_3 and e_4 be the edges incident to s. Let the angles $\alpha_1, \ldots, \alpha_4$ be defined as in Fig. 4. We distinguish two cases.
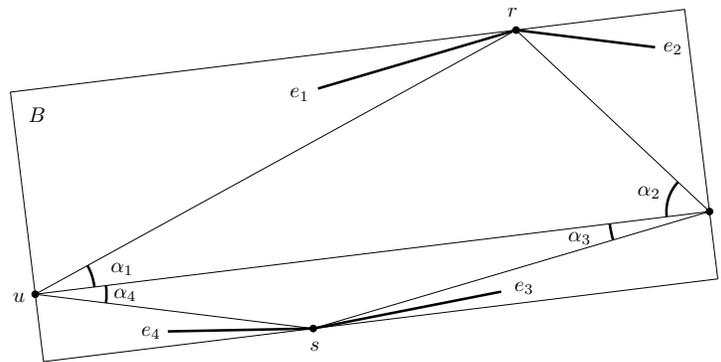
Figure 4: Illustration for the proof of Lemma 1.

- *Case (a): none of e_1, e_2, e_3, e_4 are parallel.*
  By condition (i), this implies that the angles any two edges make is at least ϕ. Hence, we have

$$\phi \leq \text{angle}(e_1, e_3) \leq \max(\alpha_1, \alpha_3), \tag{1}$$
$$\phi \leq \text{angle}(e_2, e_4) \leq \max(\alpha_2, \alpha_4), \tag{2}$$
$$\phi \leq \text{angle}(e_1, e_4) \leq \alpha_1 + \alpha_4, \tag{3}$$
$$\phi \leq \text{angle}(e_2, e_3) \leq \alpha_2 + \alpha_3. \tag{4}$$

By (3) we have $\alpha_1 + \alpha_4 \geq \phi$. Now assume without loss of generality that $\alpha_1 \geq \phi/2$. If $\alpha_2 \geq \phi/2$ as well, then

$$\text{area}(uvr) \geq (d^2/4) \cdot \sin(\phi/2).$$

Since $uvr \subset P$, this implies that

$$\text{asp}(P) \leq \frac{d^2}{(d^2/4) \cdot \sin(\phi/2)} = O(1/\phi).$$

If $\alpha_2 < \phi/2$, then we use (2) and (4) to conclude that $\alpha_4 \geq \phi$ and $\alpha_3 \geq \phi/2$. Hence, we now have $\text{area}(uvs) \geq (d^2/4) \cdot \sin(\phi/2)$, which implies that $\text{asp}(P) = O(1/\phi)$.

- *Case (b): some edges in e_1, e_2, e_3, e_4 are parallel.*
  By conditions (i)–(iii), two edges of P can be parallel only if they are contained in opposite edges of U. Hence, $|rs| \geq 1$. Moreover, since uv defines the diameter we have $|uv| \geq |rs|$. Let $\alpha := \text{angle}(uv, rs)$, as illustrated in Fig. 5. If $\alpha \geq \min\{\phi, \pi/4\}$, this is easily seen to imply that $\text{area}(P) \geq \text{area}(urvs) = \Omega(\phi)$, which means that $\text{asp}(P) = O(1/\phi)$. Now consider the case where $\alpha < \phi$ and $\alpha < \pi/4$. We show that this leads to a contradiction. Let $x := uv \cap rs$ and assume without loss of generality that, as in Figure 5, we have $\alpha = \angle rxv$. Then $\text{angle}(e_1, e_3) \leq \alpha < \phi$. By conditions (i)–(iii) this can only happen if e_1 and e_3 are contained in opposite edges of U, if we assume that r and s are chosen such that they maximize the distances between r and s, which can be done without loss of generality. We have $\text{angle}(e_3, rs) \leq \text{angle}(uv, rs) = \alpha \leq \phi < \pi/4$.
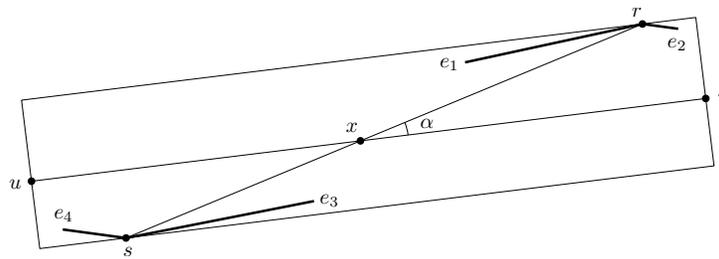
Figure 5: The case of parallel edges.

However, it is impossible to place two points r and s on opposite sides of U such that angle$(e_3, rs) < \pi/4$. The smallest angle one can obtain is $\pi/4$.

$\square$

To construct a polygonal partition we use the procedure described in Section 2. Thus, we first transform the input tree $\mathcal{T}$ into a corresponding binary tree $\mathcal{T}'$. Next, we recursively apply the angular cutting method to $\mathcal{T}'$, that is, at each node ν, we cut the polygon $P(\nu)$, using a cut c that maximizes the minimum angle c makes with any of the edges of $P(\nu)$.

Lemma 2. *Let $P(\nu)$ be the subpolygon generated by the algorithm above for a node ν at level k in $\mathcal{T}'$. Then $P(\nu)$ is a $(\pi/(2k+6))$-separated polygon.*

Proof. The proof is by induction on k.

For $k = 0$, we have $\nu = \text{root}(\mathcal{T}')$ and $P(\nu)$ is a unit square. Hence, $P(\nu)$ is $(\pi/2)$-separated and therefore also $(\pi/6)$-separated.

For $k > 0$ we argue as follows. By the induction hypothesis, the polygon $P(\mu)$ corresponding to the parent μ of ν is $(\pi/(2k+4))$-separated. Moreover, by construction it has at most $4 + (k-1) = k+3$ edges. Consider the sorted (circular) sequence of angles that these edges make with the x-axis. By the pigeon-hole principle, there must be two adjacent angles that are at least $\pi/(k+3)$ apart. Hence, the cut c that is chosen to partition $P(\mu)$ makes an angle at least $\pi/(2k+6)$ with all edges of $P(\mu)$, and by the induction hypothesis, all the other angles are at least $\pi/(2k+4)$. $\square$

Recall that the height of the binary tree $\mathcal{T}'$ is $O(\text{height}(\mathcal{T}) + \log n)$. Hence, we get the following theorem.

Theorem 1. *Let $\mathcal{T}$ be a properly weighted tree with n nodes. Then the angular partitioning method constructs a polygonal partition for $\mathcal{T}$ whose aspect ratio is $O(\text{height}(\mathcal{T}) + \log n)$.*

4   Analysis of greedy partitions

We now turn our attention to the greedy method, which at each step chooses a cut c that minimizes the maximum aspect ratio of the two subpolygons resulting from the cut. The main component of our proof that polygonal 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 areas so that the aspect ratios of the new subpolygons are bounded.

Note that the lemma requires that the number of vertices in the polygon be bounded. If the number of vertices is unbounded, the polygon may become arbitrarily close to a circle. In this case there is no good cut if one of the resulting polygons has to be much smaller than the other.

The proof of the lemma is long and consists of a case analysis. An impatient reader may prefer to omit the proof and move directly to Theorem 2.

Lemma 3 (Good cuts). *Let $P \subset \mathbb{R}^2$ be a convex polygon with k vertices, 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.*

- $\mathrm{area}(P_1) = a \cdot \mathrm{area}(P)$, *and* $\mathrm{area}(P_2) = (1 - a) \cdot \mathrm{area}(P)$.

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

Proof. We distinguish two cases, depending on whether $a \leq 1/k^2$ (that is, we are cutting off a relatively small subpolygon) or not.

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

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

Let ℓ be the angular bisector at v. Consider the cut c orthogonal to ℓ such that $\mathrm{area}(P_1) = a \cdot \mathrm{area}(P)$, where P_1 is the subpolygon induced by c having v as a vertex— see Figure 6(a) for an illustration. Let P_2 be the other subpolygon. Clearly, P_1 and P_2 are convex polygons of the required area with at most $k + 1$ vertices each. Therefore, it remains to bound the aspect ratios of P_1 and P_2.

Since $P_2 \subset P$, we have

$$\mathrm{asp}(P_2) = \frac{\mathrm{diam}(P_2)^2}{\mathrm{area}(P_2)} \leq \frac{\mathrm{diam}(P)^2}{(1 - a) \cdot \mathrm{area}(P)} = \frac{\mathrm{asp}(P)}{1 - a} < \mathrm{asp}(P)\left(1 + 2a\right)$$
$$< \mathrm{asp}(P)\left(1 + \frac{2}{k^2}\right) < \mathrm{asp}(P)\left(1 + \frac{1}{k}\right).$$

We next bound $\mathrm{asp}(P_1)$. Let x_1, x_2 be the two endpoints of the cut c, and let t be the distance between x_1 and x_2. Let h be the distance from v to c. We distinguish between two subcases.
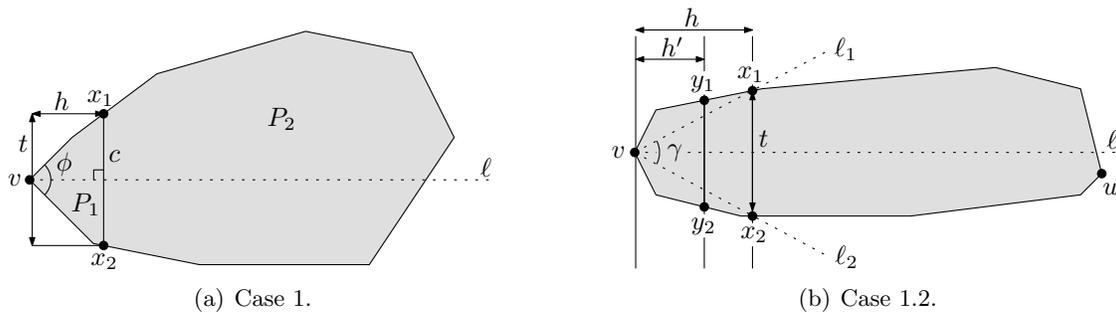
Figure 6: Partitioning P into P_1 and P_2 when $a \leq 1/k^2$.

Case 1.1: $t \geq h/k^2$. Since P is convex, the triangle vx_1x_2 is contained in P_1. Therefore,

$$\text{area}(P_1) \geq h \cdot t/2 \geq h^2/(2k^2).$$

On the other hand, since c 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

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

$$\leq 2 \cdot h / \tan(\pi/k) \leq 2 \cdot h \cdot k/\pi.$$

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

$$\text{asp}(P_1) = \frac{\text{diam}(P_1)^2}{\text{area}(P_1)} < \frac{(h + 2 \cdot h \cdot k/\pi)^2}{h^2/(2k^2)} < k^5.$$

Case 1.2: $t < h/k^2$. Let ℓ_1 be the line passing through v and x_1, and let ℓ_2 be the line passing through v and x_2. Let γ be the angle between ℓ_1 and ℓ_2. Observe that P_2 is contained between ℓ_1 and ℓ_2. Therefore, if u is the point in P_2 farthest away from v we have

$$\frac{\gamma}{2\pi}\pi|uv|^2 \geq \text{area}(P_2),$$

where $|uv|$ denotes the length of the segment uv. It follows that

$$\text{diam}(P)^2 \geq |uv|^2 \geq \frac{2}{\gamma}(1 - a) \cdot \text{area}(P).$$

Therefore,

$$\text{asp}(P) = \frac{\text{diam}(P)^2}{\text{area}(P)} \geq \frac{2}{\gamma}(1 - a) \geq \frac{2}{\gamma}\left(1 - \frac{1}{k^2}\right).$$

We now give an upper bound on the diameter of P_1. Assume without loss of generality that $|vx_2| \geq |vx_1|$. Consider a segment y_1y_2 parallel to x_1x_2 with $y_1, y_2 \in \partial P$ such that y_1y_2 lies between x_1x_2 and v—see Figure 6(b). Let h' be the distance between v and y_1y_2. We first argue that $|y_1y_2| \leq 2t$.

Assume for the sake of contradiction that $|y_1 y_2| > 2t$. 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 $|y_1 y_2| > |x_1 x_2|$, the lines g_1 and g_2 intersect in a point w such that P_2 is contained in the triangle $x_1 x_2 w$. Furthermore, the polygon $vy_1 x_1 x_2 y_2$ is contained in P_1. If $h' \geq h/2$, then the area of the triangle $vy_1 y_2$ is greater or equal to the area of the triangle $x_1 x_2 w$. Therefore, $\text{area}(P_1) \geq \text{area}(P_2)$, contradicting the fact that $a \leq 1/k^2$. If, on the other hand, $h' < h/2$, then the area of the quadrilateral $y_1 x_1 x_2 y_2$, is greater than the area of the triangle $x_1 x_2 w$, again implying that $\text{area}(P_1) \geq \text{area}(P_2)$, a contradiction. Therefore, we obtain that $|y_1 y_2| \leq 2t$.

It now follows that any point $q \in P_1$ is at distance at most $2t$ from the line segment vx_2. Moreover, we have $t < h/k^2 \leq |vx_2|/k^2$. Hence,

$$\text{diam}(P_1) = \max_{q, q' \in P_1} |qq'| \leq 2t + |vx_2| + 2t \leq |vx_2| \left(1 + \frac{4}{k^2} \right).$$

Let x^* be the point on the line segment $x_1 x_2$ that is closest to v. Since $|vx_2| \geq h$, we have

$$\text{area}(P_1) \geq \frac{\gamma}{2\pi} \pi |vx^*|^2 \geq \frac{\gamma}{2}(|vx_2| - t)^2 \geq \frac{\gamma}{2}|vx_2|^2 \left(1 - \frac{1}{k^2} \right).$$

Therefore,

$$\begin{aligned} \text{asp}(P_1) = \frac{\text{diam}(P_1)^2}{\text{area}(P_1)} &\leq \frac{2}{\gamma} \cdot \frac{(1 + 4/k^2)^2}{1 - 1/k^2} \leq \text{asp}(P)\frac{(1 + 4/k^2)^2}{(1 - 1/k^2)^2} \\ &\leq \text{asp}(P) \cdot (1 + 6/k^2)^2 \leq \text{asp}(P) \cdot (1 + 2/k)^2 \\ &\leq \text{asp}(P) \cdot (1 + 6/k). \end{aligned}$$

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

Case 2.1: $\text{asp}(P) \leq k^6$. In this case any cut giving the two subpolygons P_1 and P_2 the required areas works. Indeed,

$$\text{asp}(P_1) = \frac{\text{diam}(P_1)^2}{\text{area}(P_1)} \leq \frac{\text{diam}(P)^2}{a \cdot \text{area}(P)} \leq k^2 \cdot \text{asp}(P) \leq k^8$$

and

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

Case 2.2: $\text{asp}(P) > k^6$. Pick points $v_1, v_2 \in P$, such that $|v_1 v_2| = \text{diam}(P)$. For each $z \in [0, \text{diam}(P)]$, let $\ell(z)$ be a line normal to $v_1 v_2$ that is at distance z from v_1 and intersects P. Note that $\ell(0)$ contains v_1 and $\ell(\text{diam}(P))$ contains v_2. Define $f(z)$ to be the length of the intersection of P with $\ell(z)$. Observe that

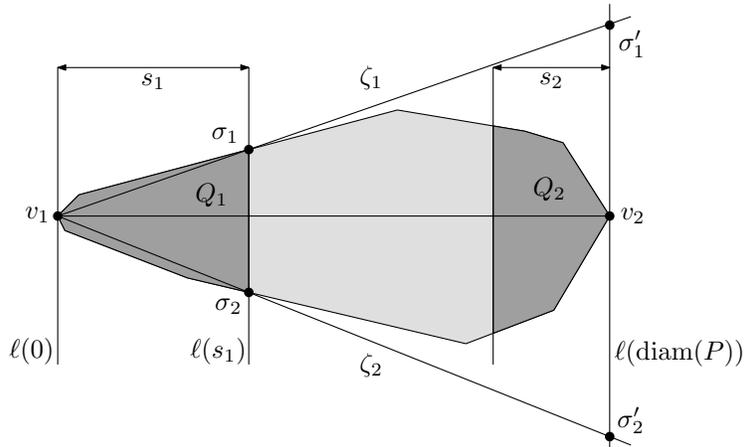$$\text{area}(P) = \int_{z=0}^{\text{diam}(P)} f(z)dz$$

Figure 7: Partitioning P into P_1 and P_2, when $\alpha > 1/k^2$: Case 2.2.

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

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

Let Q_1 be the part of P that is contained between $\ell(0)$ and $\ell(s_1)$. Similarly, let Q_2 be the part of P that is contained between $\ell(\operatorname{diam}(P) - s_2)$ and $\ell(\operatorname{diam}(P))$. Clearly, both Q_1 and Q_2 are convex polygons with at most $k + 1$ vertices. First, we will show that

$$\min\left\{ \frac{\operatorname{area}(Q_1)}{s_1}, \frac{\operatorname{area}(Q_2)}{s_2} \right\} \le \frac{\operatorname{area}(P)}{\operatorname{diam}(P)}$$

Assume for a contradiction that both $\operatorname{area}(Q_1)/s_1$ and $\operatorname{area}(Q_2)/s_2$ are greater than $\operatorname{area}(P)/\operatorname{diam}(P)$. It follows that there exist $z_1 \in [0, s_1]$ and $z_2 \in [\operatorname{diam}(P) - s_2]$ such that $f(z_1) > \operatorname{area}(P)/\operatorname{diam}(P)$ and $f(z_2) > \operatorname{area}(P)/\operatorname{diam}(P)$. Since P is convex, f is a bitonic function. Therefore, for each $z \in [z_1, z_2]$, we have $f(z) > \operatorname{area}(P)/\operatorname{diam}(P)$. It follows that

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

a contradiction.
We can therefore assume without loss of generality that

$$\frac{\operatorname{area}(Q_1)}{s_1} \le \frac{\operatorname{area}(P)}{\operatorname{diam}(P)}.$$

Note that this implies

$$s_1 \ge a \cdot \operatorname{diam}(P).$$

We set $P_1 = Q_1$, and $P_2 = P \setminus Q_1$. It remains to bound $\operatorname{asp}(P_1)$ and $\operatorname{asp}(P_2)$.

By the convexity of P, we have

$$\text{area}(P) \geq \max_{z \in [0, \text{diam}(P)]} f(z) \cdot \text{diam}(P)/2.$$

Since $\text{asp}(P) > k^6$, it follows that

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

This implies that P is contained inside a rectangle with one edge of length $\text{diam}(P)$ parallel to $v_1 v_2$, and one edge of length $\frac{4}{k^6} \cdot \text{diam}(P)$ normal to $v_1 v_2$. Thus,

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

Let σ_1, σ_2 be the two points where $\ell(s_1)$ intersects ∂P. Let ζ_1, ζ_2, be the lines passing through v_1 and σ_1, and v_1 and σ_2, respectively. Let also σ_1' and σ_2' be the points where ζ_1 and ζ_2, respectively, intersect $\ell(\text{diam}(P))$—see Figure 7.

By the convexity of P and P_1, we have

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

Since $\text{area}(P_1) = a \cdot \text{area}(P)$, it follows that $s_1 \leq \sqrt{a} \cdot \text{diam}(P)$. Using that $a > 1/k^2$ we can now derive

$$
\begin{aligned}
\text{asp}(P_1) = \frac{\text{diam}(P_1)^2}{\text{area}(P_1)} &\leq \frac{(s_1 + 4 \cdot \text{diam}(P)/k^6)^2}{\text{area}(P_1)} \\
&\leq \frac{(\sqrt{a} \cdot \text{diam}(P) + 4 \cdot \text{diam}(P)/k^6)^2}{a \cdot \text{area}(P)} \\
&< \frac{\text{diam}(P)^2}{\text{area}(P)} \cdot \left(1 + \frac{4}{k^6 \sqrt{a}} \right)^2 \leq \text{asp}(P) \cdot \left(1 + \frac{8}{k^4} + \frac{16}{k^{16}} \right) \\
&\leq \text{asp}(P) \cdot \left(1 + \frac{1}{k} \right).
\end{aligned}
$$

Since f is bitonic, it follows that

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

Therefore,

$$\frac{\text{area}(P_2)}{\text{diam}(P) - s_1} \geq \frac{\text{area}(P_1)}{s_1}.$$

Because P_2 is contained in a rectangle with one edge of length $\text{diam}(P) - s_1$ parallel to $v_1 v_2$, and one edge of length $\frac{4}{k^6} \cdot \text{diam}(P)$ normal to $v_1 v_2$, we have

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

Putting everything together, we get

$$\text{asp}(P_2) = \frac{\text{diam}(P_2)^2}{\text{area}(P_2)} \leq \frac{(\text{diam}(P) \cdot (1 + 4/k^6) - s_1)^2}{(1 - a) \cdot \text{area}(P)}$$

$$\leq \text{asp}(P) \cdot \left(\frac{1 + 4/k^6 - a}{\sqrt{1 - a}}\right)^2 \leq \text{asp}(P) \cdot \left(1 + 4 \cdot \frac{\sqrt{2}}{k^6}\right)^2$$

$$\leq \text{asp}(P) \cdot \left(1 + \frac{1}{k^2}\right)^2 \leq \text{asp}(P) \cdot \left(1 + \frac{3}{k^2}\right)$$

$$\leq \text{asp}(P) \cdot \left(1 + \frac{1}{k}\right).$$

This concludes the proof. $\square$

Now we have all the necessary tools to prove a bound on the aspect ratio of the polygonal partition constructed by the greedy method.

Theorem 2. *Let $\mathcal{T}$ be a properly weighted tree with n nodes. Then the greedy partitioning method constructs a polygonal partition for $\mathcal{T}$ whose aspect ratio is $O\left((\text{height}(\mathcal{T}) + \log n)^8\right)$.*

Proof. Recall from Section 2 that our algorithm starts by transforming $\mathcal{T}$ to a binary tree $\mathcal{T}'$ of height $O(\text{height}(\mathcal{T}) + \log n)$. This is done in such a way that a polygonal partition for $\mathcal{T}'$ induces a polygonal partition for $\mathcal{T}$ of the same (or better) aspect ratio. We then recursively apply the greedy cutting strategy to $\mathcal{T}'$. Hence, it suffices to show that a recursive application of the greedy strategy to a binary tree of height h produces a polygonal partition of aspect ratio $(h + 3)^8$.

Let $A(i)$ be the worst-case aspect ratio of a polygon $P(\nu)$ produced by the greedy method over all nodes ν at depth i in the tree. Note that $P(\text{root}(\mathcal{T}))$ is a square and each polygon at depth i has at most $i + 4$ vertices. By Lemma 3 we thus have

$$A(i) \leq \begin{cases} 2 & \text{if } i = 0, \\ \max\left\{(i+3)^8, \left(1 + \frac{6}{i+3}\right) \cdot A(i-1)\right\} & \text{if } i > 0. \end{cases}$$

We can now prove by induction that $A(i) \leq (i+3)^8$. Indeed, for $i = 0$ this obviously holds, and for $i > 0$ we have

$$A(i) \leq \max\left\{(i+3)^8, \left(1 + \frac{6}{i+3}\right) \cdot A(i-1)\right\}$$

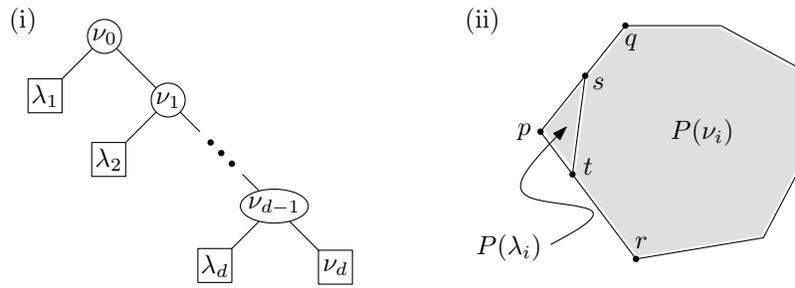$$\leq \max\left\{(i+3)^8, \left(1 + \frac{6}{i+3}\right) \cdot (i+2)^8\right\}$$

Figure 8: (i) Structure of the tree for the lower-bound construction. (ii) Illustration for the proof of Lemma 4.

and

$$\left(1 + \tfrac{6}{i+3}\right) \cdot (i+2)^8 = \left\{ \left(1 + \tfrac{6}{i+3}\right) \cdot \left(\tfrac{i+2}{i+3}\right)^8 \right\} \cdot (i+3)^8$$

$$= \left\{ \frac{1 + \tfrac{6}{i+3}}{\left(1 + \tfrac{1}{i+2}\right)^8} \right\} \cdot (i+3)^8$$

$$< \left\{ \frac{1 + \tfrac{6}{i+3}}{1 + \tfrac{8}{i+2}} \right\} \cdot (i+3)^8$$

$$< (i+3)^8.$$

Hence, $A(h) < (h+3)^8$, which finishes the proof. □

5   A lower bound for polygonal partitions

In the previous sections we have seen that any properly weighted tree $\mathcal{T}$ with n nodes admits a polygonal partition whose aspect ratio is $O(\text{height}(\mathcal{T}) + \log n)$. In this section we prove this is almost tight in the worst case, by exhibiting a tree for which any polygonal partition has aspect ratio $\Omega(\text{height}(\mathcal{T}))$.

We start with an easy lower bound on the aspect ratio of a convex polygon in terms of its smallest angle.

Observation 1. *Let P be a convex polygon and let α be the smallest interior angle of P. Then the aspect ratio of P is at least $2/\alpha$.*

Proof. Let v be a vertex of P whose interior angle is α. Then P is contained in the circular sector with radius $\text{diam}(P)$ and angle α whose apex is at v. This sector has area $(\alpha/2) \cdot \text{diam}(P)^2$, from which the observation readily follows. □

Next we show how to construct, for any given height h, a weighted tree $\mathcal{T}$ of height h such that any polygonal partition for $\mathcal{T}$ has a region with a very small angle. The lower bound on the aspect ratio then follows immediately from Observation 1.

The structure of $\mathcal{T}$ is depicted in Fig. 8(i). The tree $\mathcal{T}$ has $h+1$ nodes $\nu_0, \ldots, \nu_h$ that form a path, and h other (leaf) nodes $\lambda_1, \ldots, \lambda_h$ branching off the path. The idea will be to choose the weights of the leaves very small, so that the only way to give the region $P(\lambda_i)$ the required area, is to cut off a small triangle from $P(\nu_i)$. Then we will argue that one of these triangles must have a small angle.

To make this idea precise we define

$$x_i := \begin{cases} 1 & \text{if } i = 0, \\ x_{i-1}/(2\sqrt{h}) & \text{if } 0 < i \leq h, \end{cases}$$

and we set $w(\lambda_i) := x_{i-1}^2/(4h)$ for $1 \leq i \leq h$. Note that defining the weights for $\lambda_1, \ldots, \lambda_h$ implicitly defines the weights for $\nu_1, \ldots, \nu_h$ as well.

Lemma 4. *If each region created in a polygonal partition for $\mathcal{T}$ has aspect ratio at most h, then we have for $0 \leq i \leq h$,*

(i) $P(\lambda_{i-1})$ is a triangle (if λ_{i-1} exists, that is, if $i \neq 0$);

(ii) $P(\nu_i)$ has $i + 4$ sides, each of length at least x_i.

Proof. We will prove the lemma by induction on i. For $i = 0$ the lemma is obviously true, since $P(\nu_0)$ is the unit square.

Now let $i > 0$. By the induction hypothesis, each side of $P(\nu_{i-1})$ has length at least x_{i-1}. If $P(\lambda_i)$ fully contained an edge of $P(\nu_{i-1})$, its diameter would therefore be more than x_{i-1}. But then its aspect ratio would be

$$\frac{\text{diam}(P(\lambda_i))^2}{\text{area}(P(\lambda_i))} \geq \frac{x_{i-1}^2}{w(\lambda_i)} = \frac{x_{i-1}^2}{x_{i-1}^2/4h} = 4h,$$

which contradicts the assumptions. Hence, $P(\lambda_i)$ is a triangle, as claimed. It also follows that $P(\nu_i)$ has $i + 4$ sides. It remains to show that these sides have length at least x_i.

Let st be the segment that cuts $P(\nu_{i-1})$ into $P(\nu_i)$ and $P(\lambda_i)$, let p be the corner of $P(\nu_{i-1})$ that is cut off, and let q and r be the corners of $P(\nu_{i-1})$ adjacent to p—see Fig. 8(ii). All sides of $P(\nu_i)$ except qs, st, and tr have length at least x_{i-1} so they definitely have length at least x_i.

It is easy to see that the angles of any polygon $P(\nu_j)$ are at least $\pi/2$—indeed, when a corner is cut off from some $P(\nu_j)$, the two new angles appearing in $P(\nu_{j+1})$ are larger than the angle at the corner that is cut off. Hence, st is the longest edge of the triangle $pst = P(\lambda_i)$, and we have

$$|st| \geq \sqrt{\text{area}(P(\lambda_i))} = \sqrt{w(\lambda_i)} = \sqrt{\frac{x_{i-1}^2}{4h}} = \frac{x_{i-1}}{2\sqrt{h}} = x_i.$$

Next we show that qs has length at least x_i; the argument for tr is similar. Note that $|ps| \leq \sqrt{h \cdot w(\lambda_i)}$, otherwise $P(\lambda_i)$'s aspect ratio would be larger than h. Hence, we have

$$
\begin{aligned}
|qs| &= |pq| - |ps| \\
&\geq x_{i-1} - |ps| \qquad \text{(induction hypothesis)} \\
&\geq x_{i-1} - \sqrt{h \cdot w(\lambda_i)} \\
&= x_{i-1} - \sqrt{h \cdot \frac{x_{i-1}^2}{4h}} \\
&= \frac{x_{i-1}}{2} \\
&\geq x_i.
\end{aligned}
$$

$\square$

Next we show that one of the triangles $P(\lambda_i)$ must have large aspect ratio.

Lemma 5. *If each region created in a polygonal partition for $\mathcal{T}$ has aspect ratio at most h, then there is a region $P(\lambda_i)$ where one of whose interior angles is at most $2\pi/(h+4)$.*

Proof. By the previous lemma, the region $P(\nu_h)$ has $h+4$ sides. The sum of the interior angles of a $(h+4)$-gon is exactly $(h+2) \cdot 2\pi$, so one of the interior angles of $P(\nu_h)$ must be at least

$$
\frac{(h+2) \cdot 2\pi}{h+4} = 2\pi - \frac{4\pi}{h+4}.
$$

If the two edges meeting at some corner p of $P(\nu_h)$ make an angle of at least $2\pi - 4\pi/(h+4)$ in $P(\nu_h)$, then they must make an angle of at most $4\pi/(h+4)$ in a region $P(\lambda_i)$ adjacent to p. $\square$

Theorem 3. *For any h, there is a weighted tree $\mathcal{T}$ of height h such that any polygonal partition for $\mathcal{T}$ has aspect ratio $\Omega(h)$.*

Proof. Consider the tree $\mathcal{T}$ described above. Assume the aspect ratio of the polygonal partition for $\mathcal{T}$ is less than h—otherwise we are done. Then by Lemma 5 there is a region with interior angle $4\pi/(h+4)$. By Observation 1 this region has aspect ratio at least $(h+4)/2\pi = \Omega(h)$. $\square$

6 Partitions with slack

Recall that a rectangular partition is a polygonal partition in which all polygons are rectangles. We show that if we allow a small distortion of the areas of the rectangles, then there exists a rectangular partition of small aspect ratio. This result can also be obtained for partitions of a hypercube in $\mathbb{R}^d$. We now define more precisely which type of distortion we allow.

Let $\mathcal{T}$ be a properly weighted tree. A *rectangular partition with ε-slack* in $\mathbb{R}^d$ for $\mathcal{T}$ assigns a d-dimensional hyperrectangle $R(\nu)$ to each node $\nu \in \mathcal{T}$ such that

- the hyperrectangle $R(\text{root}(\mathcal{T}))$ is the unit hypercube in $\mathbb{R}^d$;

- for any two nodes ν, μ such that μ is a child of ν we have

$$(1 - \varepsilon) \cdot \frac{\text{vol}(R(\mu))}{w(\mu)} \leq \frac{\text{vol}(R(\nu))}{w(\nu)} \leq \frac{\text{vol}(R(\mu))}{w(\mu)}.$$

- for any node ν, the hyperrectangles assigned to the children of ν have pairwise disjoint interiors and are contained in $R(\nu)$.

Observe that as we go down the tree $\mathcal{T}$, the volumes of the hyperrectangles can start to deviate more and more from their weights. However, the relative volumes of the hyperrectangles of the children of a node ν stay roughly the same and together they still cover $R(\nu)$ almost entirely.

For convenience, we will work with the *rectangular aspect ratio* of a d-dimensional hyperrectangle rather than using the aspect-ratio definition given earlier. The rectangular aspect ratio $\text{asp}_{\text{rect}}(R)$ of hyperrectangle R with side lengths $s_1, s_2, \ldots, s_d$ is defined as $\text{asp}_{\text{rect}}(R) := \frac{\max_i s_i}{\min_i s_i}$. It can easily be shown that for 2-dimensional rectangles, the aspect ratio and the rectangular aspect ratio are within a constant factor. We define the rectangular aspect ratio of a rectangular partition as the maximum rectangular aspect ratio of any of the hyperrectangles in the partition.

Our algorithm to construct a rectangular partition always cuts perpendicular to the longest side of a hyperrectangle $R(\nu)$. Since we can shrink each of the hyperrectangles of the children of ν by a factor $1 - \varepsilon$, we have some extra space to keep the aspect ratios under control. We will prove that this means that the longest-side-first strategy can ensure a rectangular aspect ratio of $1/\varepsilon$. The basic tool is the following lemma.

Lemma 6. *Let $0 < \varepsilon < 1/3$, and let R be a hyperrectangle with $\text{asp}_{\text{rect}}(R) \leq 1/\varepsilon$. Let $S = \{w_1, \ldots, w_k\}$ be a set of weights with $\sum_{i=1}^{k} w_i = (1 - \varepsilon) \cdot \text{vol}(R)$. Then there exists a set $\{R_1, \ldots, R_k\}$ of pairwise disjoint hyperrectangles, each contained in R, such that for all $1 \leq i \leq k$ we have $\text{vol}(R_i) = w_i$ and $\text{asp}_{\text{rect}}(R_i) \leq 1/\varepsilon$.*

Proof. We will prove this by induction on k. The case $k = 1$ is trivial, so now assume $k > 1$. For a subset $S' \subset S$, we define $w(S') := \sum_{w_i \in S'} w_i$. Assume without loss of generality that $w_1 = \max_i w_i$. There are two cases.

- If $w_1 \leq (1 - \varepsilon) \cdot w(S)$, then we can split S into subsets S' and S'' such that $w(S') \geq \varepsilon \cdot w(S)$ and $w(S'') \geq \varepsilon \cdot w(S)$. Indeed, if i^* is the minimum index such that $\sum_{i=1}^{i^*} w_i \geq \varepsilon \cdot w(S)$, then $\sum_{i=1}^{i^*} w_i \leq (1 - \varepsilon) \cdot w(S)$ since $\varepsilon < 1/3$. Hence, setting $S' = \{w_1, \ldots, w_{i^*}\}$ satisfies the condition.

  We now cut R perpendicular to its longest side into hyperrectangles R' and R'' such that
  $$\text{vol}(R') = \text{vol}(R) \cdot \frac{w(S')}{w(S)} \quad \text{and} \quad \text{vol}(R'') = \text{vol}(R) \cdot \frac{w(S'')}{w(S)}.$$

Note that this implies that

$$w(S') = \frac{w(S)}{\text{vol}(R)} \cdot \text{vol}(R') = (1 - \varepsilon) \cdot \text{vol}(R').$$

Since $w(S') \geq \varepsilon \cdot w(S)$ and the cut is perpendicular to the longest side, we have $\text{asp}_{\text{rect}}(R') \leq 1/\varepsilon$. Hence, by induction there exists a set of pairwise disjoint hyperrectangles, each contained in R', whose volumes are the weights in S' and whose aspect ratios are at most $1/\varepsilon$. Similarly, there is a collection of suitable hyperrectangles contained in R'' for the weights in S''. Hence, we have found a set of hyperrectangles for S satisfying all the conditions.

- If $w_1 > (1-\varepsilon) \cdot w(S)$ we split R into R_1 and R' by cutting perpendicular to the longest side, such that $\text{vol}(R_1) = w_1$. Since

$$w_1 > (1 - \varepsilon) \cdot w(S) = (1 - \varepsilon)^2 \cdot \text{vol}(R) > \varepsilon \cdot \text{vol}(R),$$

we have $\text{asp}_{\text{rect}}(R_1) \leq 1/\varepsilon$. Furthermore, since $w(S) = (1 - \varepsilon) \cdot \text{vol}(R)$ we certainly have $w_1 < (1 - \varepsilon) \cdot \text{vol}(R)$, and so

$$\text{vol}(R') = \text{vol}(R) - w_1 > \varepsilon \cdot \text{vol}(R).$$

This implies that $\text{asp}_{\text{rect}}(R') \leq 1/\varepsilon$. Finally, we note that

$$\frac{w(S \setminus \{w_1\})}{\text{vol}(R')} = \frac{w(S) - w_1}{\text{vol}(R) - w_1} < \frac{w(S)}{\text{vol}(R)} = 1 - \varepsilon.$$

Hence, we can slightly shrink R' such that $w(S \setminus \{w_1\}) = (1-\varepsilon) \cdot \text{vol}(R')$, which means we can apply the induction hypothesis to obtain a suitable set of hyperrectangles for the weights in $S \setminus \{w_1\}$ inside R'. Together with the hyperrectangle R_1 for w_1, this gives us a collection of hyperrectangles for the weights in S satisfying all the conditions.

$\square$

We can now prove our final result on partitions with slack.

Theorem 4. *Let $0 < \epsilon < 1/3$, and let $d \geq 2$. Let $\mathcal{T}$ be a properly weighted tree. Then there exists a rectangular partition with ε-slack in $\mathbb{R}^d$ for $\mathcal{T}$ whose aspect ratio is at most $1/\varepsilon$.*

Proof. We construct the rectangular partition recursively, as follows. We start at the root of $\mathcal{T}$, where we set $R(\text{root}(\mathcal{T}))$ to the unit hypercube. Note that the rectangular aspect ratio of a hypercube is 1. Now, given a hyperrectangle $R(\nu)$ of an internal node ν such that the rectangular aspect ratio is at most $1/\varepsilon$, we will construct hyperrectangles for the children of ν of the required aspect ratio. To this end, for any child μ of ν we define

$$w^*(\mu) := (1 - \varepsilon) \cdot w(\mu) \cdot \frac{\text{vol}(R(\nu))}{w(\nu)}.$$

Observe that since $\sum_{\mu} w(\mu) = w(\nu)$, we have

$$\sum_{\mu} w^*(\mu) = (1 - \varepsilon) \cdot \text{vol}(R(\nu)).$$

Hence, we can apply Lemma 6 with $S = \{w^*(\mu) : \mu \text{ is a child of } \nu\}$ to partition $R(\nu)$ into pairwise disjoint hyperrectangles R_μ, each contained in $R(\nu)$ and of aspect ratio at most $1/\varepsilon$. The volume of each $R(\mu)$ will be $w^*(\mu)$, which is in the allowed range. Finally, we recurse on each non-leaf child. After we the recursive process has finished, we have the desired partition.   □

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

Before we describe our algorithm for embedding ultrametrics into $\mathbb{R}^d$, we define α-*hierarchical well separated trees*, or α-HSTs for short, introduced by Bartal [4]. Let $\alpha > 1$ be a parameter. An α-HST is a rooted tree $\mathcal{T}$ with all leaves on the same level, where each node ν has an associated label $l(\nu)$ such that for any node ν with parent μ we have $l(\mu) = \alpha \cdot l(\nu)$. The metric space that corresponds to the HST $\mathcal{T}$ is defined on the leaves of $\mathcal{T}$, and the distance between any two leaves in $\mathcal{T}$ is equal to the label of the their lowest common ancestor.

   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. For any $\alpha > 1$, we can embed M into an α-HST, with distortion α [5]. Given M, we initially compute an embedding of M into a 2-HST $\mathcal{T}$ with distortion 2. Let $M_{\mathcal{T}} = (X, D_{\mathcal{T}})$ be the metric space corresponding to $\mathcal{T}$. Any embedding of $M_{\mathcal{T}}$ into $\mathbb{R}^d$ with distortion c', yields an embedding of M into $\mathbb{R}^d$ with distortion $O(c')$. It therefore suffices to embed of $M_{\mathcal{T}}$ into $\mathbb{R}^d$. With a slight abuse of notation we will from now on denote the leaf of $\mathcal{T}$ corresponding to a point $x \in X$ simply by x.

A lower bound.   To prove the approximation ratio of our embedding algorithm, we need a lower bound on the optimal distortion. We will use the lower bound proved by Bădoiu *et al.* [9], which we describe next.

   Consider an embedding ϕ of $M_{\mathcal{T}}$ into $\mathbb{R}^d$. Assume without loss of generality that ϕ is non-contracting, that is, ϕ does not make any distances smaller. For each node $\nu \in \mathcal{T}$ we define a set $A_\nu \subset \mathbb{R}^d$ as follows. Let $B_d(r)$ denote the ball in $\mathbb{R}^d$ centered at the origin and of radius r. For a leaf ν of $\mathcal{T}$, let A_ν be equal to the ball $B_d(\frac{1}{2})$ translated so that its center is $\phi(\nu)$. For a non-leaf node ν with children $\mu_1, \ldots, \mu_k$, let A_ν be the Minkowski sum of $\bigcup_{i=1}^{k} A_{\mu_i}$ with $B_d(l(\nu))$. By the non-contraction of ϕ it follows that for each pair of nodes ν, ν' that are on the same level of $\mathcal{T}$, the regions A_ν and $A_{\nu'}$ have disjoint interiors—see Figure 9 for an example. Intuitively, the volume A_ν is necessary to embed all the points corresponding to the leaves below ν such that the embedding is non-contracting. This in turn can be used via an isoperimetric argument to obtain a lower bound on the maximum distance (and, hence, distortion) between the images of these leaves.
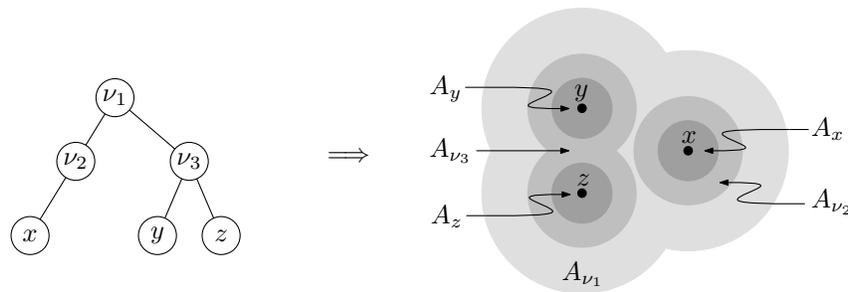
Figure 9: The sets A_ν for a non-contracting embedding of an HST.

We cannot compute A_ν exactly, however, since we do not know the embedding. Hence, following Bădoiu *et al.* [9] we define for each $\nu \in \mathcal{T}$ a value $A^*(\nu)$ that estimates the volume of A_ν. Intuitively, the estimate on A_ν is derived by the Brunn-Minkowski inequality. More precisely, we define $A^*(\nu)$ as follows.

$$A^*(\nu) = \begin{cases} \text{vol}\left(B_d(\tfrac{1}{2})\right) & \text{if } \nu \text{ is a leaf,} \\ \sum_{\mu \text{ is a child of } \nu} \left(A^*(\mu)^{1/d} + \text{vol}\left(B_d\left(\tfrac{l(\nu)}{4}\right)\right)^{1/d}\right)^d & \text{otherwise.} \end{cases}$$

The estimate $A^*(\nu)$ can be used to obtain a lower bound on the distortion, as made precise in the next lemma. For any $V > 0$, let $r_d(V)$ be the radius of a d-dimensional ball with volume V; thus we have

$$r_d(V) = \left(\frac{V \cdot \Gamma(1 + d/2)}{\pi^{d/2}}\right)^{1/d} = \Theta\left(\sqrt{d} \cdot V^{1/d}\right),$$

where $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$.

Lemma 7 ([9], Corollary 1). *Let* OPT *denote the distortion of an optimal embedding of* $M_\mathcal{T}$ *into* $\mathbb{R}^d$. *Then*

$$\text{OPT} \geq \max_{\nu \text{ is internal node of } \mathcal{T}} \frac{r_d(A^*(\nu))}{l(\nu)} - 1.$$

The algorithm. We are now ready to describe the embedding f of $M_\mathcal{T}$ into $\mathbb{R}^d$. The intuition behind our algorithm is as follows. The lower bound given by Lemma 7 implies that an embedding is nearly-optimal if it results in sets A_ν with small aspect ratio. Our approach, however, is essentially reversed. We first compute a hyperrectangular partition of $\mathbb{R}^d$ into hyperrectangles with small aspect ratio. The hyperrectangle computed for the leaves in $\mathcal{T}$ will roughly correspond to the balls around the embedded points $x \in X$, so given the partition we will be able to obtain the embedding by placing $f(x)$ at the center of the hyperrectangle computed for x. The weights we use for the nodes of $\mathcal{T}$ will be the values $A^*(\nu)$, which using Lemma 7 can then be used to bound the distortion.

More precisely, the algorithm works as follows.

1. Compute for each node $\nu \in \mathcal{T}$ the value A_ν^*.

2. Compute a hyperrectangular partition with slack $\varepsilon := \min(\frac{1}{3}, \frac{1}{\log \Delta})$ for $\mathcal{T}$, where A_ν^* is used as the weight of a node ν. Note that $\mathcal{T}$ is not properly weighted: the weight of root($\mathcal{T}$) will be greater than 1 and the weight of an internal node ν is larger than the sum of the weights of its children. However, we can still apply Theorem 4. Indeed, by scaling the weights appropriately we can ensure that the root has weight 1, and the fact that the weight of an internal node ν is larger than the sum of the weights of its children only makes it easier to obtain a small aspect ratio. Thus we can compute a hyperrectangular partition for $\mathcal{T}$ whose rectangular aspect ratio is at most $\log \Delta$.

3. Let $P(\nu)$ denote the hyperrectangle computed for node $\nu \in \mathcal{T}$ in Step 2. We slightly modify the hyperrectangles $P(\nu)$, as follows. Starting from the root of $\mathcal{T}$, we traverse all the nodes of $\mathcal{T}$. When we visit an internal node ν, we shrink all hyperrectangles of the nodes in the subtree rooted at ν by a factor of $1 - 1/\log \Delta$, with the center of the (current) hyperrectangle $P'(\nu)$ of ν being the fixed point in the transformation. Note that $P'(\nu)$ itself is not shrunk. Thus the shrinking step moves the hyperrectangles contained inside $P'(\nu)$ away from its boundary and towards its center, thus preventing points in different subtrees to get too close to each other.

4. Let $P'(\nu)$ denote the hyperrectangle computed in Step 3. Then we define the embedding $f(x)$ of a point $x \in X$ to be the center of the hyperrectangle $P'(x)$.

It remains to bound the distortion of f. We will need the following observation, which bounds the diameter of a hyperrectangle in terms of its volume and aspect ratio.

Observation 2. *Let P be a hyperrectangle in $\mathbb{R}^d$ with aspect ratio at most α. Then, assuming $\alpha \geq \sqrt{d-1}$,*
$$\mathrm{diam}(P) \leq \alpha\sqrt{2} \cdot \mathrm{vol}(P)^{1/d}.$$

From now on we assume that $\log \Delta \geq \sqrt{d-1}$.

Lemma 8. *The expansion of f is $O\left(\frac{1}{\sqrt{d}} \cdot \log \Delta \cdot \mathrm{OPT}\right)$.*

Proof. Consider points $x, y \in X$. Let ν be the lowest common ancestor of x and y in $\mathcal{T}$. We have $D_\mathcal{T}(x,y) = l(\nu)$. Both $f(x)$ and $f(y)$ are contained in $P'(\nu)$, which in turn is contained in $P(\nu)$. Hence, $\mathrm{diam}(P(\nu))$ gives an upper bound on $|xy|$. Note that $\mathrm{asp}_{\mathrm{rect}}(P(\nu)) \leq \log \Delta$ and that $\mathrm{vol}(P(\nu)) \leq A_\nu^*$. (We do not necessarily have $\mathrm{vol}(P(\nu)) = A_\nu^*$ because of the slack in the partition.) Hence,

$$
\begin{aligned}
|xy| &\leq \mathrm{diam}(P(\nu)) \\
&\leq \log \Delta \cdot \sqrt{2} \cdot \mathrm{vol}(P(\nu))^{1/d} \\
&\leq \log \Delta \cdot \sqrt{2} \cdot (A_\nu^*)^{1/d} \\
&= O\left(\log \Delta \cdot r_d(A_\nu^*) \cdot \frac{1}{\sqrt{d}}\right) \\
&= O\left(\log \Delta \cdot (\mathrm{OPT} + 1) \cdot l(\nu) \cdot \frac{1}{\sqrt{d}}\right) \\
&= O\left(\log \Delta \cdot \mathrm{OPT} \cdot D_\mathcal{T}(x,y) \cdot \frac{1}{\sqrt{d}}\right),
\end{aligned}
$$

where the second-to-last transition follows from Lemma 7. This shows that the expansion is $O(\log \Delta \cdot \text{OPT} \cdot \frac{1}{\sqrt{d}})$. □

Lemma 9. *The contraction of f is $O\left(\sqrt{d} \cdot \log^2 \Delta\right)$.*

Proof. Since $\mathcal{T}$ is a 2-HST, we have $\Delta = l(\text{root}(\mathcal{T})) = 2^{\text{height}(\mathcal{T})}$. Hence, $\text{height}(\mathcal{T}) = \log \Delta$. It follows that for each node $\nu \in \mathcal{T}$,

$$\text{vol}(P'(\nu)) = \Omega\left(\left(1 - \frac{1}{\log \Delta}\right)^{\log \Delta} \cdot \text{vol}(P(\nu))\right) = \Omega\left(\text{vol}(P(\nu))\right)$$

$$= \Omega\left(\left(1 - \frac{1}{\log \Delta}\right)^{\log \Delta} \cdot A_\nu^*\right) = \Omega\left(A_\nu^*\right).$$

Consider points $x, y \in X$, and let ν be the lowest common ancestor of x and y in $\mathcal{T}$. We will consider the following two cases for ν:

- *Case 1: ν is the parent of x and y in $\mathcal{T}$.*
  Since the minimum distance in $M_\mathcal{T}$ is 1, it follows that $D_\mathcal{T}(x, y) = 1$. By construction, $f(x)$ is the center of $P'(x)$. Let t be the distance between $f(x)$ and $\partial P'(x)$. Since $\text{asp}_{\text{rect}}(P'(x)) \leq 1/\varepsilon$, we have

$$t \geq \varepsilon \cdot \text{vol}(P'(x))^{1/d} = \Omega\left(\varepsilon \cdot (A_x^*)^{1/d}\right) = \Omega\left(\varepsilon \cdot \frac{1}{\sqrt{d}}\right) = \Omega\left(\frac{1}{\sqrt{d}\log \Delta}\right).$$

  Thus, $|f(x)f(y)| \geq t = \Omega\left(\frac{D(x,y)}{\sqrt{d}\log \Delta}\right)$.

- *Case 2: ν is not the parent of x and y in $\mathcal{T}$.*
  Let μ be the child of ν that lies on the path from ν to x. Let t be the distance between x and $\partial P'(\mu)$. Because of the shrinking performed in Step 3, we know that $t \geq (1/\log \Delta) \cdot (s/2)$, where s is the length of the shortest edge of $P'(\mu)$. Since $\text{asp}_{\text{rect}}(P'(\mu)) \leq 1/\varepsilon$, we have $s \geq \varepsilon^{1-1/d} \cdot \text{vol}(P'(\mu))^{1/d}$. Hence,

$$t \geq \frac{1}{\log \Delta} \cdot \frac{s}{2}$$
$$= \Omega\left(\frac{\varepsilon^{1-1/d}}{\log \Delta} \cdot \text{vol}(P'(\mu))^{1/d}\right)$$
$$= \Omega\left(\frac{\varepsilon^{1-1/d}}{\log \Delta} \cdot (A_\mu^*)^{1/d}\right)$$
$$= \Omega\left(\frac{1}{\log^2 \Delta} \cdot (A_\mu^*)^{1/d}\right)$$
$$= \Omega\left(\frac{1}{\log^2 \Delta} \cdot l(\mu) \cdot \frac{1}{\sqrt{d}}\right)$$
$$= \Omega\left(\frac{D(x,y)}{\sqrt{d}\log^2 \Delta}\right).$$

□

Combining Lemmas 8 and 9 we obtain the main result of the section.

Theorem 5. *For any fixed $d \geq 2$, there exists a polynomial-time, $O\left(\sqrt{d} \cdot \log^3 \Delta\right)$-approximation algorithm for the problem of embedding ultrametrics into $\mathbb{R}^d$ with minimum distortion.*

We remark that the running time of the above algorithm depends on the way the input is given. If the ultrametric is given as a matrix of pairwise distances, then one needs to first compute a tree representation, with the points being the leaves of the tree. For an n-point ultrametric, this task takes at least $\Omega(n^2)$ time. Given such a tree representation, for any fixed dimension d, it is fairly easy to implement our algorithm in roughly quadratic time. It is an interesting open problem to obtain an algorithm with near-linear running time.

As a final comment, note that while the approximation factor is relatively small (for instance, for $\Delta = n^{O(1)}$ and constant d, it becomes $O(\log^3 n)$), the distortion itself can be much larger. For example, embedding the uniform metric on n vertices (which is an ultrametric as well) into $\mathbb{R}^d$ requires distortion $\Omega(n^{1/d})$ for constant d.

Acknowledgments

References

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

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

[3] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. In *37th Annual Symposium on Foundations of Computer Science (Burlington, VT, 1996)*, pages 184–193. IEEE Comput. Soc. Press, Los Alamitos, CA, 1996.

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

[5] 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.

[6] 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.

[7] J. Bourgain. On lipschitz embedding of finite metric spaces into hilbert space. *Isreal Journal of Mathematics*, 52:46–52, 1985.

[8] 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.

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

[10] M. de Berg, B. Speckmann, and V. van der Weele. Treemaps with bounded aspect ratio. *CoRR*, abs/1012.1749, 2010.

[11] M. de Berg, A. F. van der Stappen, J. Vleugels, and M. J. Katz. Realistic input models for geometric algorithms. *Algorithmica*, 34(1):81–97, 2002.

[12] P. Indyk. Tutorial: Algorithmic applications of low-distortion geometric embeddings. *Annual Symposium on Foundations of Computer Science*, 2001.

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

[14] 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.

[15] W.-A. Jungmeister and D. Turo. Adapting treemaps to stock portfolio visualization. Technical Report UMCP-CSD CS-TR-2996, College Park, Maryland 20742, U.S.A., 1992.

[16] J. Matoušek. Bi-Lipschitz embeddings into low-dimensional Euclidean spaces. *Comment. Math. Univ. Carolinae*, 31:589–600, 1990.

[17] J. Matoušek and A. Sidiropoulos. Inapproximability for metric embeddings into $\mathbb{R}^d$. In *49th IEEE Symposium on Foundations of Computer Science*, 2008.

[18] K. Onak and A. Sidiropoulos. Circular partitions with applications to visualization and embeddings. In *Proceedings of the 24nd ACM Symposium on Computational Geometry*, pages 28–37, 2008.

[19] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.

[20] B. Shneiderman. Treemaps for space-constrained visualization of hierarchies. Available at `http://www.cs.umd.edu/hcil/treemap-history/index.shtml`.

[21] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.

[22] B. Shneiderman and M. Wattenberg. Ordered treemap layouts. In *IEEE Symposium on Information Visualization, INFOVIS '01*, pages 73–78, 2001.

[23] D. Turo and B. Johnson. Improving the visualization of hierarchies with treemaps: Design issues and experimentation. *IEEE Visualization*, pages 124–131, 1992.

[24] 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.

[25] S. Wan. Blog treemap visualizer (`http://www.samuelwan.com/information/archives/000159.html`).

[26] M. Weskamp. Newsmap. http://marumushi.com/apps/newsmap/newsmap.cfm.

[27] K. Wetzel. Using Circular Treemaps to visualize disk usage. Available at `http://lip.sourceforge.net/ctreemap.html`.