Appl. Comput. Harmon. Anal. • • • (• • • •) • • •–• • •

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Approximating snowflake metrics by trees

William Leeb[a,b]

[a] Dept. of Mathematics, Yale University, New Haven, CT 06511, United States[1]
[b] PACM, Princeton University, Princeton, NJ 08554, United States

ARTICLE INFO

Article history:
Received 25 January 2015
Received in revised form 4 November 2015
Accepted 7 October 2016
Available online xxxx
Communicated by Charles K. Chui

MSC:
68W05
68W25
05C05
54E35

Keywords:
Tree metric
Partition trees
Tree approximation
Snowflake metric
Dimension
Spaces of homogeneous type
Earth Mover's Distance
EMD

ABSTRACT

Tree metrics are encountered throughout pure and applied mathematics. Their simple structure makes them a convenient choice of metric in many applications from machine learning and computer science. At the same time, there is an elegant theory of harmonic analysis with respect to tree metrics that parallels the classical theory.

A basic question in this field, which is of both theoretical and practical interest, is how to design efficient algorithms for building trees with good metric properties. In particular, given a finite metric space, we seek a random family of dominating tree metrics approximating the underlying metric in expectation. For general metrics, this problem has been solved: on the one hand, there are finite metric spaces that cannot be approximated by trees without incurring a distortion logarithmic in the size of the space, while the tree construction of Fakcharoenphol, Rao, and Talwar (FRT, 2003) shows how to achieve such a logarithmic error for arbitrary metrics.

Since a distortion that grows even logarithmically with the size of the set may be too large for practical use in many settings, one naturally asks if there is a more restricted class of metrics where one can do better. The main result of this paper is that certain random family of trees already studied in the computer science literature, including the FRT trees, can be used to approximate snowflake metrics (metrics raised to a power less than 1) with expected distortion bounded by its doubling dimension and the degree of snowflaking. We also show that without snowflaking, the metric distortion can be bounded by a term logarithmic in the distance being approximated and linear in the dimension.

We also present an optimal algorithm for building a single FRT tree, whose running time is bounded independently of all problem parameters other than the number of points. We conclude by demonstrating our theoretical results on a numerical example, and applying them to the approximation of the Earth Mover's Distance between probability distributions.

E-mail address: wleeb@math.princeton.edu.

[1] Previous address.

http://dx.doi.org/10.1016/j.acha.2016.10.002
1063-5203/© 2016 Published by Elsevier Inc.

1. Introduction

Tree metrics are an especially simple kind of distance function that appear throughout pure and applied mathematics. Informally, tree metrics are derived by breaking the metric space into a tree $\mathcal{T}$ of nested subsets F, called folders, and assigning each folder a diameter $w(F)$. The distance $d_{\mathcal{T}}(x, y)$ between any two points x and y is then the diameter of the smallest folder containing both points.

There is an extensive theory of harmonic analysis for tree metrics that parallels the classical Euclidean theory. This theory allows us to adapt signal-processing type algorithms to data sets of much more varied structure, and has proven useful in a wide array of problems in machine learning [1–3]. Tree metrics' simple structure also yields fast algorithms for metric tasks from computer science, such as nearest neighbor searches, the k-server problem, distributed paging, the vehicle routing problem, and many more [4,5].

Unfortunately, it is rarely the case that the "natural" metric for a given problem in machine learning or computer science will be a tree metric. A basic goal in metric space theory, therefore, is to approximate arbitrary finite metrics by tree metrics. Of course, the extreme simplicity of tree metrics makes it implausible that an arbitrary metric could be well-approximated by a single tree metric. We therefore consider a modified problem, namely finding a probability distribution over tree metrics so that the expected tree distance yields a good approximation, and such that it is computationally feasible to draw a tree from the distribution.

The formal problem, as considered in [6,4,5,7–9] and elsewhere is as follows. Given a finite metric space (X, d), we seek a family of trees $\mathcal{T}$ and corresponding tree metrics $d_{\mathcal{T}}$ that have the following properties:

1. Each tree metric is *dominating*; that is,
$$d(x, y) \le d_{\mathcal{T}}(x, y) \tag{1}$$
for every $\mathcal{T}$ and for all $x, y \in X$.
2. The expected tree distance satisfies
$$\mathbb{E}_{\mathcal{T}}[d_{\mathcal{T}}(x, y)] \le K d(x, y) \tag{2}$$
for some constant $K \ge 1$.

Bartal's paper [4] describes such an explicit distribution over trees, where the constant K is of size $O(\log^2 n)$ where n denotes the number of points in X; this result was later improved to $K = O(\log n \log \log n)$ in [5]. With access to such a distribution over trees, many tasks that depend on the original metric can be performed with randomly drawn tree metrics instead, and then combined to produce an approximation to that task for the original metric. Bartal [4,5] discusses a number of such problems from computer science, while Charikar's paper [10] shows how this method can produce an approximation to the *Earth Mover's Distance*, a powerful metric between probability distributions widely used in machine learning [11–14]. We will go into more detail on this particular application in Section 5.

The question that naturally arises is: how small (that is, how close to 1) can we make the constant K from (2)? The paper of Fakcharoenphol, Rao, and Talwar [9] describes a randomized construction of partition trees whose constant of distortion K is of size $O(\log n)$. As there are metric spaces for which no family of trees can achieve a distortion smaller than $\Omega(\log n)$ [4], this result is optimal in the general case.

If n is large, however, a size $O(\log n)$ distortion can be too big for practical applications. Indeed, in a statistical or machine learning environment, if X is a data set drawn from a population about which we wish to make inferences, it is critical to be able to handle very large values of n, as well-designed statistical procedures perform better with increasing sample size.

In this paper we show that a broad class of metrics can in fact be approximated by trees with constant of distortion bounded independently of n. These metrics, known as *snowflake metrics*, are of the form $d(x, y)^\alpha$

where $0 < \alpha < 1$ and $d(x, y)$ is itself a metric [15]. We will prove approximation guarantees for two different tree constructions, both of which have appeared previously in the literature, namely in [9] and [16].

More precisely, in Section 3, namely Theorem 1, we prove the following result: for $R \geq 0$ and any $0 < \alpha < 1$, the trees defined in [9] and [16] can be used to approximate the snowflake metric $d(x, y)^\alpha$ for distances exceeding R with expected distortion bounded independently of the number of points. Rather, the expected distortion depends on the dimension of X at scale R, a quantity that captures the growth of metric balls exceeding radius R (when R is 0, the approximation guarantee holds for all distances).

The proof of Theorem 1 is very simple; it consists of observing that a divergent series of distances multiplied by probabilities becomes convergent when those distances are raised to a power less than 1. Similar observations are important ingredients in proofs of Assouad theorems on embedding snowflake metrics into Euclidean space [15,17,18]. However, for certain applications, such as the approximation of Earth Mover's Distance between probability distributions, it is more natural to work with embeddings into trees than into Euclidean space. We will discuss such applications in more detail in Section 5.

In Section 4 we give an algorithm for constructing the trees from [9] whose cost is $O(n^2)$, where the constants are universal. In particular, the cost can be bounded independently of other problem parameters, such as the distances $d(x, y)$. The existence of an $O(n^2)$ algorithm is stated in [9], though we have not seen it described anywhere in the literature, and it is unclear whether the algorithm referred to in [9] has cost independent of the metric itself.

In Section 5, we illustrate the results of the paper on numerical examples. As mentioned, we also explore how to apply tree approximations to the approximation of the Earth Mover's Distance between probability distributions.

2. Preliminaries

In this section, we discuss the general notions from metric space theory that we will be using throughout this paper, and state a result (Proposition 1 below) about one of the notions of metric dimension we study in this paper.

2.1. Doubling dimension of a metric space

One natural way of extending the definition of "dimension" to an abstract metric space is to measure the rate of growth of metric balls as the radius doubles. More precisely, if (X, d) is a metric space, μ is a measure on X, and $R \geq 0$ is a non-negative number, we define the *doubling dimension of X with respect to μ at scale R* to be

$$\dim_{\mu,R}(X) = \sup_{x \in X, r > R} \log_2 \left(\frac{\mu(B(x, 2r))}{\mu(B(x, r))} \right) \tag{3}$$

where $B(x, r)$ denotes the closed ball of radius r around the point $x \in X$.

When X is finite and μ is just counting measure, (3) is almost identical to the *Karger–Ruhl (KR) dimension* found in [19]; the only difference is that with KR dimension, the supremum is taken over balls with a minimum volume rather than a minimum radius. Metrics with finite KR dimension are known as *growth restricted* metrics. The KR dimension is often used to quantify the performance of certain algorithms in computer science, such as nearest neighbor searches and metric embeddings. More generally, (3) is also similar to that used in the theory of spaces of homogeneous type [20], as a defining property of these spaces is that $\dim_{\mu,0}(X) < \infty$.

Another definition of dimension is the *doubling dimension* [15]. This is defined as the base 2 logarithm of the number of balls of radius $r/2$ required to cover a ball of radius r; it therefore captures the rate-of-growth of the space without explicitly referring to the measure.

In this paper, we will consider the doubling dimension at scale R, denoted $\lambda_R(X)$, which only looks at balls with radius $r > R$. More formally, for any $x \in X$ and $r > R$, let

$$N(x,r) = \min \left\{ N \in \mathbb{N}_+ : \exists x_1, \dots x_N \text{ s.t. } B(x,r) \subset \bigcup_{i=1}^{N} B(x_i, r/2) \right\} \tag{4}$$

and define $\lambda_R(X)$ by

$$\lambda_R(X) = \sup_{x \in X, r > R} \log_2(N(x,r)). \tag{5}$$

It is a well-known result of metric space theory that a space has finite doubling dimension if and only if it has a doubling measure, and the two dimensions are equivalent (their ratio is bounded above and below by universal constants) [15]. The same is not true for arbitrary measures, such as counting measure. For instance, any space with finite KR dimension has finite doubling dimension, but the converse is false [21]. Both KR dimension and doubling dimension (especially the latter) are widely used to quantify performance guarantees of algorithms in computer science, such as nearest neighbor searches, metric embeddings, network routing, and others; see, for instance, [22–24,19,25–27] for a sampling of such applications.

A nice property of doubling dimension is that it is hereditary: subsets of a metric space can only have smaller doubling dimension than the large space. Unfortunately, there is no corresponding result in general for $\dim_{\mu,R}(X)$. In [19], however, it is shown that if a finite metric space X is formed by randomly sampling from a larger finite space with KR dimension d, say, then X has KR dimension $d+1$ with high probability. In the same spirit, we show the following result (the proof is in Appendix A):

Proposition 1. *Suppose $R \geq 0$ and let M be any metric/measure space with total measure 1 and dimension d at scale $R/2$; that is, for all $x \in M$ and $r > R/2$,*

$$\frac{\mu(B(x,2r))}{\mu(B(x,r))} \leq 2^d \tag{6}$$

where μ is the measure on M. Suppose too that

$$\eta \equiv \inf_{x \in M, r \geq R} \mu(B(x,r)) > 0. \tag{7}$$

Let X be a finite set formed by uniformly sampling n points from M; that is, the probability that a point appears in a subset $S \subset M$ is equal to the measure $\mu(S)$. Equip X with counting measure ν. Then we can bound the probability that $\dim_{\nu,R}(X) \leq d+1$ as follows:

$$\Pr[\dim_{\nu,R}(X) \leq d+1] \geq 1 - n^2 \epsilon^{n-1} \tag{8}$$

for some number $\epsilon \in (0,1)$ depending on d and η. In particular, this probability converges to 1 as $n \to \infty$.

In many applications in machine learning, it is assumed that the dataset in question has been sampled from a compact Riemannian manifold (which satisfies the assumptions of Proposition 1); for instance, see [28,29]. Proposition 1 tells us that when the sampling is uniform, such datasets are likely to have small KR dimension when the dimension of the underlying manifold is small as well, and that the probability of the data having small KR dimension increases with the number of samples.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

2.2. Snowflake metrics

Another basic definition we need is of a *snowflake metric* [15]. A snowflake metric $\rho(x, y)$ satisfies the property that $\rho(x, y)^p$ is also a metric for some $p > 1$. Put another way, if we start with any metric $d(x, y)$, the new metric $\rho(x, y) = d(x, y)^\alpha$ is a snowflake metric whenever $0 < \alpha < 1$. We will refer to $d(x, y)^\alpha$ as the *α-snowflake of* $d(x, y)$, or, if α is clear from the context, just the *snowflake of* $d(x, y)$.

It is a subject of considerable interest in mathematics to determine the effects of replacing a metric $d(x, y)$ with its snowflake $d(x, y)^\alpha$. For example, the resulting spaces do not have rectifiable curves, and have larger Hausdorff dimension than the original space [30].

In certain other respects, however, the snowflake metric $d(x, y)^\alpha$ can be better-behaved than the original metric $d(x, y)$. For example, Assouad's Theorem states that any snowflake metric space with finite doubling dimension can be embedded into a finite-dimensional Euclidean space [17,30,15,18]. Such embeddings do not exist for arbitrary metric spaces; the Heisenberg group is a counterexample [31]. Another feature of snowflake metrics is found in classical harmonic analysis. Letting $|x - y|$ denote the Euclidean distance, spaces of functions that are Lipschitz with respect to the snowflake metric $|x - y|^\alpha$ – the Hölder functions – and their dual spaces can be easily characterized by wavelet norms when $0 < \alpha < 1$, but not when $\alpha = 1$ [32].

2.3. Tree metrics

We introduce the definition of *tree metric* we will be using in this paper. A *partition tree* $\mathcal{T}$ on a set X is a collection of subsets $F \subset X$, which we will call *folders*, with the following properties:

1. The set X itself is in $\mathcal{T}$;
2. For any two folders F and F' in $\mathcal{T}$, either $F \subset F'$, $F' \subset F$, or F and F' are disjoint.

To have the partition tree $\mathcal{T}$ induce a metric on X, on each folder $F \in \mathcal{T}$ we place a weight $w(F)$. We require that if $F \subsetneq F'$, then $w(F) < w(F')$; and that every singleton folder has weight zero, or $w(\{x\}) = 0$ for all $x \in X$. We then define the tree distance $d_\mathcal{T}(x, y)$ between distinct points x and y to be the weight of the smallest folder containing both x and y. It is easy to see that this is a distance; in fact, it is an *ultrametric*, meaning

$$d_\mathcal{T}(x, z) \le \max\{d_\mathcal{T}(x, y), d_\mathcal{T}(y, z)\}. \tag{9}$$

For all the tree metrics we consider in this paper, there will be a constant $0 < A < 1$ such that whenever $F \subsetneq F'$,

$$w(F) \le Aw(F'). \tag{10}$$

That is, the folder weights decay geometrically.

As we discussed in the introduction, trees and tree metrics are of considerable interest throughout applied mathematics. Metric tasks in computer science tend to be very easy when the underlying metric is a tree metric; many of these are explained in [4,5,10]. Furthermore, there is a theory of harmonic analysis with respect to tree metrics that adapts classical signal-processing algorithms to problems in machine learning; see [1–3] for work in this direction.

3. Tree approximations of snowflake metrics

In this section, we consider two randomized constructions of trees on a finite metric space (X, d) that have appeared previously in the literature, and assess the approximation guarantees for the metric $d(x, y)^\alpha$.

It is not hard to see that for any tree approximation, the maximum ratio for the α-snowflakes can never be worse than the αth power of the maximum ratio for approximating the original metric. This observation follows immediately from Jensen's inequality, as

$$\frac{\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^\alpha]}{\mathrm{diam}(S)^\alpha} = \mathbb{E}_{\mathcal{T}}\left[\left(\frac{\mathrm{diam}_{\mathcal{T}}(S)}{\mathrm{diam}(S)}\right)^\alpha\right] \le \left(\frac{\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)]}{\mathrm{diam}(S)}\right)^\alpha, \tag{11}$$

where we denote the diameter of a set $S \subset X$ under the distance $d_{\mathcal{T}}(x, y)$ induced by the tree $\mathcal{T}$ and the weights $w(F)$ as $\mathrm{diam}_{\mathcal{T}}(S)$, and denote the diameter of S under the original metric $d(x, y)$ as $\mathrm{diam}(S)$.

In the remainder of the paper, we will always assume without loss of generality that the diameter of X (that is, the maximum distance between any two points) is 1.

3.1. Partitions at a single scale

This section examines two related methods of constructing trees and tree metrics. Both follow the same basic structure, which we outline here.

For every integer $l \ge 0$, we construct an initial partition $\mathcal{P}_l$ of X; every set in $\mathcal{P}_l$ has diameter less than or equal to 2^{-l+2}. These partitions will not necessarily be nested (that is, $\mathcal{P}_{l+1}$ need not be a refinement of $\mathcal{P}_l$). However, we convert these partitions into a partition tree by the following method. The topmost folder (level 0) consists of the single folder X itself. We form the lth level of the tree as follows. Given a folder F at level $l - 1$, the child folders of F are formed by grouping together those points in F that are assigned to the same folder in $\mathcal{P}_l$.

We state two key results about the initial, unrefined partitions $\mathcal{P}_l$ that follow from results previously published in the literature. Fix $R \ge 0$, a subset $Y \subset X$, and suppose that every point in X is within $\Delta \ge 4R$ of Y. For instance, Y could be all of X, or a maximal Δ-net in X. Suppose μ is a probability measure on Y that assigns every point positive mass, and we permute Y according to μ. Assign every point in X to the first point in Y it is within β of, where $\beta \sim$ Uniform$(\Delta, 2\Delta)$. Since $\beta \ge \Delta$, every point in X gets assigned somewhere.

Proposition 2. *Suppose $Y = X$. Then if $S \subset X$ is any subset, the probability that S gets split (i.e. that two points in S are assigned to different points in X) is no more than*

$$\Pr[S \textit{ split}] \le C\frac{\mathrm{diam}(S)}{\Delta}\,\mathrm{dim}_{\mu,R}(X) \tag{12}$$

where C is a universal constant.

Proof. Since $S \subset B(x, \mathrm{diam}(S))$, where $x \in S$ is arbitrary, the result then follows immediately from the proof of Theorem 3.17 in [33]. More precisely, it is shown that

$$\Pr[B(x, t) \text{ is cut}] \le C\frac{t}{\Delta}\log\left(\frac{\mu(B(x, 2\Delta))}{\mu(B(x, \Delta))}\right). \tag{13}$$

Taking $t = \mathrm{diam}(S)$ and noting that $S \subset B(x, t)$ yields the desired result. □

Proposition 3. *Suppose Y is a Δ-net of X, and the measure μ is counting measure; that is, the permutation on Y is drawn uniformly at random. Then if $S \subset X$ is any subset, the probability that S gets split (i.e. that two points in S are assigned to different points in Y) is no more than*

$$\Pr[S\ \mathit{split}] \leq C\frac{\operatorname{diam}(S)}{\Delta}\lambda_R(X), \tag{14}$$

where C is a universal constant.

Proof. As in the proof of Proposition 2, using the fact that $S \subset B(x, \operatorname{diam}(S))$ for $x \in S$, the result is then immediate from the proof of Theorem 3.2 in [16]. □

3.2. μ-FRT trees

We first consider the random trees of [9], with the modification described in [33]. There are two random objects that define each tree: a random permutation π of the points in X, and a random number $\beta \sim \mathrm{Uniform}(1, 2)$. We require that π and β be drawn independently of each other. The permutation π is drawn according to a pre-specified measure μ on X.

For $l \geq 0$, we define

$$\beta_l = 2^{-l}\beta. \tag{15}$$

We define a tree on X by specifying the initial, unrefined partitions $\mathcal{P}_l$, and then applying the general procedure described in Section 3.1 for converting these partitions into a partition tree. $\mathcal{P}_l$ is defined as follows: for each $x \in X$, let x_l^* be the first point in X (according to the permutation π) such that $d(x, x_l^*) \leq \beta_l$. Then $\mathcal{P}_l$ partitions X into points all assigned to the same x_l^*.

Let $\mathcal{T}$ be the tree induced by these partitions, as described in Section 3.1. That is, the only level 0 folder in the tree is the entire set X, and its center is the first point on the list (since the diameter of X is 1, and $\beta_0 \geq 1$). For each folder F at level l, the subfolders of F (at level $l + 1$) are formed by grouping together the points in F that were assigned to the same point at level $l + 1$.

The weight we place on a folder at level l is

$$w(F) = 2^{-l+2}. \tag{16}$$

Note that $w(F)$ is an upper bound on the diameter of F.

We will refer to the trees defined in this matter as μ-FRT trees, since the construction is found in [9] (the use of a general measure μ in place of counting measure appears in [33], however).

3.3. R-nets trees

There is another natural means of forming the partitions $\mathcal{P}_l$. Namely, at scale l we suppose that we are given a maximal 2^{-l}-net Y_l of X; that is, Y_l is a collection of points that are at distance 2^{-l} from each other, and every point in X is within 2^{-l} of some point in Y_l. We then draw a permutation π_l of Y_l uniformly at random, and independently draw a number $\beta \sim \mathrm{Uniform}(1, 2)$. Again, for $l \geq 0$ we define

$$\beta_l = 2^{-l}\beta. \tag{17}$$

We define a tree on X by specifying the initial, unrefined partitions $\mathcal{P}_l$, and then applying the general procedure described in Section 3.1 to convert the $\mathcal{P}_l$ into a partition tree. $\mathcal{P}_l$ is defined as follows: for each

$x \in X$, let x_l^* be the first point in Y_l (according to the permutation π_l) such that $d(x, x_l^*) \leq \beta_l$. Then $\mathcal{P}_l$ partitions X into points all assigned to the same x_l^*.

We will consider a generalization of this procedure, which only makes use of those partitions $\mathcal{P}_l$ with $2^{-l} \geq R$. Once again, the weight we place on a folder at level l is

$$w(F) = 2^{-l+2}. \tag{18}$$

Note that $w(F)$ is an upper bound on the diameter of F. We will refer to the trees so constructed as R-nets trees.

3.4. Approximation guarantees

We prove bounds on the expected distortion of the tree metric raised to the power α, $0 < \alpha \leq 1$, for both FRT trees and nets trees. It is immediate that $\mathrm{diam}_{\mathcal{T}}(S) \geq \mathrm{diam}(S)$, and consequently the same inequality holds for the snowflake metrics: $\mathrm{diam}_{\mathcal{T}}(S)^\alpha \geq \mathrm{diam}(S)^\alpha$.

We will prove that in expectation, the reverse inequality is true, up to a certain distortion which we bound. Specifically, we have the following theorem:

Theorem 1. *Let*

$$D(X,R) = \begin{cases} \mathrm{dim}_{\mu,R}(X) & \text{if } \mu\text{-FRT trees are used} \\ \lambda_R(X) & \text{if } R\text{-nets trees are used} \end{cases} \tag{19}$$

Both the μ-FRT and the R-nets tree constructions produce family of trees $\mathcal{T}$ with the following properties (C denotes a universal constant):

1. *For any $0 < \alpha < 1$, any $0 \leq R \leq 1$ and any subset $S \subset X$ with diameter $\mathrm{diam}(S) \geq R$,*

$$\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^\alpha] \leq C \frac{D(X,R)}{1-\alpha} \mathrm{diam}(S)^\alpha \tag{20}$$

and

$$\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)] \leq C \left[1 + D(X,R) \log_2 \left(\frac{1}{\mathrm{diam}(S)} \right) \right] \mathrm{diam}(S). \tag{21}$$

2. *For any $0 < \alpha < 1$, any $0 \leq R \leq 1$ and any subset $S \subset X$ with diameter $\mathrm{diam}(S) < R$,*

$$\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^\alpha] \leq C \left(\frac{D(X,R)}{1-\alpha} + \frac{1}{\alpha} \right) R^\alpha \tag{22}$$

and

$$\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)] \leq C \left[1 + D(X,R) \log_2 \left(\frac{1}{R} \right) \right] R. \tag{23}$$

Proof. Define the integer $l^* \geq 0$ by

$$2^{-l^*-1} < \mathrm{diam}(S) \leq 2^{-l^*} \tag{24}$$

and the integer $m^* \geq 0$ by

$$2^{-m^*-1} < R \leq 2^{-m^*}. \tag{25}$$

Observe that if all points in S are in the same folder at level l, the diameter of S must be less than the diameter of their shared folder, which implies that $\mathrm{diam}(S) \le 2^{-l+2}$. Therefore,

$$l \le l^* + 2. \tag{26}$$

Let G_l be the event that all points in S are assigned to the same point at level l. Then we have shown

$$\begin{aligned}\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^\alpha] &\le \sum_{l=0}^{\infty} 2^{-(l-2)\alpha}\Pr[G_l \setminus G_{l+1}] \\ &\le \sum_{l=0}^{l^*+2} 2^{-(l-2)\alpha}\Pr[G_{l+1}^c].\end{aligned} \tag{27}$$

Propositions 2 and 3 provide upper bounds on $\Pr[G_{l+1}^c]$ that will give us the desired result. Suppose $l \le \min\{l^*, m^*\} - 2$. Then

$$\Pr[G_{l+1}^c] \le C2^l \mathrm{diam}(S)(1 + D(X,R)). \tag{28}$$

To prove (20) and (21), suppose $\mathrm{diam}(S) \ge R$. Then $l^* \le m^*$, so (28) is applicable when $l \le l^* - 2$. When $\alpha < 1$, we therefore have

$$\begin{aligned}\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^\alpha] &\le \sum_{l=0}^{l^*+2} 2^{-(l-2)\alpha}\Pr[G_{l+1}^c] = \left\{ \sum_{l=0}^{l^*-2} + \sum_{l=l^*-1}^{l^*+2} \right\} 2^{-(l-2)\alpha}\Pr[G_{l+1}^c] \\ &\le C\mathrm{diam}(S)D(X,R)\sum_{l=0}^{l^*-2} 2^{l(1-\alpha)} + \sum_{l=l^*-1}^{l^*+2} 2^{-l\alpha} \\ &\le C\left(2^{-l^*}D(X,R)\frac{1}{2^{1-\alpha}-1}2^{(l^*-1)(1-\alpha)} + 2^{-l^*\alpha}\right) \\ &\le C\left(1 + \frac{D(X,R)}{1-\alpha}\right)\mathrm{diam}(S)^\alpha,\end{aligned} \tag{29}$$

which is the inequality (20).

For the case when $\alpha = 1$, we have

$$\begin{aligned}\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)] &\le \sum_{l=0}^{l^*+2} 2^{-(l-2)}\Pr[G_{l+1}^c] = \left\{ \sum_{l=0}^{l^*-2} + \sum_{l=l^*-1}^{l^*+2} \right\} 2^{-(l-2)}\Pr[G_{l+1}^c] \\ &\le \sum_{l=0}^{l^*-2} 2^{-l+2}2^l\mathrm{diam}(S)D(X,R) + \sum_{l=l^*-1}^{l^*+2} 2^{-(l-2)} \\ &\le C(\mathrm{diam}(S)D(X,R)l^* + 2^{-l^*}) \\ &\le C\mathrm{diam}(S)\left[D(X,R)\log_2\left(\frac{1}{\mathrm{diam}(S)}\right) + 1\right].\end{aligned} \tag{30}$$

This is the inequality (21).

To prove (22) and (23), suppose $\mathrm{diam}(S) < R$. Then $m^* \le l^*$, so (28) is applicable when $l \le m^* - 2$. We therefore have

$$
\begin{aligned}
\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)^{\alpha}] \le \sum_{l=0}^{l^*+2} 2^{-(l-2)\alpha}\Pr[G_{l+1}^c] &= \Bigg\{ \sum_{l=0}^{m^*-2} + \sum_{l=m^*-1}^{l^*+2} \Bigg\} 2^{-(l-2)\alpha}\Pr[G_{l+1}^c] \\
&\le C\mathrm{diam}(S)D(X,R)4^{\alpha}\sum_{l=0}^{m^*-2} 2^{l(1-\alpha)} + \sum_{l=m^*-3}^{\infty} 2^{-l\alpha} \\
&\le C2^{-m^*}D(X,R)\frac{1}{2^{1-\alpha}-1}2^{(m^*-1)(1-\alpha)} + \frac{2^{-(m^*-3)\alpha}}{1-2^{-\alpha}} \\
&\le C\left(\frac{D(X,R)}{2^{1-\alpha}-1} + \frac{1}{1-2^{-\alpha}}\right)2^{-m^*\alpha} \\
&\le C\left(\frac{D(X,R)}{1-\alpha} + \frac{1}{\alpha}\right)R^{\alpha}
\end{aligned}
\tag{31}
$$

which is inequality (22).

Finally, to prove (23) we have

$$
\begin{aligned}
\mathbb{E}_{\mathcal{T}}[\mathrm{diam}_{\mathcal{T}}(S)] \le \sum_{l=0}^{l^*+2} 2^{-(l-2)}\Pr[G_{l+1}^c] &= \Bigg\{ \sum_{l=0}^{m^*-2} + \sum_{l=m^*-1}^{l^*+2} \Bigg\} 2^{-(l-2)}\Pr[G_{l+1}^c] \\
&\le C\sum_{l=0}^{m^*-2} 2^{-l+2}2^{l}\mathrm{diam}(S)D(X,R) + \sum_{l=m^*-1}^{\infty} 2^{-(l-2)} \\
&\le C(\mathrm{diam}(S)D(X,R)m^* + 2^{-m^*}) \\
&\le C\left[1 + D(X,R)\log_2\left(\frac{1}{R}\right)\right]R.
\end{aligned}
\tag{32}
$$

This completes the proof. □

4. Algorithm for constructing a single μ-FRT tree

In this section we describe an explicit algorithm for constructing a single μ-FRT tree $\mathcal{T}$, given the permutation π and the parameter β. We will show that the algorithm has cost $O(n^2)$, which is linear in the problem size. In [9], the authors state the existence of such an algorithm, though we have not seen it described in the literature. Furthermore, the cost of the algorithm we present does not depend on other problem parameters, such as the distances themselves.

To elaborate on the last point, a naïve algorithm for constructing a μ-FRT tree may repeat the same folder multiple times at different levels. This will occur when all the points in a folder at level l are assigned to the same point at level $l+1$. Of course, the tree distance in this case will only be determined by the copy of this folder at the smallest level, so there is no need to include the redundant copies in the tree.

Furthermore, we note that any algorithm whose running time is to be controlled solely in terms of the size of X must avoid forming redundant folders. To see this, consider a metric space with three points, $X = \{x, y, z\}$. Suppose $d(x, y) = \epsilon$, $d(x, z) = 1$, and $d(y, z) = 1 - \epsilon$ for some $\epsilon < 1/2$. Suppose too that the permutation π places x first, y second, and z third. Then it is easy to see that if $0 < l < \log_2(\beta/\epsilon)$, the level l partition consists of the two folders $F_1 = \{x, y\}$ and $F_2 = \{z\}$; in particular, there are at least $\log_2(\beta/\epsilon)$ many levels before the tree splits into singletons and the construction terminates. Consequently, if an algorithm performs operations level-wise, the running time on this example will grow like $\log_2(1/\epsilon)$, even though $n = 3$. Therefore, an algorithm whose cost depends only on n must automatically skip over redundant folders.

We list the points in the order given by π as $x_1, \dots, x_n$. The following lemma will be useful.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

Lemma 1. *Suppose x has been assigned to x_{k_0} at level l_0, and to x_{k_1} at level l_1. Then if $l_1 > l_0$, it must be that $k_1 \geq k_0$.*

Proof. Suppose $k_1 < k_0$, i.e. x_{k_1} occurs before x_{k_0} on the list. Since x is assumed to be assigned to x_{k_1} at level l_1, therefore $d(x, x_{k_1}) \leq \beta_{l_1} < \beta_{l_0}$. But then, since x_{k_1} precedes x_{k_0}, x would have been assigned to x_{k_1} at level l_0; this is a contradiction. □

In other words, we never need to backtrack through the list when looking for the next point to which x is assigned.

Given any points x and y in X define

$$l(x, y) = \lfloor \log_2(\beta) - \log_2(d(x, y)) \rfloor. \tag{33}$$

Lemma 2. *Suppose that x has been assigned to x_k at level l. Then $l \leq l(x, x_k)$, and x will be assigned to x_k at all levels l' such that $l \leq l' \leq l(x, x_k)$.*

Proof. By definition, $l(x, x_k)$ is the largest integer such that $d(x, x_k) \leq 2^{-l(x,y)}\beta$. Since $d(x, x_k) \leq \beta_l = 2^{-l}\beta$, we must have $l \leq l(x, x_k)$. Now suppose that x gets assigned to x_j at level l', $l \leq l' \leq l(x, x_k)$. By Lemma 1, $j \geq k$, i.e. x_j does not occur before x_k in the list. On the other hand, $d(x, x_k) \leq 2^{-l(x,y)}\beta \leq 2^{-l'}\beta = \beta_{l'}$; so x will not be assigned to any point occurring after x_k at level l'. Consequently, x is assigned to x_k at level l'. □

In other words, if x is ever assigned to a point y, then $l(x, y)$ is the last level at which x is assigned to y.

We introduce some terminology. For every folder F on the tree, we will refer to:

- The *center* of F. This is the point x_k that all points in F were assigned to when they became members of F.
- The *level* of F. This is the minimum of $l(x, x_k)$, $x \in F$, where x_k is the center of F. If l' denotes the level of F, then $2^{-l'+2}$ is an upper bound for F's diameter.

Observe that if a folder has center x_k and level l', then by Lemma 2, $l' + 1$ is the first level at which the folder F can be split into subfolders. Consequently, when we are splitting F into its children we never need to consider any subfolders at levels less than l', since they will all be equal to F. Also, if $x \in F$ and $l(x, x_k) = l'$, by Lemma 1 the point x_j to which x is assigned at level $l' + 1$, is the first point on the list $x_{k+1}, x_{k+2}, \dots$ such that $l(x, x_j) > l'$.

These observations yield the following algorithm for constructing the tree. Initialize the tree with the single folder X, with center point x_1, and compute its level. Recursively build folders as follows. Take any folder F whose children have not yet been added to the tree. Let x_k be its center and l' its level. Take those points $x \in F$ with $l(x, x_k) > l'$, if any exist. These points will remain assigned to x_k at level $l' + 1$. So one of the children of F will consist of all points with $l(x, x_k) > l'$, if there are any.

The points with $l(x, x_k) = l'$ are no longer assigned to x_k at level $l' + 1$. To find where they go, for each such point x search through $x_{k+1}, x_{k+2}, \dots$ until the first x_j, $j > k$, is encountered with $l(x, x_j) > l(x, x_k)$. This x_j is the next point to which x is assigned. Therefore, the remaining children of F are formed by grouping together those points that have been advanced to the same point in this manner.

Of course, it could happen that the numbers $l(x, x_k)$ are equal for all $x \in F$, and that all $x \in F$ get advanced to the same point x_j after x_k. In this case, we can keep the identity of F intact, update its center to x_j, find its new level, and repeat the process.

We give a summary of the algorithm:

Algorithm for building $\mathcal{T}$

```
I. Initialize the tree with folder $X$ and center $x_1$
II. Take any non-singleton folder $F$ with center $x_k$ and level $l'$ whose children are not on the tree
    1. If possible, form a child $F_0$ of $F$ consisting of points with $l(x, x_k) > l'$
    2. Advance each remaining $x$ to the first $x_j$, $j > k$, with $l(x, x_j) > l'$
    3. There are two cases:
        i. If $F_0 = \emptyset$ and all points in $F$ advanced to the same point $x_j$, simply make $x_j$ the new center of $F$ and update $F$'s level to be $l' = \min_{x \in F} l(x, x_j)$
        ii. Otherwise, break $F \setminus F_0$ into children of $F$ by grouping the points that advanced to the same $x_j$, and let $l'$ be the level of $F$
    Repeat step II until the children of every folder have been added to the tree.
```

We now analyze the cost of this algorithm. Observe that every time a folder is processed, the operations fall into two categories. First, there is the cost of advancing each point $x \in F$ to the next point to which it is assigned. However, once a point x is advanced to x_k, it is never necessary, when considering x, to look at any points preceding x_k in the list, by Lemma 1; so the most that all such advances can cost over the entire algorithm is $O(n^2)$, since each point x sweeps over all the points in X exactly once.

Second, there are those operations whose costs are directly proportional to the number of points in F, such as the cost of computing $l(x, x_k)$ for each x, where x_k is the center of F. We will break these costs into two cases. The first is when the folder F ends up being broken apart into subfolders. Since this only happens once per folder, the total cost of all such operations can be bounded above by a constant times

$$\sum_{F \in \mathcal{T}} |F| \le \sum_{F \in \mathcal{T}} |X| = O(n^2) \tag{34}$$

since there are at most $2n - 1$ folders in the tree.

The second case is when F does not get broken into subfolders. This can only happen when every point in F is advanced to the same point (so the center of F changes, but F is not broken apart). This does not pose any additional costs, however, since we have already counted these costs when we computed the cost of all advances.

The total cost of the algorithm, therefore, is $O(n^2)$.

5. Numerical results and EMD

We illustrate the results of this paper on examples. In Section 5.1, we examine the performance of the tree approximations on a synthetic low-dimensional dataset of a perturbed grid of points in the plane. In particular, we demonstrate the quality of the approximations for varying degrees of snowflaking (different values of α) and different numbers of trees. In Section 5.2, we show how to use the trees we build to approximate the Earth Mover's Distance between digits in the USPS dataset.

In the experiments that follow, we measure the distortion of one metric $d_1(x, y)$ by another metric $d_2(x, y)$ on a space X as follows:

$$\operatorname{dist}(d_1, d_2) = \left(\sup_{x \ne y} \frac{d_1(x, y)}{d_2(x, y)} \right) \times \left(\inf_{x \ne y} \frac{d_1(x, y)}{d_2(x, y)} \right). \tag{35}$$

This quantity is symmetric in d_1 and d_2, and invariant to rescalings of either metric.

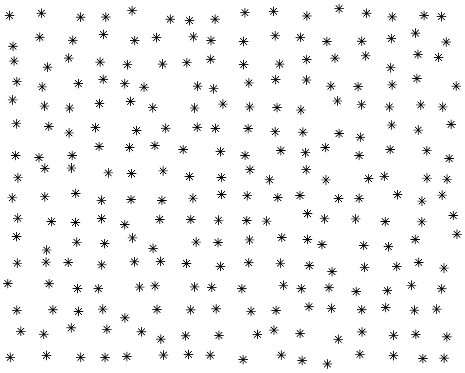

Fig. 1. A perturbed grid.

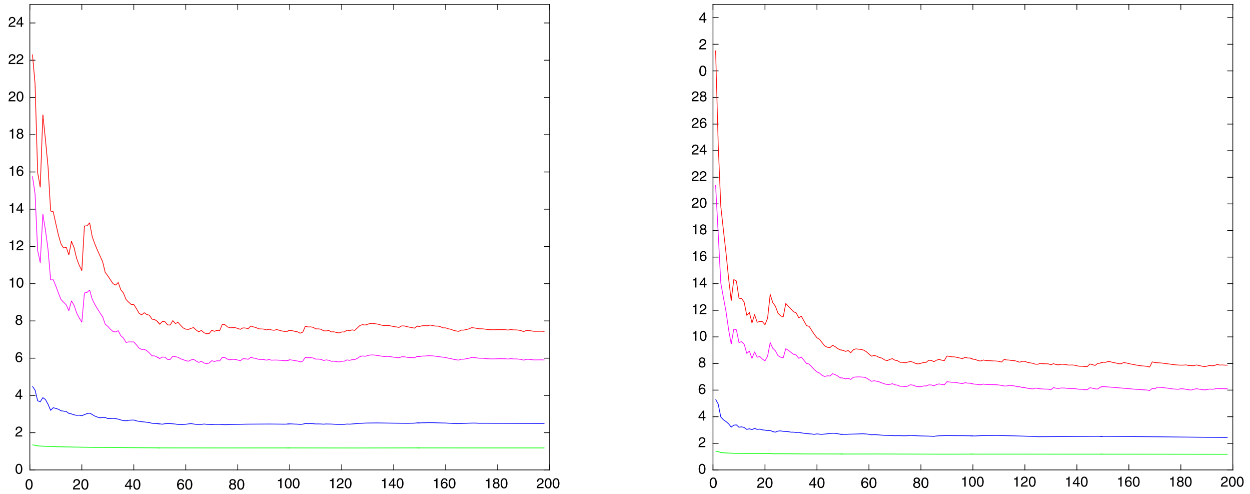

Fig. 2. Distortion for the perturbed grid, plotted as a function of the number of trees being averaged. Left: nets trees. Right: FRT trees. The values of α are 1 (in red, top), .9 (in magenta), .5 (in blue) and .1 (in green, bottom). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.1. The perturbed grid

As an example of a low-dimensional metric space we take a 16-by-16 grid of points in the plane, where each point has been perturbed by a small Gaussian. A realization of such a space is shown in Fig. 1. The space is endowed with the standard Euclidean distance. As the points are almost equispaced, this space is indeed low-dimensional. In the experiments that follow, we show average results over five random draws of this grid.

Fig. 2 shows the distortions incurred by approximating the original metric and its α-snowflakes for $\alpha = .1, .5$, and .9, as a function of the number of tree metrics being averaged, ranging from 1 to 200. As expected, the distortion becomes smaller as α shrinks.

In Fig. 3, we show the distortion incurred by the 1/2-snowflake metric alongside the square root of the distortion of the original metric. Inequality (11) guarantees that $\mathbb{E}_{\mathcal{T}}[d_{\mathcal{T}}(x,y)^\alpha]$ is closer to $d(x,y)^\alpha$ than is $(\mathbb{E}_{\mathcal{T}}[d_{\mathcal{T}}(x,y)])^\alpha$, although it does not guarantee that the distortion will be smaller (as measured by (35)). Fig. 3 suggests that the average snowflaked tree metric does indeed give a better approximation to $d(x,y)^\alpha$ than the snowflake of the average.

Finally, in Fig. 4, we plot the distortion of approximation of the original metrics at each point against the true distance between the points. Theorem 1, in particular (21) and (23), predicts that the distortion of approximating the distance $d(x,y)$ will grow like $\log(1/d(x,y))$ as $d(x,y)$ goes to zero; the plot illustrates this claim.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

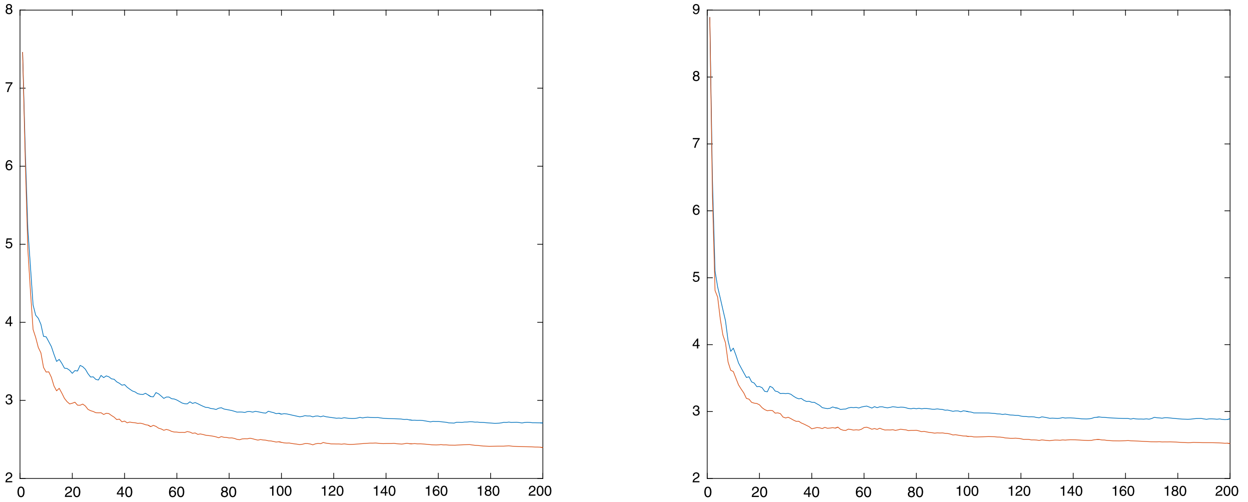

Fig. 3. The distortions of approximation for $\alpha = .5$ (red, bottom curve) versus the square root of the distortion for $\alpha = 1$ (blue, top curve). Left: nets trees. Right: FRT trees. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

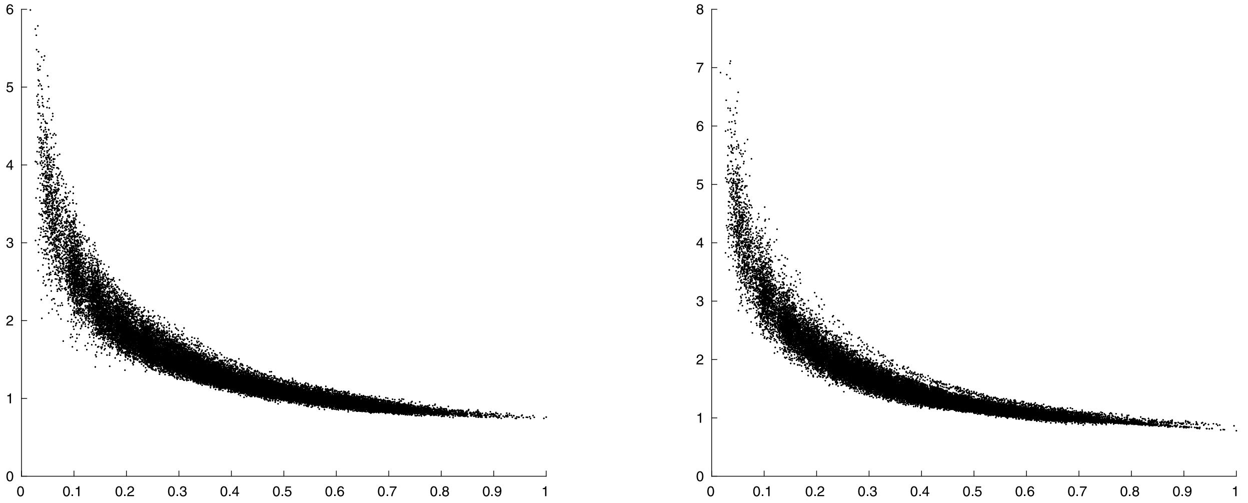

Fig. 4. The ratio as a function of the distance, using 50 trees. Left: nets trees. Right: FRT trees.

5.2. *Earth Mover's Distance*

We apply the tree approximation to the approximation of the Earth Mover's Distance between probability distributions, which we define now. If p and q are probability distributions over a (finite) metric space X equipped with metric $\rho(x, y)$ then the Earth Mover's Distance between p and q is defined as:

$$\text{EMD}(p, q) = \min_{\Pi} \sum_{x \in X} \sum_{y \in X} \rho(x, y)\Pi(x, y) \tag{36}$$

where the minimum is taken over all probability measures on $X \times X$ with

$$\sum_{x \in X} \Pi(x, y) = p(y), \quad \text{for all } y \in X \tag{37}$$

$$\sum_{y \in X} \Pi(x, y) = q(x), \quad \text{for all } x \in X. \tag{38}$$

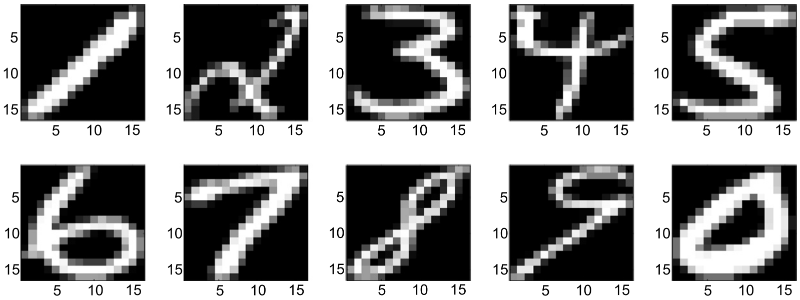

Fig. 5. Samples from the USPS dataset.

Any probability measure Π satisfying (37) and (38) is called a *transport* between p and q; that is, if we interpret $\Pi(x, y)$ as the amount of mass we move from location x to location y, (37) and (38) imply that the transport rearranges the distribution of mass described by q to obtain the distribution described by p. If $\rho(x, y)$ is the cost per unit mass of moving mass from x to y, then $\mathrm{EMD}(p, q)$ is the cheapest way of turning q into p. The distance $\rho(x, y)$ used to measure the cost of moving mass is referred to as the *ground distance*.

EMD has many desirable properties, and is used in a variety of applications in machine learning [11–13]. However, a major drawback is that solving the linear programming problem described by (36), (37) and (38) can be costly, typically scaling supercubically in the size of X [34,35]. Consequently, many approximations to EMD have been proposed.

In [10], a formula is presented for the computation of EMD when the underlying metric is a tree metric, and a proof is given that if a family of dominating tree metrics approximates a given metric with distortion K, then the average EMD over trees approximates the true EMD with distortion no more than K. More precisely, for a tree $\mathcal{T}$ over X, with corresponding tree metric $d_{\mathcal{T}}(x, y)$ induced by weights $w(F)$ on folders $F \in \mathcal{T}$, we let $\mathrm{EMD}_{\mathcal{T}}(p, q)$ denote the Earth Mover's Distance between p and q with respect to the ground distance $d_{\mathcal{T}}(x, y)$; then

$$\mathrm{EMD}_{\mathcal{T}}(p, q) = \sum_{F \in \mathcal{T}, F \neq X} \frac{1}{2}(w(F') - w(F))|p(F) - q(F)| \tag{39}$$

where F' denotes the parent folder of the folder F, and $p(F) = \sum_{x \in F} p(x)$, and similarly for $q(F)$. Since the sums over folders can be computed recursively, expression (39) can be evaluated in $O(n)$ operations. In fact, [10] proves this formula for an even more general class of tree metrics than the ones considered in this paper; see also [36] for another proof of this same general result.

It is also shown in [10,36] that for a family of dominating tree metrics $d_{\mathcal{T}}(x, y)$ that approximate a distance $d(x, y)$ in the sense of (1) and (2), the average $\mathrm{EMD}_{\mathcal{T}}(p, q)$ approximates $\mathrm{EMD}(p, q)$ (using the original distance $d(x, y)$ as the ground distance) in the sense that

$$\mathrm{EMD}(p, q) \leq \mathrm{EMD}_{\mathcal{T}}(p, q) \tag{40}$$

for every tree $\mathcal{T}$, and

$$\mathbb{E}_{\mathcal{T}}[\mathrm{EMD}_{\mathcal{T}}(p, q)] \leq K \mathrm{EMD}(p, q) \tag{41}$$

where K is the same constant of distortion of the ground distance from (2).

We test this result on snowflake ground distances by comparing handwritten digits from the USPS dataset. This dataset consists of 11000 digits in total, 1100 from each of the ten classes 0 through 9. Each image is 16-by-16 pixels large. Sample images from this dataset are shown in Fig. 5. We normalize each image to sum to 1, so we can view them as probability distributions over the 16-by-16 grid, and equip the grid with Euclidean ground distance.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

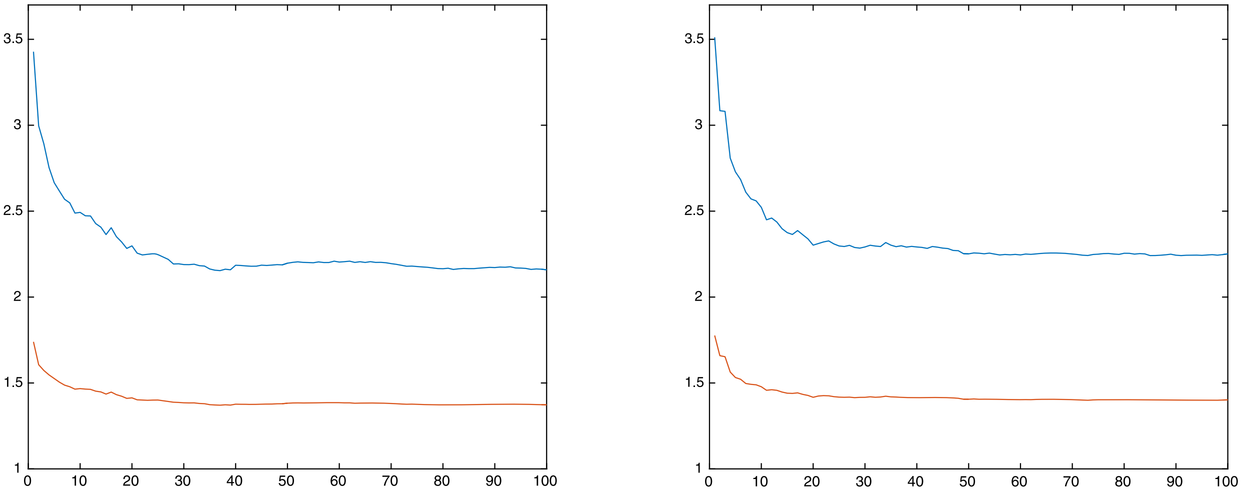

Fig. 6. The distortions of approximation for EMD with ground distance $|x - y|^\alpha$ with $\alpha = 1$ (blue, top curve) and $\alpha = .5$ (red, bottom curve), plotted as a function of the number of trees being averaged. Left: nets trees. Right: FRT trees. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

EMD is a natural metric to use to compare images, or indeed any kind of data with a lot of variability, as it is insensitive to perturbations in the data; see, for instance, Proposition 14 in [37]. However, EMD is typically not applied on images directly, as we are doing here, but rather on features that are extracted from the images, as in [12]. Our purpose in this experiment is only to illustrate the approximation ratios for estimating EMD using trees, not to suggest an algorithm for image analysis or digit classification.

In our experiments, we computed the EMD (the solution to the linear program (36), (37) and (38)) between 2000 random pairs of images from the database, taking as ground distance either the original Euclidean distance $|x - y|$ or the snowflake distance $|x - y|^{1/2}$. We compute the EMD using the code accompanying the paper [35]. We also compute the approximate EMD for each ground distance by building both nets and FRT trees on the grid, applying formula (39) for each tree, and taking the average. The results we report are averaged over five repetitions of the entire experiment.

In Fig. 6, we show the distortions incurred by the EMD approximations using the snowflake ground distance $|x - y|^\alpha$ for $\alpha = .5$ and $\alpha = 1$, for both nets and FRT trees. In addition to the distortions being fairly small, we observe that the value of adding more trees does not increase substantially after the first few; in other words, the experiments suggest that the number of trees needed to obtain decent approximations is quite small.

In Fig. 7, we simultaneously plot the distortion incurred by the EMD approximations with snowflake ground distance $|x - y|^\alpha$, alongside the distortion of the tree approximations to the underlying ground distance $|x - y|^\alpha$ on all pairs of points, using nets trees. Again, we use the values $\alpha = .5$ and $\alpha = 1$. Fig. 8 contains the same plots for FRT trees. We observe that the EMD approximation is considerably better than the approximation to the underlying tree distance, indicating that the estimate guaranteed by (41) may be somewhat pessimistic in practice. Of course, it cannot be improved upon in the worst case, as p and q may be taken to be two diracs.

In Fig. 9, we show scatterplots of the true EMDs against the approximate EMDs using 25 trees, for ground distance $|x - y|$. Fig. 10 contains the same plots, with the snowflaked ground distance $|x - y|^{1/2}$. As the plots lie very close to a straight line, the metrics are close to each other. These plots also provide a visual illustration that the distortion for EMD with snowflake ground distance is substantially smaller than that with non-snowflaked ground distance.

Acknowledgments

I thank Ronald Coifman for his feedback and support during the development of this work. I also thank Assaf Naor, Manor Mendal, and Ofer Neiman for helpful discussions. Finally, I thank the anonymous

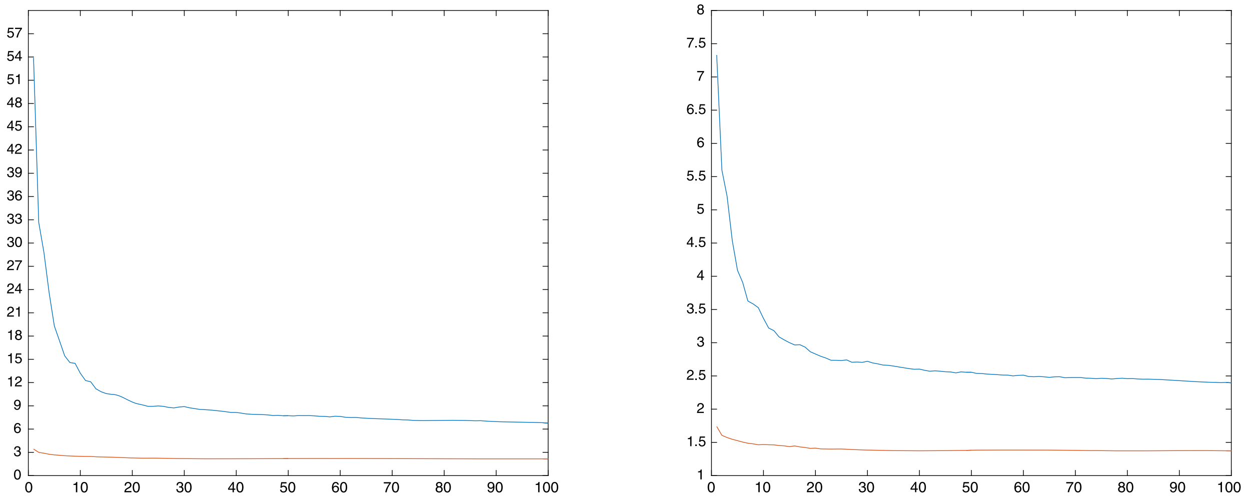

Fig. 7. The distortions of approximation for the ground distance $|x - y|^\alpha$ plotted alongside the distortions of approximation of EMD, using nets trees. Left: $\alpha = 1$. Right: $\alpha = .5$.

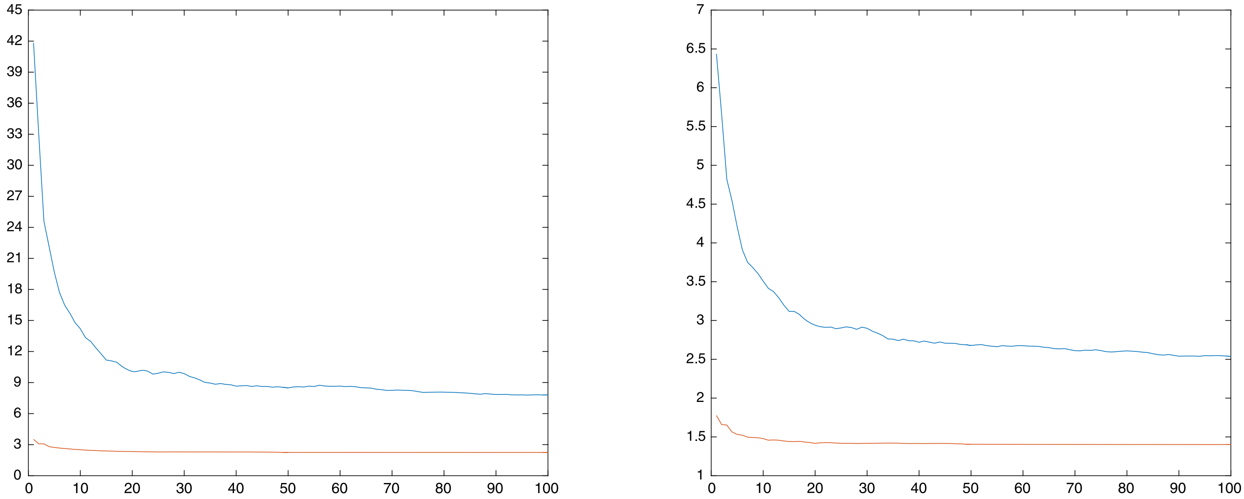

Fig. 8. The distortions of approximation for the ground distance $|x - y|^\alpha$ plotted alongside the distortions of approximation of EMD, using FRT trees. Left: $\alpha = 1$. Right: $\alpha = .5$.

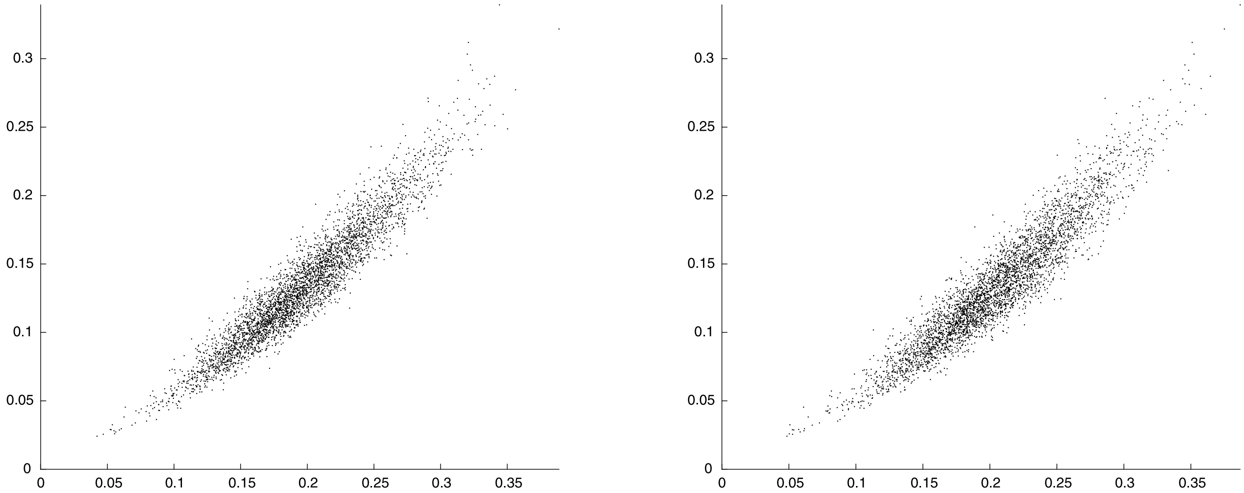

Fig. 9. True EMD versus approximate EMD, using 25 trees and $\alpha = 1$. Left: nets trees. Right: FRT trees.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

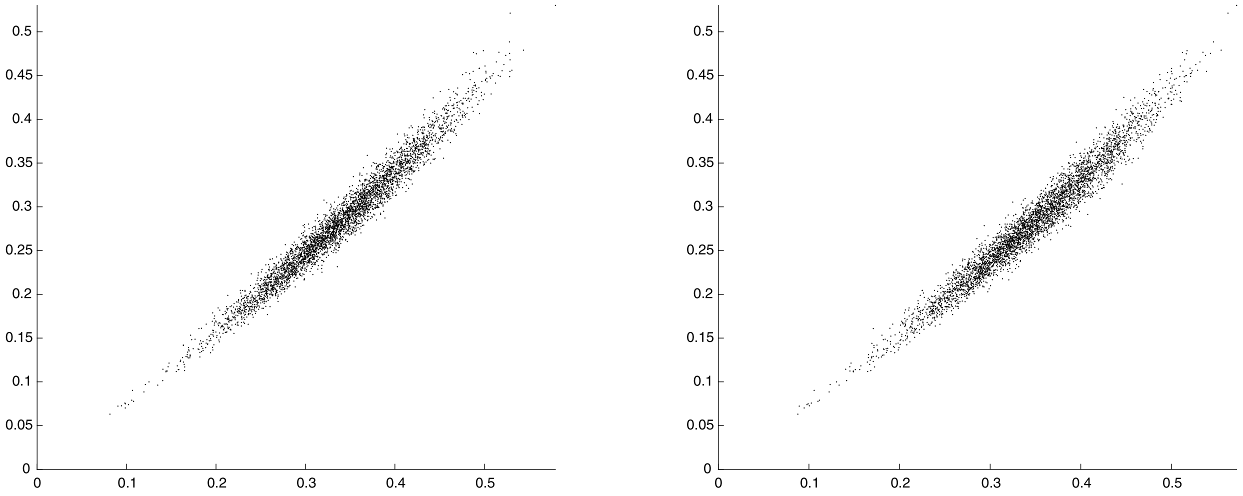

Fig. 10. True EMD versus approximate EMD, using 25 trees and $\alpha = .5$. Left: nets trees. Right: FRT trees.

reviewers for their insightful comments. I acknowledge support from the Simons Collaboration on Algorithms and Geometry.

Appendix A. The proof of Proposition 1

We prove Proposition 1 from Section 2. The proof is based on the proof of Lemma 2 from [19].

Proof of Proposition 1. Let $c = 2^d$. Suppose X consists of the samples $x_1, \dots, x_n \in M$, chosen uniformly and independently at random. For $r > 0$, let $V(x_i, r)$ denote the number of samples that fall within distance r of x_i. Temporarily fix any $1 \le i \le n$. Let $K_r = V(x_i, r)$, for brevity; note that since $x_i \in B(x_i, r)$, $K_r \ge 1$. We will estimate the probability that $K_{2r}/K_r \le 2c$.

Suppose $r \ge R/2$. Fix some $k = 1, \dots, n$, and let us condition on the event $K_{2r} = k$. Conditional on this event, $K_r - 1 \sim \text{Binomial}(p, k-1)$, where $p = \mu(B(x,r))/\mu(B(x,2r)) \ge 1/c$. Consequently, by a standard Chernoff bound we have

$$\begin{aligned} \Pr\left[K_r - 1 \ge \frac{k-1}{2c} \middle| K_{2r} = k\right] &\ge \Pr\left[K_r - 1 \ge p\frac{k-1}{2} \middle| K_{2r} = k\right] \\ &\ge 1 - e^{-p(k-1)/8} \ge 1 - e^{-(k-1)/8c}. \end{aligned} \tag{A.1}$$

Now, observe that the random variable $K_{2r} - 1 \sim \text{Binomial}(q, n-1)$, where $q = \mu(B(x, 2r))$. Therefore,

$$\begin{aligned} \Pr\left[\frac{K_{2r}}{K_r} \le 2c\right] &\ge \Pr\left[\frac{K_{2r}-1}{K_r-1} \le 2c\right] = \Pr\left[K_r - 1 \ge \frac{K_{2r}-1}{2c}\right] \\ &= \sum_{k=1}^{n} \Pr\left[K_r - 1 \ge \frac{k-1}{2c} \middle| K_{2r} = k\right] \Pr[K_{2r} = k] \\ &\ge 1 - \sum_{k=1}^{n} e^{-(k-1)/8c} \Pr[K_{2r} = k] = 1 - \sum_{k=0}^{n-1} e^{-k/8c} \Pr[K_{2r} - 1 = k] \\ &= 1 - (1 - q + qe^{-1/8c})^{n-1} \ge 1 - (1 - \eta + \eta e^{-1/8c})^{n-1} \end{aligned} \tag{A.2}$$

where we have used the formula for the moment-generating function of a Binomial random variable.

If $V(x_i, r)$ denotes the number of samples within radius r of x_i, we have shown that for any fixed $i = 1, \dots, n$ and any fixed radius $r \ge R$,

$$\Pr\left[\frac{V(x_i,2r)}{V(x_i,r)} \le 2c\right] \ge 1-(1-\eta+\eta e^{-1/8c})^{n-1}. \tag{A.3}$$

We will convert this into an estimate of the probability that the dimension of $X=\{x_1,\dots,x_n\}$ is bounded by $\log_2(2c)=d+1$ by using the union bound. Let $r_{i,j}$ be the distance between x_i and x_j. We observe that for all ratios $V(x_i,2r)/V(x_i,r)$ to be bounded by $2c$ for $r\ge R$, it is enough for the ratios $V(x_i,r_{i,j})/V(x_i,r_{i,j}/2)\le 2c$ whenever $r_{i,j}\ge 2R$. Indeed, under the latter assumption, if we take any $r\ge R$, take j so that $r_{i,j}$ is the largest $r_{i,j'}$ that is less than or equal to $2r$. Then $V(x_i,r_{i,j})=V(x_i,2r)$, and so

$$\frac{V(x_i,2r)}{V(x_i,r)} \le \frac{V(x_i,r_{i,j})}{V(x_i,r_{i,j}/2)} \le 2c. \tag{A.4}$$

Now, there are at most $2\binom{n}{2}\le n^2$ conditions $V(x_i,r_{i,j})/V(x_i,r_{i,j}/2)\le 2c$; consequently, the union bound implies that

$$\Pr[\dim_{\nu,R}(X)\le d+1] \ge 1-n^2(1-\eta+\eta e^{-1/8c})^{n-1} = 1-n^2\epsilon^{n-1} \tag{A.5}$$

with $\epsilon = 1-\eta+\eta e^{-1/8c}\in(0,1)$, completing the proof. □

References

[1] M. Gavish, R. Coifman, Harmonic analysis of digital databases, in: J. Cohen, A.I. Zayed (Eds.), Wavelets and Multiscale Analysis, Birkhäuser, 2011, pp. 161–197.

[2] M. Gavish, R.R. Coifman, Sampling, denoising and compression of matrices by coherent matrix organization, Appl. Comput. Harmon. Anal. 33 (3) (2012) 354–369.

[3] M. Gavish, B. Nadler, R.R. Coifman, Multiscale wavelets on trees, graphs and high dimensional data: theory and applications to semi supervised learning, in: Proceedings of the 27th International Conference on Machine Learning, ICML-10, 2010, pp. 367–374.

[4] Y. Bartal, Probabilistic approximation of metric spaces and its algorithmic applications, in: 37th Annual Symposium on Foundations of Computer Science, IEEE, 1996, pp. 184–193.

[5] Y. Bartal, On approximating arbitrary metrics by tree metrics, in: Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, ACM Press, 1998, pp. 161–168.

[6] N. Alon, R.M. Karp, D. Peleg, D. West, A graph-theoretic game and its application to the k-server problem, SIAM J. Comput. 24 (1) (1995) 78–100.

[7] M. Charikar, C. Chekuri, A. Goel, S. Guha, S. Plotkin, Approximating a finite metric by a small number of tree metrics, in: Proceedings of the 39th Annual Symposium on Foundations of Computer Science, IEEE, 1998, pp. 379–388.

[8] A. Gupta, Steiner points in tree metrics don't (really) help, in: Proceedings of the Twelfth Annual ACM–SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2001, pp. 220–227.

[9] J. Fakcharoenphol, S. Rao, K. Talwar, A tight bound on approximating arbitrary metrics by tree metrics, in: Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, ACM, 2003, pp. 448–455.

[10] M.S. Charikar, Similarity estimation techniques from rounding algorithms, in: Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing, 2002, pp. 380–388.

[11] S. Marinai, B. Miotti, G. Soda, Using earth mover's distance in the bag-of-visual-words model for mathematical symbol retrieval, in: 2011 International Conference on Document Analysis and Recognition, 2011, pp. 1309–1313.

[12] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, Int. J. Comput. Vis. 40 (2) (2000) 99–121.

[13] R. Sandler, M. Lindenbaum, Nonnegative matrix factorization with earth mover's distance metric for image analysis, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1590–1602.

[14] X. Wan, A novel document similarity measure based on earth mover's distance, Inform. Sci. 177 (18) (2007) 3718–3730.

[15] J. Heinonen, Lectures on Analysis on Metric Spaces, Springer-Verlag, 2001.

[16] A. Gupta, R. Krauthgamer, J.R. Lee, Bounded geometries, fractals, and low-distortion embeddings, in: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, 2003.

[17] P. Assouad, Plongements lipschitziens dans $\mathbb{R}^n$, Bull. Soc. Math. France 111 (4) (1983) 429–448.

[18] A. Naor, O. Neiman, Assouad's theorem with dimension independent of the snowflaking, Rev. Mat. Iberoam. 28 (4) (2012) 1–21.

[19] D.R. Karger, M. Ruhl, Finding nearest neighbors in growth-restricted metrics, in: Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing, 2002, pp. 741–750.

[20] D.G. Deng, Y. Han, Harmonic Analysis on Spaces of Homogeneous Type, Springer, 2009.

[21] A. Gupta, R. Krauthgamer, J.R. Lee, Bounded geometries, fractals, and low-distortion embeddings, in: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, IEEE, 2003, pp. 534–543.

[22] A. Beygelzimer, S. Kakade, J. Langford, Cover trees for nearest neighbor, in: Proceedings of the 23rd International Conference on Machine Learning, ICML-10, 2006, pp. 97–104.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002

[23] T.-H.H. Chan, K. Dhamdhere, A. Gupta, J. Kleinberg, A. Slivkins, Metric embeddings with relaxed guarantees, SIAM J. Comput. 38 (6) (2009) 2303–2329.

[24] K. Hildrum, J. Kubiatowicz, S. Ma, S. Rao, A note on the nearest neighbor in growth-restricted metrics, in: Proceedings of the Fifteenth Annual ACM–SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2004, pp. 560–561.

[25] I. Abraham, C. Gavoille, A.V. Goldberg, D. Malkhi, Routing in networks with low doubling dimension, in: Proceedings of the 26th IEEE International Conference on Distributed Computing Systems, IEEE, 2006, pp. 1–10.

[26] H.T.-H. Chan, A. Gupta, B.M. Maggs, S. Zhou, On hierarchical routing in doubling metrics, in: Proceedings of the Sixteenth Annual ACM–SIAM Symposium on Discrete Algorithms, 2005, pp. 762–771.

[27] S. Dasgupta, Y. Freund, Random projection trees and low dimensional manifolds, in: Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing, ACM, 2008, pp. 537–546.

[28] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.

[29] R.R. Coifman, S. Lafon, Diffusion maps, Appl. Comput. Harmon. Anal. 21 (1) (2006) 5–30.

[30] A. Brudnyi, Y. Brudnyi, Methods of Geometric Analysis in Extension and Trace Problems, vol. 1, Springer, 2012.

[31] S. Semmes, On the nonexistence of bilipschitz parameterizations and geometric problems about A_∞-weights, Rev. Mat. Iberoam. 12 (2) (1996) 337–410.

[32] Y. Meyer, Wavelets and Operators, Cambridge University Press, 1992.

[33] J.R. Lee, A. Naor, Extending Lipschitz functions via random metric partitions, Invent. Math. 1 (2005) 59–95.

[34] S. Shirdhonkar, D.W. Jacobs, Approximate earth mover's distance in linear time, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[35] O. Pele, M. Werman, Fast and robust earth mover's distances, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 460–467.

[36] W. Leeb, The mixed Lipschitz space and its dual for tree metrics, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.06.008.

[37] W. Leeb, R. Coifman, Hölder–Lipschitz norms and their duals on spaces with semigroups, with applications to Earth mover's distance, J. Fourier Anal. Appl. 22 (4) (2016) 910–953.

Please cite this article in press as: W. Leeb, Approximating snowflake metrics by trees, Appl. Comput. Harmon. Anal. (2016), http://dx.doi.org/10.1016/j.acha.2016.10.002