Testing Metric Properties

Michal Parnas
The Academic College
of Tel-Aviv-Yaffo
Tel-Aviv, Israel
michalp@mta.ac.il

Dana Ron
Department of EE – Systems
Tel-Aviv University
Ramat Aviv, Israel
danar@eng.tau.ac.il

Abstract

Finite metric spaces, and in particular tree metrics play an important role in various disciplines such as evolutionary biology and statistics. A natural family of problems concerning metrics is deciding, given a matrix M, whether or not it is a distance metric of a certain predetermined type. Here we consider the following relaxed version of such decision problems: For any given matrix M and parameter ϵ, we are interested in determining (by probing M) whether M has a particular metric property P, or whether it is ϵ-*far* from having the property. In ϵ-*far* we mean that at least an ϵ-fraction of the entries of M must be modified so that it obtains the property. The algorithm may query the matrix on entries $M[i, j]$ of its choice, and is allowed a constant probability of error.

We describe algorithms for testing Euclidean metrics, tree metrics and ultrametrics. Furthermore, we present an algorithm that tests whether a matrix M is an *approximate ultrametric*. In all cases the query complexity and running time are polynomial in $1/\epsilon$ and independent of the size of the matrix. Finally, our algorithms can be used to solve relaxed versions of the corresponding search problems in time that is sub-linear in the size of the matrix.

1 Introduction

Finite metric spaces, and in particular tree metrics play an important role in various disciplines such as evolutionary biology and statistics (see for example [20, 10, 13, 5]). A tree metric is defined by a weighted tree that spans a set of points, where the distance between two points equals the sum of the weights on the edges along the path between these points. Ultrametrics are a special case of tree metrics, which are of particular interest. In Ultrametric trees all points correspond to leaves of the tree, and the tree can be rooted so that the distance from the root to every leaf is the same. Tree metrics, also known as *additive metrics*, are especially appealing since they can be used to model hierarchical structures. For example, in the context of evolutionary biology, a tree metric can be defined on species, where the weights of the tree edges are determined by the time elapsed since the species separated.

A natural family of problems concerning metrics is deciding, given a matrix M, whether or not it is a distance metric of a certain predetermined type.[1] Specifically, we may be interested in knowing whether the matrix is a tree metric, an ultrametric, or possibly a Euclidean metric of some bounded dimension d (i.e., whether there exists an embedding of the points in the d-dimensional Euclidean space, whose pairwise distances correspond to the entries of the matrix).

In this paper we study relaxed versions of these decision problems, from within the framework of *property testing*. Namely, instead of determining whether M has a certain metric property P *or not*, we would like to determine whether it has property P or *should be modified significantly* in order to obtain P. More precisely, given query access to an $n \times n$ matrix M, and a distance parameter ϵ, the goal is to determine with high probability whether M has the property P or whether more than an ϵ-fraction of its entries should be modified so that it obtains the property. In the latter case we say that M is *ϵ-far* from having the property. Given this relaxation, we seek algorithms that are much more efficient than those required for exactly deciding the property. In particular, we are interested in algorithms that have complexity that is sub-linear in the size of the matrix, or even independent of this size, and polynomial in $1/\epsilon$.

Our Results. All our algorithms work by taking a uniformly selected sample S from $[n] = \{1, \ldots, n\}$, whose size is polynomial in $1/\epsilon$ (and *independent* of n). Then the algorithms query M on entries $M[i,j]$ for pairs of points $i, j \in S$. In what follows we state the size of the sample S. The query complexity and running time of the algorithms is at most quadratic in the sample size.

- We show that it is possible to test whether a matrix is an ultrametric using a sample of size $O(\log(1/\epsilon)/\epsilon^3)$.
- The size of the sample sufficient for testing whether a matrix is a general tree metric is $O(\log(1/\epsilon)/\epsilon^3)$ as well.
- The sample size sufficient for testing whether a matrix is a *d-dimensional Euclidean Metric* is $O(d\log(d)/\epsilon)$.
- We also consider the problem of testing whether a matrix is an *approximate ultrametric*. For a given approximation parameter δ, we say that a matrix M is a *δ-approximate* ultrametric (or simply a δ-ultrametric) if there exists an ultrametric matrix M' such that for every $i, j \in [n]$, $|M[i,j] - M'[i,j]| \leq \delta$. For any given approximation parameter δ (and for every distance parameter ϵ), we require that the testing algorithm accept M if it is a δ-ultrametric, and reject

[1] With a slight abuse of terminology, we use the term *distance* while actually we mean *pseudo-distance*. Namely, the properties of symmetry, triangle-inequality, and non-negativity are obeyed, but we do not necessary require that $M[i,j] > 0$ for every $i \neq j$. As we later show, this relaxation is inevitable for the problems we solve.

M with high probability if it is ϵ-far from from being a $c \cdot \delta$-ultrametric (where c is a fixed constant). The sample used is of size $O(\log(1/\epsilon)/\epsilon^3)$.

Our testing algorithms can be used to solve relaxed versions of the related search problems of all properties presented above, in time linear in n and polynomial in $1/\epsilon$. For example, in the case of tree metrics, we can construct a tree that agrees with M on all but an ϵ-fraction of its entries. Note that these procedures are actually sub-linear in the size of the matrix, and in particular observe only a small fraction of the matrix.

TECHNIQUES. All our results have a common thread. As noted previously, our algorithms all take a uniform sample of points from $[n]$. Specifically, the algorithms select two sub-samples, where each serves a different role. The first sub-sample is used to induce certain constraints on almost all entries in the matrix. These constraints are *always* satisfied in case the matrix has the property. The heart of our proofs is in showing that in case the matrix is far from having the property, then necessarily there are many entries in the matrix that violate the constraints induced by the first sub-sample. The second sub-sample is then used to provide *witnesses* to these violations.

In order to prove that the first sub-sample induces such constraints, we view it as being selected in *phases*. Each phase either adds more constraints, or contains itself a witness to the fact that the matrix does not have the tested property.

It is interesting to note that a similar proof structure has been useful in very different contexts of property testing (e.g., graph properties [14] and clustering [2]), and so it is possible that some unifying approach may be found.

CONTEXT AND RELATED WORK. Property testing was first defined and applied in the context of algebraic properties of functions [19], and has since been applied in various contexts such as properties of graphs (e.g., [14, 15, 3]), strings (e.g., [4]), clusters of points [2], and geometric objects [11, 8]. For a survey see [18].

The research on metric spaces is clearly too rich and broad to cover within the limits of this introduction. Here we only mention the most closely related results.

Finding a tree that corresponds (or approximately corresponds) to a given distance matrix, is usually referred to as the *Numerical Taxonomy Problem*. This problem was first explicitly stated in [6]. Waterman et. al. [22] showed that if a given matrix M is a tree metric, then there is a unique tree that corresponds with M, and it can be constructed in time $O(n^2)$. Culberson and Rudnicki [7] describe an algorithm that has a running time of $O(kn \log_k n)$ when the degree of the tree is bounded by k. The problem of constructing an ultrametric tree for a given matrix (if such a tree exists) is clearly a special case of the above, and there are simpler procedures (though not more efficient in general) for constructing such a tree (e.g. [16]). Deciding whether M is a tree metric or ultrametric can clearly be done by trying to construct the tree. To the best of our knowledge, no faster decision algorithm is known. For d-dimensional Euclidean metrics, a decision can be performed in time polynomial in the size of the matrix, by checking that a related matrix is a positive semidefinite matrix of rank at most d.

When the matrix M is not a tree metric (or an ultrametric), then we may consider the problem of finding a tree metric (ultrametric) M' such that $\|M - M'\|_p$ is minimized for a given L_p norm. It was shown by Day [9] that this problem is NP-hard for the L_1 and L_2 norms, for both general tree metrics and ultrametrics. When the L_∞ norm is considered, then the problem can be solved in time $O(n^2)$ for ultrametrics [12, 17]. However, in the case of general tree metrics the problem is also NP-hard for the L_∞ norm [1]. The problem remains NP-hard even when we are given a matrix M for which there exists a tree metric M' such that $\|M - M'\|_\infty \le \delta$, and we are required to find a tree metric M'' such that $\|M - M''\|_\infty \le \frac{9}{8}\delta$ [1]. On the bright side, it should be noted that it is possible to find in time $O(n^2)$ a tree metric M'' such that $\|M - M''\|_\infty \le 3\delta$ [1].

Recall that for both exact search problems (for ultrametrics and general tree metrics), and for approximate ultrametrics (where the approximation is with respect to the L_∞ norm), we solve relaxed versions of these problems in time linear in n (and polynomial in $1/\epsilon$). We believe that our results can be extended to deal with the approximation of general trees as well.

ORGANIZATION. In Section 2 we provide the preliminaries for this work. In Section 3 we discuss ultrametrics, in Section 4, approximate ultrametrics, and in Section 5, general tree metrics. Testing Euclidean metrics is considered in Section 6. Finally, in Section 7 we show our lower bounds.

2 Preliminaries

In all that follows we consider matrices whose entries are rational numbers.

Definition 2.1 (Distance to Having a Property) *Let P be a* property *of matrices, let M be an $n \times n$ matrix, and let $0 \le \epsilon \le 1$. The matrix M is ϵ-*far *from having property P, if the minimum fraction of M's entries (among all n^2 entries) that should be modified so that M obtains property P is greater than ϵ. Otherwise, M is ϵ-*close *to having property P.*

Definition 2.2 (Testing Properties of Matrices) *A* testing algorithm *for a matrix property P is given a distance parameter ϵ and may query M on entries $M[i,j]$ of its choice. If M has property P then the algorithm should* accept*, and if M is ϵ-far from having property P, then the algorithm should* reject *with probability at least $2/3$.*

The above definition requires that the algorithm have a one-sided error probability. In general, property testing algorithms may be allowed a two-sided error. However, since all our algorithms have a one sided error, we shall use this more restricted definition.

The matrix properties we shall be interested in are all properties of distance or pseudo-distance matrices.

Definition 2.3 (Distance Matrices) *We say that an $n \times n$ matrix M is a* pseudo-distance *matrix if the following conditions hold:*

1. *For every $i, j \in [n]$, $M[i,j] \ge 0$, where $M[i,i] = 0$ for every i* (non-negativity).
2. *For every $i, j \in [n]$, $M[i,j] = M[j,i]$* (symmetry).
3. *For every $i, j, k \in [n]$, $M[i,j] \le M[i,k] + M[k,j]$* (triangle inequality).

If Item 1 is strengthened to require that $M[i,j] > 0$ for every $i \ne j$, then M is a distance *matrix.*

With a slight abuse of terminology, we shall sometimes refer to M as being a metric or pseudo-metric.[2] As we shall see, in some cases the additional requirement that $M[i,j]$ be strictly positive

[2]Formally, a (pseudo) metric is a pair (X, d) where X is a set and $d : X \times X \to \Re_{\ge 0}$ is a (pseudo) distance function. Hence, if M is a (pseudo) distance matrix, then $([n], d_M)$, where $d_M(i,j) = M[i,j]$, is a (pseudo) metric.

for $i \neq j$, makes the task of testing significantly harder. In particular, in these cases, without this requirement there exists a testing algorithm having complexity poly$(1/\epsilon)$ (that is, independent of n), while adding the requirement implies a lower bound of $\Omega(\sqrt{n})$ (for a constant ϵ).

In what follows we assume for simplicity that M obeys the first two conditions in the above definition of pseudo-distance matrices. We next argue that this assumption can be made without loss of generality.

Proposition 1 *Let A be a testing algorithm for a property P of pseudo-distance matrices, whose correctness relies on M obeying conditions (1) and (2) in Definition 2.3. Then there exists a testing algorithm A' for the property P whose correctness does not rely on these assumptions. Furthermore, if $Q_A(\epsilon)$ and $T_A(\epsilon)$ are the query complexity and running time of A, respectively, then the query complexity and running time of A' are $O(Q_A(\epsilon/2) + 1/\epsilon))$ and $O(T_A(\epsilon/2) + 1/\epsilon))$, respectively.*

Proof: First assume that A is such that its failure probability is at most $1/6$ instead of $1/3$ (since this can easily be achieved by running the algorithm several times and taking the majority output). Algorithm A' will start by taking a uniform sample of $6/\epsilon$ pairs of indices $i, j \in [n]$, and checking whether (i) $M[i,j] \geq 0$ (where $M[i,j] = 0$ if $i = j$). (ii) $M[i,j] = M[j,i]$. If any of the pairs selected does not satisfy either (i) or (ii) then A' rejects. If the total number of pairs $i, j \in [n]$ that do not satisfy either (i) or (ii) is greater than $\frac{\epsilon}{2}n^2$ then the algorithm will select such a pair with probability at least $1 - (1 - \frac{\epsilon}{2})^{6/\epsilon} > 1 - e^{-3} > 5/6$ and reject. On the other hand, if M has property P, and thus in particular is a pseudo-distance matrix, then no such pair exists.

Thus, assume that the number of pairs of indices in $[n] \times [n]$ that do not satisfy either (i) or (ii) is at most $\frac{\epsilon}{2}n^2$. Consider a matrix M' that differs from M only on those indices, and satisfies (i) and (ii) for all i, j. More precisely: For every $i \in [n]$, $M'[i,i] = 0$; For every pair $i \neq j$ such that $M[i,j] < 0$ we set $M'[i,j]$ to some arbitrary non-negative value. For every pair i, j such that $M[i,j] \neq M[j,i]$, we set $M'[i,j] = M'[j,i] = M[i,j]$. Since M and M' differ on at most $\frac{\epsilon}{2}n^2$ entries, if M is ϵ-far from having property P then M' is $\frac{\epsilon}{2}$-far from having property P.

Now A' applies A with the distance parameter set to $\frac{\epsilon}{2}$. For each entry $M[i,j]$ that A observes, A' also checks that (i) and (ii) are satisfied. If for any entry either (i) or (ii) is not satisfied, then A' rejects. Otherwise, it outputs the accept/reject decision of A.

Since for every M that has property P, algorithm A always accepts, the same is true for A'. On the other hand, if M is ϵ-far from having property P, then for M' as defined above, algorithm A should reject M' with probability at least $5/6$. But this directly implies that A' rejects M with probability at least $5/6$ as well. ■

3 Testing Ultrametrics

In this section we present an algorithm that tests whether a given matrix M is an ultrametric. Some of the ideas introduced in this section serve as a basis for our results in the next two sections. Here we assume that M is actually strictly positive everywhere except on its diagonal. This assumption can be made without loss of generality by a slight variant of Proposition 1. We start with a few definitions.

Let T be a tree with positive weights on the edges. We view the weight of each edge as its length. The distance between two nodes i and j in T is defined as usual as the sum of the weights on the path from i to j. This distance will be denoted by $T(i,j)$. For every node i, $T(i,i)$ is defined to be 0.

Definition 3.1 (Ultrametric Trees) *We say that a tree T with positive weights on the edges is an* ultrametric tree *if the following holds:*

1. *T is rooted and the distance between* every *leaf and the root is the same fixed value.*
2. *All internal nodes in T have at least 2 children.*

Definition 3.2 (Ultrametrics) *We say that an $n \times n$ matrix M is an* ultrametric *if there exists an ultrametric tree T for which the following holds:*

1. *There exists a 1-to-1 mapping ϕ from $[n]$ onto the leaves of T.*
2. *For any two leaves i, j in the tree, $T(\phi(i), \phi(j)) = M[i,j]$.*

With a slight abuse of notation, we shall write $T(i,j)$ instead of $T(\phi(i), \phi(j))$.

The following fact is sometimes used as an alternative definition for ultrametrics, and it will assist us in our proofs.

Fact 1 (The Three Points Condition) *A metric is an ultrametric if and only if for every i, j, k,*

$$M[i,j] \leq \max\{M[i,k], M[j,k]\} \ .$$

In particular this implies that for every i, j and k, if $M[i,k] > M[j,k]$ then $M[i,j] = M[i,k]$.

Since our algorithm will try to construct a tree on a subset of points in $[n]$, the following definition will be useful.

Definition 3.3 (Consistent Trees) *For a given matrix M and a subset $U \subseteq [n]$, we say that an ultrametric tree T_U is* consistent *with M on U if for every $i, j \in U$, $T_U(i,j) = M[i,j]$. When $U = [n]$ we simply say that T is consistent with M.*

3.1 Constructing Ultrametric Trees

It is well known that if M is an ultrametric then there exists a unique ultrametric tree T that is consistent with M. Furthermore, such a tree can be found efficiently (see for example [16]).

Here we describe an iterative procedure for constructing an ultrametric tree that is consistent with M on a given subset of $[n]$ (assuming that such a tree exists). The presentation of this procedure will aid us in describing and analyzing our testing algorithm. For the sake of the presentation we assume that the given subset is $\{1, \ldots, s\}$.

Ultrametric Tree Construction Procedure
Input: an $n \times n$ matrix M; a subset $\{1, \ldots, s\}$ of indices.

	1	2	3	4	5
1		8	10	10	10
2			10	10	10
3				2	6
4					6
5					

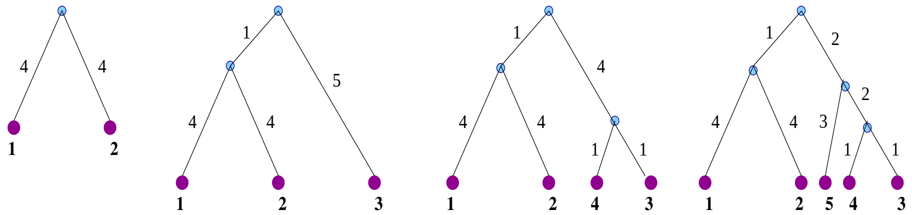

Figure 1: Construction of an ultrametric tree that is consistent with the accompanying matrix (which is symmetric and 0 on the diagonal). In the first stage **1** and **2** are put at equal distance 4 (half the distance between them) from the root. When **3** is added, it is at distance greater than 8 from **1**, and so a new root is created. When **4** is added, the closest point is **3**, and the point of departure of **4** is at distance 1 from **3** on the path between **3** and the root. Finally, **5** is closest to **3** as well, and its point of departure is at distance 3 from node **3**.

1. Initialize $U = \{1, 2\}$ and let T_U consist of a root r, and two leaves, 1 and 2 that are at equal distance $\frac{M[1,2]}{2}$ from r.
2. For $j = 3, \ldots, s$
   (a) $T_{U \cup \{j\}} \leftarrow$ **Add-Point**(j, T_U, M).
   (b) $U \leftarrow U \cup \{j\}$.

Procedure Add-Point
Input: an $n \times n$ matrix M; an index j, and an ultrametric tree T_U that is consistent with M on $U = \{1, \ldots, j-1\}$.

1. Let k, $1 \leq k < j$ be any point for which $M[j,k]$ is minimized.
2. If $M[j,k] > 2 \cdot T_U(r,k)$ (where r is the root of T_U), then create a new root, add an edge of length $\frac{M[j,k]}{2}$ between the new root and the new leaf j, and connect the old root of T_U to the new root by an edge of length $\frac{M[j,k]-T_U(r,k)}{2}$. Let the new root now be called r.
3. Otherwise, either (i) there exists a node p in T_U on the path from k to the root r such that $T_U(k,p) = \frac{M[j,k]}{2}$, or (ii) there exists an edge (u,v) on the path from k to r such that $T_U(k,u) < \frac{M[j,k]}{2} < T_U(k,v)$. In case (ii), replace the edge (u,v) with two edges (u,p) and (p,v) so that the distance from k to p equals $\frac{M[j,k]}{2}$. In either case add an edge from p to a new leaf j having length $\frac{M[j,k]}{2}$.

We refer to the node p defined in Step 3 of the procedure Add-Point as the *departure point* of j from T_U. If j causes the creation of a new root (Step 2), then its departure point is defined to be the previous root. For an illustration of the above construction see Figure 1.

Lemma 3.1 *If M is an ultrametric, then T_U as constructed in the Ultrametric Tree Construction Procedure, is consistent with M on U.*

Proof: We prove the lemma by induction on j. The base case, $j = 2$, is straightforward. Let $U = \{1, ..., j-1\}$ and assume by the induction hypothesis that T_U is consistent with M on U for $j - 1 \geq 2$. We show that after the addition of j to the tree, $T_{U \cup \{j\}}$ is consistent with M on $U \cup \{j\}$.

Note that all distances in $T_{U\cup\{j\}}$ between pairs of points that are different than j, are exactly as in T_U. By construction, $T_{U\cup\{j\}}(k,j) = M[k,j]$. For any $i \in U$ such that $i \neq k$, we consider the following three cases:

1. $\boldsymbol{M[k,i] > M[k,j]}$: In this case, since M is an ultrametric, we have that $M[j,i] = M[k,i]$. Let p' be the least common ancestor of k and i, so that $T_{U\cup\{j\}}(i,p') = \frac{M[k,i]}{2}$. By construction of the tree,
$$\begin{aligned} T_{U\cup\{j\}}(j,i) &= T_{U\cup\{j\}}(j,p) + T_{U\cup\{j\}}(p,p') + T_{U\cup\{j\}}(p',i) \\ &= \frac{M[k,j]}{2} + \left(\frac{M[k,i]}{2} - \frac{M[k,j]}{2}\right) + \frac{M[k,i]}{2} \\ &= M[k,i] \\ &= M[j,i] \end{aligned} \tag{1}$$

2. $\boldsymbol{M[k,i] < M[k,j]}$: In this case it must hold that $M[j,i] = M[k,j]$, and a similar calculation to the one in the previous item shows that $T_{U\cup\{j\}}(j,i) = M[k,j]$, and so $T_{U\cup\{j\}}(j,i) = M[j,i]$ as required.

3. $\boldsymbol{M[k,i] = M[k,j]}$: Here we only know that $M[j,i] \le M[k,j]$, but since k was chosen to be closest to j in T_U, it must be that $M[j,i] = M[k,j]$. Here the construction ensures that $T_{U\cup\{j\}}(j,i) = T_{U\cup\{j\}}(j,k)$ and so $T_{U\cup\{j\}}(j,i) = M[j,i]$.

■

3.2 Testing Ultrametrics

As noted above, for any ultrametric M and subset $U \subseteq [n]$, there is a unique tree T_U that is consistent with M on U. While the pairwise distances between points in $[n]\setminus U$ and points in U do not uniquely determine the position in the tree of every point in $[n]\setminus U$, a small sample of points can be used to construct a "scaffold" tree that induces certain constraints on all other points. In case M is an ultrametric then these constraints are always obeyed. We shall prove that if M is ϵ-far from being an ultrametric then with high probability over the choice of the sample, there are many points (or pairs of points) that do not obey the constraints induced by the scaffold tree. To this end we first need to introduce a few definitions.

For a subset $U \subset [n]$, let T_U be an ultrametric tree whose leaves are associated with the points in U. We refer to T_U as a *scaffold*. We start by considering how a scaffold that is consistent with M on U restricts the distances of a point $j \notin U$ to the points in U.

Definition 3.4 (Consistent points) *Let T_U be an ultrametric tree that is consistent with M on U. We say that a point $j \notin U$ is* consistent *with T_U, if after adding j to T_U by applying the procedure Add-Point(j, T_U, M), the resulting tree $T_{U\cup\{j\}}$ in an ultrametric tree. Otherwise, j is* inconsistent *with T_U. The set of points in $[n]\setminus U$ that are consistent with T_U is denoted by Γ_U.*

If M is an ultrametric, then $\Gamma_U = [n]\setminus U$ for every U. Hence, a point $j \notin U$ that is inconsistent with T_U provides evidence that M is not an ultrametric. Since T_U is uniquely defined given U, we can refer to points as being consistent or inconsistent with U (instead of T_U).

We now show that the scaffold also restricts the distances between pairs of points that do not belong to U. We first introduce the notion of a partition induced by U.

Definition 3.5 (The scaffold partition) *Let $U \subset [n]$ be such that there exists an ultrametric tree T_U that is consistent with M on U. For each point $j \in \Gamma_U$, consider all its distances to points in U (according to M). Then two points belong to the same* class *in the* partition *$\mathcal{P}_U$ of Γ_U, if all their pair-wise distances to points in U are the same.*

For an illustration of the partition see Figure 2.

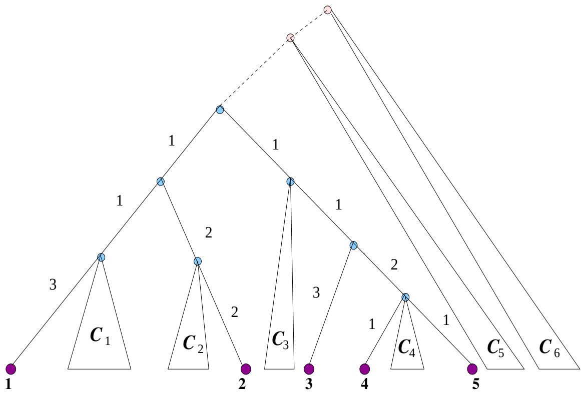

Figure 2: An illustration of the partition induced by a scaffold. For example, the points in C_1 are all at distance 6 from **1**, at distance 8 from **2**, and at distance 10 from points **3**, **4** and **5**. The scaffold distance between every $i \in C_1$ and $j \in C_2$ is 8.

Observe that for any class C of the partition $\mathcal{P}_U$, all points in C have the same point of departure from T_U. Furthermore, with the exception of the points whose point of departure is the root of T_U, if i and j have the same point of departure from T_U, then they are in the same class. Also observe that if M is an ultrametric, then each class C corresponds to a subtree in the ultrametric tree T that is consistent with M.

Definition 3.6 (The scaffold distance D_U) *Let T_U be an ultrametric tree that is consistent with M on U. Consider (as a mental experiment) adding all points in $[n] \setminus U$ to T_U by applying the procedure Add-Point(j, T_U, M) to all points $j \notin U$* in parallel, *and let the resulting tree be $\widehat{T_U}$.*[3] *Then, let $D_U(\cdot,\cdot)$ be the distance induced by $\widehat{T_U}$ (which is, by definition, an extension of the distance induced by T_U on the pairs of points in U).*

It is easy to verify that if M is an ultrametric then for every pair of points i, j that belong to different classes of $\mathcal{P}_U$, $D_U(i,j) = M[i,j]$ (since adding i and j to T_U sequentially, and in parallel results in the same tree). Similarly, for every pair of points i, j that belong to the same class, $M[i,j] \leq D_U(i,j)$.

Therefore, if one of the above is violated for a pair of points $i, j \in \Gamma_U$, then we have evidence that M is not an ultrametric. This observation motivates the following definition.

[3]To be a little more precise, let B be the maximum value in M, and consider first adding to U a fictitious point x whose distance from all points is greater than B. If we now consider adding all points in $[n] \setminus U$ to $T_{U \cup \{x\}}$ then there is never a need to create a new root, and the addition process is well defined. We can now remove the point x from the tree, and let the resulting tree be $\widehat{T_U}$. Also note that by our assumption that $M[i,j] > 0$ for every $i \neq j$, a point j cannot be added in the same place in the tree as an existing point in U.

Definition 3.7 (Violating pairs) *Let T_U be an ultrametric tree that is consistent with M on U. A pair of points $i, j \in \Gamma_U$ are said to be a* violating pair *with respect to T_U, if either (1) i and j are in different classes in $\mathcal{P}_U$ and $M[i,j] \neq D_U(i,j)$, or (2) i and j belong to the same class in $\mathcal{P}_U$ and $M[i,j] > D_U(i,j)$.*

As noted above, if M is an ultrametric, then there are no inconsistent points and no violating pairs with respect to T_U, for any subset U. We shall show that if M is ϵ-far from being an ultrametric, then with high probability over the choice of a sufficiently large sample U, either there are many inconsistent points or many violating pairs with respect to T_U.

We are now ready to present our testing algorithm.

Algorithm 1 Testing Algorithm for Ultrametrics

1. *Uniformly and independently select $s = O(\log(1/\epsilon)/\epsilon^3)$ points from $[n]$. Denote the set of points selected by U.*
2. *Construct a scaffold tree T_U as described in the Ultrametric Tree Construction Procedure.*
3. *If for any two points $i, j \in U$, $T_U(i,j) \neq M[i,j]$ then* reject.
4. *Uniformly and independently select $m = O(1/\epsilon)$ pairs of points in $[n]$.*
5. *If any of the points selected in Step 4 is inconsistent with T_U then* reject.
6. *Partition the $2m$ points selected in Step 4 into classes according to the partition $\mathcal{P}_U$ induces by the scaffold. If any of the pairs is a violating pair then* reject.
7. *If no step caused rejection then* accept.

Theorem 2 *Algorithm 1 is a testing algorithm for ultrametrics.*

Note that whenever the algorithm rejects then it provides *evidence* that M is not an ultrametric.

As a corollary of Theorem 2 we get:

Corollary 3 *Let the "natural" testing algorithm be the algorithm that simply selects a uniform sample of $s + 2m = O(\log(1/\epsilon)/\epsilon^3)$ points from $[n]$ (where s and m are defined as in Algorithm 1), and accepts if and only if there is an ultrametric tree consistent with M on the sample. Then the natural algorithm is a testing algorithm for ultrametrics.*

As discussed previously, if M is an ultrametric then it is always accepted. We thus assume from now that M is ϵ-far from being an ultrametric, and strive to show that it is rejected with probability at least 2/3. Before embarking on the proof of this part of Theorem 2, we try and gain intuition by considering the following special case. In order to describe it we introduce the following definition.

Definition 3.8 (Separated points) *Let $U \subset [n]$ be such that there exists an ultrametric tree T_U that is consistent with M on U. A pair of points $i, j \in \Gamma_U$ are said to be* separated *with respect to U if they belong to different classes of the partition $\mathcal{P}_U$. Otherwise, they are* non-separated.

Suppose that the initial sample $U \subset [n]$ is such that the number of non-separated pairs of points in Γ_U is at most $\frac{\epsilon}{3}n^2$. We claim that in this case if M is ϵ-far from being an ultrametric, then either there are more than $\frac{\epsilon}{3}n$ inconsistent points, or there are more than $\frac{\epsilon}{3}n^2$ violating pairs with respect to T_U. This would cause the algorithm to reject with high probability either in Step 5 or Step 6 of the algorithm. To see why the claim is true, let us define a matrix M' such that $M'[i,j] = D_U(i,j)$ for every $i,j \in [n]$. Thus, M' is an ultrametric by definition (as it is defined by the ultrametric tree $\widehat{T_U}$). However, it is not hard to verify that M' and M differ on at most ϵn^2 entries (at most $\frac{\epsilon}{3}n^2$ entries due to non-separated pairs, at most $\frac{\epsilon}{3}n^2$ due to violating pairs, and at most $\frac{\epsilon}{3}n^2$ pairs of points in which at least one of the points is inconsistent with T_U). But this contradicts our assumption on M.

Roughly speaking, this scenario suggests that we gain from *separating* points into different classes. This motivates the following definition.

Definition 3.9 (Separators) *We say that a point k is a* separator *for a pair of points i,j, if $M[i,k] \neq M[j,k]$.*

Thus, a pair of points in Γ_U are separated with respect to U (as defined in Definition 3.8), if and only if they have a separator in U. Notice that if M is an ultrametric then a point k can separate only pairs of points i,j that belong to the same class as k.

Definition 3.10 (Effective Separators) *We say that point k is an α-*effective separator *with respect to U, if the number of pairs of points in Γ_U that are not separated with respect to U but are separated with respect to $U \cup \{k\}$, is at least $(\alpha n)^2$.*

By the above definition, the addition to U of a point $k \in \Gamma_U$ that is an α-effective separator with respect to U, has the following effect. For at least $(\alpha n)^2$ pairs of points in Γ_U, either both points in a pair are in $\Gamma_{U\cup\{k\}}$ and are now separated by $U \cup \{k\}$, or at least one of the points in the pair is not in $\Gamma_{U\cup\{k\}}$ (that is, it is inconsistent with T_U). In either case, the number of non-separated pairs of consistent points decreases by at least $(\alpha n)^2$.

We shall view the sample U as being selected in *phases*. As we prove more formally subsequently, as long as there is a sufficient number of α-effective separators with respect to the sample selected so far, then with high probability a new separator is selected in the next phase, and separates many pairs of points. However, what if there are only a few effective separators with respect to the sample selected so far, but there are still many non-separated pairs? In this case we can prove the following lemma concerning the distances between points that belong to the *same class* (non-separated pairs).

Lemma 3.2 *Let $C \subseteq [n] \setminus U$ be a class in $\mathcal{P}_U$ such that there are at most βn points in C that are α-effective separators with respect to U. Then there exists an ultrametric (star shaped) tree T_C such that for at most $(2\beta + 3\alpha)n \cdot |C|$ of the pairs $i,j \in C$, we have $T_C(i,j) \neq M[i,j]$.*

In order to prove Lemma 3.2 we shall need the following technical claim.

Claim 3.3 *Let $q \leq n$ be an integer, Q a $q \times q$ real valued matrix and $0 \leq \phi, \theta \leq 1$. Suppose that for at least $q - \phi n$ of the rows i in Q, there exists a value r_i such that for at least $q - \theta n$ of the* *entries $Q[i,j]$ we have $Q[i,j] = r_i$, and that an analogous claim holds for the columns of Q. Then there exists a single value, denoted val, such that for all but at most $(2\phi + 3\theta)n \cdot q$ of the entries $Q[i,j]$, we have $Q[i,j] = val$.*

Proof: For each $i \in \{1, \ldots, q\}$, let r_i (c_i) denote the most common value in the i'th row (column) in Q. We say that a row (column) is *θ-pure* if all but at most θn of the entries in the row (column), have value r_i (c_i). By the premise of the lemma, all but at most ϕn of the rows (columns) are θ-pure. We say that entry $Q[i,j]$ is *row-representative*, if row i is θ-pure and $Q[i,j] = r_i$. Define a *column-representative* entry analogously. Then the number of entries that are *not* row-representative is at most $\phi n \cdot q + q \cdot \theta n = (\phi + \theta) \cdot q \cdot n$. Similarly, the number of entries that are not column-representative is at most $(\phi + \theta) \cdot q \cdot n$. Hence, the number of entries that are both row-representative and column-representative is at least $q^2 - 2(\phi + \theta) \cdot q \cdot n$.

Now, consider any row i that has at least $q - 2(\phi + \theta) \cdot n$ entries that are both row-representative and column-representative. Such a row must exists since this is the average number per row. Then the total number of entries in Q that do not have value r_i is at most $2(\phi + \theta) \cdot n \cdot q + q \cdot \theta n = (2\phi + 3\theta)n \cdot q$ (where the first term is due to all entries in columns j such that $Q[i,j]$ is either not row-representative or not column-representative, and the second term is due to entries that are not column-representative for columns j such that $Q[i,j]$ is both row-representative and column-representative). ■

Proof of Lemma 3.2: Consider the sub-matrix M_C of M that is induced by the class C, and let $i \in C$ be a point that is not an α-effective separator with respect to U. By definition, this means that in the i'th row of M_C, the number of pairs of entries that contain a different value is at most $(\alpha n)^2$. We claim that this implies that there exists a value r_i such that for at least $q - \alpha n$ of the entries $M_C[i,j]$ in the i'th row we have $M_C[i,j] = r_i$ (and that an analogous statement holds for the columns of M_C). The claim trivially holds for $q \leq \alpha n$, thus let $q > \alpha n$. Assume, contrary to the claim, that for every value in the i'th row there are less than $q - \alpha n$ entries with that value. Then for each $1 \leq j \leq q$, there are more than $q - (q - \alpha n) = \alpha n$ entries $M_C[i,\ell]$ such that $M_C[i,j] \neq M_C[i,\ell]$. Hence the total number of such pairs of different entries in row i is greater than $q \cdot \alpha n$ which is greater than $(\alpha n)^2$, contradicting our initial assumption on the i'th row.

Since there are at most βn points in C that are α-effective separators, we can apply Claim 3.3 with $Q = M_C$, $\phi = \beta$, and $\theta = \alpha$. Thus, there exists a value val, such that for all but at most $(2\beta + 3\alpha)n \cdot |C|$ of the entries in M_C we have $M_C[i,j] = val$. Note that since we assume that $M[i,j] > 0$ for every $i \neq j$ then $val > 0$. Define the sub-tree T_C to be a star-shaped tree, whose leaves are the points in C, and the distance of each leaf from the root of T_C is $val/2$. The lemma follows. ■

Proof of Theorem 2: As noted previously, the correctness of the algorithm for an ultrametric M directly follows from the algorithm. We thus focus on the second part of the theorem, and assume that M is ϵ-far from being an ultrametric.

Let $\alpha = \frac{\epsilon}{12}$, and $\beta = \frac{\epsilon}{8}$. We view the sample U as being selected in $p = 1/\alpha^2$ *phases*, where in each phase an independent sample of $s' = \frac{\ln(6p)}{\beta}$ points is selected. If in any phase the sample contains a point that is inconsistent with the previously selected points, then clearly the algorithm will reject in Step 3 (as it will not be able to construct an ultrametric tree T_U that is consistent with M). Otherwise, consider any *fixed* phase for which the number of α-effective separators with respect to the sample selected so far is at least βn. The probability that none of these separators is selected is at most

$$(1-\beta)^{s'} < e^{-\beta s'} \leq \frac{1}{6 \cdot p} .$$

Thus, the probability that in *some* phase in which the number of α-effective separators (with respect to the sample selected so far), is at least βn, no separator is selected, is at most $1/6$.

At the start of the first phase (where no sample has yet been selected), the total number of non-separated and consistent pairs of points is $n(n-1)$. In each phase where an α-effective separator is selected, the number of non-separated and consistent pairs of points decreases by at least $(\alpha n)^2$. It follows that the number of such phases is bounded by $1/\alpha^2$. Hence, with probability at least $5/6$, after $p = 1/\alpha^2$ phases, either there is no tree T_U that is consistent with M on U, or there are at most βn α-effective separators with respect to U.

Claim. *Let $U \subset [n]$ be such that there exists an ultrametric tree T_U that is consistent with M on U, and the number of α-effective separators with respect to U is bounded by βn. If M is ϵ-far from being an ultrametric, then there are either more than $\frac{\epsilon}{4}n$ inconsistent points with respect to U, or more than $\frac{\epsilon}{4}n^2$ violating pairs with respect to U.*

The above claim implies that with probability at least $5/6$ over the choice of the $m = \frac{8}{\epsilon}$ pairs of points in Step 4 of the algorithm, the algorithm rejects either in Step 5 or in Step 6. Taking into account the probability of at most $1/6$ that the number of α-effective is greater than βn, the second part of the theorem follows.

Thus, to conclude the proof of the theorem we prove the claim. Assume, contrary to the claim, that there are *at most* $\frac{\epsilon}{4}n$ inconsistent points, and *at most* $\frac{\epsilon}{4}n^2$ violating pairs. We next show that we can then define an ultrametric M' that disagrees with M on at most ϵn^2 entries, thus contradicting the assumption on M.

1. For every $i, j \in U$: $M'[i,j] = D_U(i,j)$ $(= M[i,j])$. Similarly, for every $i \in U$ and $j \in \Gamma_U$: $M'[i,j] = D_U(i,j)$ $(= M[i,j])$.
2. For every $i, j \in \Gamma_U$:
   (a) If i and j are separated then $M'[i,j] = D_U(i,j)$. Hence, among these pairs, M' and M only differ on the violating pairs (that belong to different classes).
   (b) If i and j are non-separated then $M'[i,j] = \min\{D_U(i,j), T_C(i,j)\}$, where C is the class they both belong to and T_C is the tree guaranteed by Lemma 3.2. (Taking the minimum among the two values is essential in order that M' be an ultrametric.) Here M' may differ from M on violating pairs (that belong to a common class, for which $M[i,j] = T_C[i,j] > D_U(i,j)$), and on the at most $(2\beta + 3\alpha)n|C| \leq \frac{\epsilon}{2}n|C|$ pairs of points $i, j \in C$ such that $M[i,j] \neq T_C(i,j)$.
3. If either $i \notin \Gamma_U$ or $j \notin \Gamma_U$: then $M'[i,j] = D_U(i,j)$, which may differ from $M[i,j]$. Since there are at most $\frac{\epsilon}{4}n$ inconsistent points, among these pairs there are at most $\frac{\epsilon}{4}n^2$ on which M' and M differ.

The total number of entries on which M' and M differ is hence at most

$$\frac{\epsilon}{4}n^2 + \sum_{C \in \mathcal{P}_U} \frac{\epsilon}{2}n|C| + \frac{\epsilon}{4}n^2 \quad \leq \epsilon n^2$$

where the first term is due to the violating pairs, the second term is due to those pairs i, j that belong to the same class C but for which $M[i,j] \neq T_C(i,j)$, and the third term is due to the pairs containing inconsistent points. ■

3.3 Constructing Almost Consistent Ultrametric Trees

Suppose that M is an ultrametric. Then our analysis can be used to imply that with high probability we can construct in time $n \cdot \text{poly}(1/\epsilon)$ an ultrametric tree T' that disagrees with M on at most an ϵ-fraction of its entries. Details follow.

By definition, if M is an ultrametric, then for every subset $U \subseteq [n]$, all points in $[n] \setminus U$ are consistent with T_U, and all pairs of points are non-violating. Note that given a set U, we can partition all points in $[n] \setminus U$ into the classes of the partition $\mathcal{P}_U$ in time $O(n \cdot |U|)$. As argued in the proof of Theorem 2, with high probability over the choice of U (where $|U| = \tilde{\Theta}(1/\epsilon^3)$), there are at most βn α-effective separators with respect to U (where α and β are as in the proof of the theorem). By Lemma 3.2, this implies that for every class C there exists a star shaped (sub-)tree T_C such that for at most $(2\beta + 3\alpha)n \cdot |C|$ of the pairs $i, j \in C$, we have $T_C(i,j) \neq M[i,j]$. By sampling from each class we can find, with high probability, the height of the star-shaped tree T_C and construct it. Following the argument in the proof of Theorem 2, it can be shown that the resulting tree disagrees with M on at most ϵn^2 entries.

Similar results can be obtained in the same manner for the other properties we test.

4 Testing Approximate Ultrametrics

In this section we extend the results from Section 3 to testing *approximate* ultrametrics. Namely, here we relax the condition of acceptance to matrices M that may not be exactly ultrametrics, but that are *close in the L_∞ norm* to an ultrametric.

Definition 4.1 (δ-Ultrametrics) *A matrix M is a δ-*ultrametric *if there exists an ultrametric M' such that $\|M - M'\|_\infty \leq \delta$.*

Below we describe a testing algorithm that for any given matrix M and parameters δ and ϵ, accepts M if it is a δ-ultrametric, and rejects M with probability at least $2/3$ if it is ϵ-far from any $c\delta$-ultrametric (for some fixed constant c). The structure of the algorithm and its analysis are similar to those for the exact case ($\delta = 0$). The algorithm tries to find evidence to M not being a δ-ultrametric. As in the exact case, showing that every δ-ultrametric passes the test will be relatively easy (though not as straightforward). Showing that a matrix M that is ϵ-far from any $c\delta$-ultrametric is rejected with high probability, will follow the same lines as in the exact case, but will be somewhat more involved.

We start by adapting the definitions from the exact case.

Definition 4.2 (δ-Consistent) *An ultrametric tree T_U is δ-*consistent *with a matrix M on U, if for every $i, j \in U$, $|T_U(i,j) - M[i,j]| \leq \delta$. In case $U = [n]$, we simply say that T is δ-consistent with M.*

Farach, Kannan and Warnow [12] give a polynomial-time algorithm for constructing a tree T that is δ-consistent with a given δ-ultrametric M.

Definition 4.3 (η-Consistent point) *Let T_U be an ultrametric tree that is δ-consistent with an $n \times n$ matrix M on $U \subseteq [n]$. We say that point $j \notin U$ is* η-consistent *with T_U if the following holds. Let T be the tree resulting from adding j to T_U by applying the procedure Add-Point(j, T_U, M) (described in Subsection 3.1). Then we ask that for every $k \in U$, $|T(j,k) - M[j,k]| \leq \eta$. Let Γ_U^η denote the set of all η-consistent points in $[n] \setminus U$.*

Definition 4.4 (λ-Separators) *Let M be an $n \times n$ matrix and $i, j \in [n]$. A point $k \in [n]$ is called a* λ-separator *for i and j if $|M[i,k] - M[j,k]| > \lambda$.*

If i and j have a λ-separator in the set U, then they are λ-separated *by U.*

Definition 4.5 (Effective Separators) *We say that a point $k \in [n] \setminus U$ is an* (α, λ)-effective separator *with respect to $U \subset [n]$, if the number of pairs of points in $[n] \setminus U$ that are λ-separated by $U \cup \{k\}$, and are* **not** *λ-separated by U, is at least $(\alpha n)^2$.*

Definition 4.6 (Violating pairs) *Let M be an $n \times n$ matrix and $i, j \in [n] \setminus U$. We say that i and j are a* violating pair *with respect to $U \subset [n]$, if either:*

1. *There exists a 2δ-separator $k \in U$ such that $|M[i,j] - \max\{M[i,k], M[j,k]\}| > 2\delta$;*
2. *For some $k \in U$ (that is not necessarily a 2δ-separator), $M[i,j] > \max\{M[i,k], M[j,k]\}| + 2\delta$.*

Algorithm 2 Testing Algorithm for Approximate Ultrametrics

1. *Uniformly and independently select $s = \Theta(\log(1/\epsilon)/\epsilon^3)$ points in $[n]$. Denote the set of points selected by U.*
2. *Construct a scaffold tree T_U that is δ-consistent with M on U using the algorithm in [12]. If this is not possible –* reject.
3. *Uniformly and independently select $m = \Theta(1/\epsilon^2)$ pairs of points in $[n]$.*
4. *For each point selected in Step 3, check whether it is 3δ-consistent with T_U. If not –* reject.
5. *For each pair of points selected in Step 3, check whether they are a violating pair, and if so –* reject.
6. *If no step caused rejection then* accept.

Theorem 4 *Algorithm 2 accepts every matrix M that is a δ-ultrametric, and rejects with probability at least $2/3$ any M that is ϵ-far from being a $c\delta$-ultrametric for some fixed constant c.*

The constant c that our analysis implies, is 84. However, we believe that a tighter analysis is possible.

Corollary 5 *Let the "natural" testing algorithm be the algorithm that simply selects a uniform sample of $\tilde{O}(1/\epsilon^3)$ points from $[n]$ and accepts if and only if it is possible to construct a tree that is δ-consistent with M on the sample. Then this algorithm accepts every matrix M that is a δ-ultrametric, and rejects with probability at least $2/3$ any M that is ϵ-far from being a $c\delta$-ultrametric for some fixed constant c.*

We shall prove Theorem 4 via a sequence of lemmas. The first two lemmas are used to prove the first part of the theorem, and the remaining lemmas to prove the second part of the theorem.

Lemma 4.1 *Let M be a δ-ultrametric, and let T_U be an ultrametric tree that is δ-consistent with M on $U \subseteq [n]$. Then every point $j \notin U$ is 3δ-consistent with T_U.*

Proof: Since M is a δ-ultrametric, there exists an ultrametric M' such that $\|M - M'\|_\infty \leq \delta$. Let T be the tree resulting from adding j to T_U by applying the procedure Add-Point(j, T_U, M). Let k be the point in U for which $M[k,j]$ is minimized, so that $T(k,j) = M[k,j]$. Note that for every $i \in U$, $T(j,i) \geq T(k,i)$. We thus need to consider three cases concerning the relations between the pairwise distances of i, j, and k in M', and for each there are two subcases depending on the pairwise distances according to T.

1. $M'[j,i] = M'[k,i] \geq M'[k,j]$:
   Since $\|M - M'\|_\infty \leq \delta$,

   $$M[j,i] \leq M'[j,i] + \delta = M'[k,i] + \delta \leq M[k,i] + 2\delta$$

   In a similar way it is possible to show that $M[j,i] \geq M[k,i] - 2\delta$.

   (a) $T(j,i) = T(k,i) \geq T(k,j)$:
   Recall that T is δ-consistent with M. Thus,

   $$T(j,i) = T(k,i) \leq M[k,i] + \delta \leq M[j,i] + 3\delta$$

   In a similar way it is possible to show that $T(j,i) \geq M[j,i] - 3\delta$.

   (b) $T(j,i) = T(k,j) \geq T(k,i)$:
   Since k is the nearest point to j in U, we have $M[k,j] \leq M[j,i]$. Thus,

   $$T(j,i) = T(k,j) = M[k,j] \leq M[j,i].$$

   On the other hand,

   $$T(j,i) \geq T(k,i) \geq M[k,i] - \delta \geq M[j,i] - 3\delta.$$

2. $M'[j,i] = M'[k,j] \geq M'[k,i]$:
   In this case it is possible to bound $M[j,i]$ as follows,

   $$M[k,j] - 2\delta \leq M[j,i] \leq M[k,j] + 2\delta$$

   (a) $T(j,i) = T(k,i) \geq T(k,j)$:

   $$T(j,i) = T(k,i) \leq M[k,i] + \delta \leq M'[k,i] + 2\delta \leq M'[j,i] + 2\delta \leq M[j,i] + 3\delta$$

   On the Other hand,

   $$T(j,i) \geq T(k,j) = M[k,j] \geq M[j,i] - 2\delta$$

(b) $T(j,i) = T(k,j) \geq T(k,i)$:
We have $T(j,i) = T(k,j) = M[k,j]$. Thus,

$$M[j,i] - 2\delta \leq M[k,j] = T(j,i) = M[k,j] \leq M[j,i]$$

3. $M'[k,i] = M'[k,j] \geq M'[j,i]$:
Here we can bound $M[j,i]$ as follows,

$$M[k,j] \leq M[j,i] \leq M'[j,i] + \delta \leq M'[k,j] + \delta \leq M[k,j] + 2\delta$$

We also need the following inequality,

$$M[k,i] \leq M'[k,i] + \delta = M'[k,j] + \delta \leq M[k,j] + 2\delta$$

(a) $T(j,i) = T(k,i) \geq T(k,j)$:

$$T(j,i) = T(k,i) \leq M[k,i] + \delta \leq \mathrm{M}[k,j] + 3\delta \leq M[j,i] + 3\delta$$

$$T(j,i) \geq T(k,j) = M[k,j] \geq M[j,i] - 3\delta$$

(b) $T(j,i) = T(k,j) \geq T(k,i)$: Identical to case 2b.

■

Lemma 4.2 *Let M be a δ-ultrametric. Then for every pair of points $i,j \in [n]$, and for every point $k \in [n]$, if k is a 2δ-separator for i and j then $|M[i,j] - \max\{M[i,k], M[j,k]\}| \leq 2\delta$, and otherwise, $M[i,j] \leq \max\{M[i,k], M[j,k]\} + 2\delta$.*

Proof: Since M is a δ-ultrametric, there exists an ultrametric M' such that $\|M - M'\|_\infty \leq \delta$. Therefore, for every $k \in [n]$, $|M'[i,k] - M[i,k]| \leq \delta$, and $|M'[j,k] - M[j,k]| \leq \delta$.

In particular, this is true of any 2δ-separating point k (of i and j). For such a point k, we thus have that $M'[i,k] \neq M'[j,k]$, and since M' is an ultrametric, $M'[i,j] = \max\{M'[i,k], M'[j,k]\}$, implying that

$$\max\{M[i,k], M[j,k]\} - 2\delta \leq M[i,j] \leq \max\{M[i,k], M[j,k]\} + 2\delta$$

In case k is not a 2δ-separator, then by the Three-Point Condition (Fact 1),

$$M'[i,j] \leq \max\{M'[i,k], M'[j,k]\}$$

and so

$$M[i,j] \leq M'[i,j] + \delta \leq \max\{M[i,k], M[j,k]\} + 2\delta$$

■

It directly follows from Lemma 4.1 and Lemma 4.2, that if M is a δ-ultrametric then it always passes the test.

We now continue with the more involved part of Theorem 4, that is, of proving that any M that is ϵ-far from being an $O(\delta)$-ultrametric is rejected with probability at least 2/3. More precisely, we prove a sequence of claims from which the contrapositive statement will follow.

The first lemma deals with pairs of points that are consistent with T_U, are not violating, and are "well separated" by U. Such pairs are analogous to consistent, non-violating pairs of points that belong to different classes in the exact case. Here the distance D_U is as defined in the exact case given the tree T_U (Definition 3.6).

Lemma 4.3 *Let M be an $n \times n$ matrix, let T_U be an ultrametric tree that is δ-consistent with M on $U \subseteq [n]$, and let $i, j \in \Gamma_U^{3\delta}$. If there exists a point $k \in U$ that is a 6δ-separator for i and j, and i and j are not violating with respect to U, then $|D_U(i,j) - M[i,j]| \leq 5\delta$.*

Proof: Since i and j are both 3δ-consistent with T_U (and D_U is an extension of the distance defined by T_U), we have that $|M[i,k] - D_U(i,k)| \leq 3\delta$, and similarly $|M[j,k] - D_U(j,k)| \leq 3\delta$. Since k is a 6δ-separator for i and j, $|M[i,k] - M[j,k]| > 6\delta$, and so $|D_U(i,k) - D_U(j,k)| > 0$. By Fact 1 we have that $D_U(i,j) = \max\{D_U(i,k), D_U(j,k)\}$. But since i and j are not violating (and k is a 2δ-separator for the pair), $|M[i,j] - \max\{M[i,k], M[j,k]\}| \leq 2\delta$. Therefore,

$$D_U(i,j) = \max\{D_U(i,k), D_U(j,k)\} \leq \max\{M[i,k] + 3\delta, M[j,k] + 3\delta\} \leq M[i,j] + 5\delta$$

and similarly $\max\{D_U(i,k), D_U(j,k)\} \geq M[i,j] - 5\delta$. ■

The next lemma is analogous to Lemma 3.2 which dealt with distances between points that belong to the same class.

Lemma 4.4 *Let $S \subseteq [n] \setminus U$ be such that there are at most βn points in S that are (α, λ)-effective separators. Then there exists an ultrametric (star shaped) tree T_S such that for at most $(2\beta + 6\alpha)n \cdot |S|$ of the pairs $i, j \in S$, we have $|T_S(i,j) - M[i,j]| > \frac{3}{2}\lambda$.*

In order to prove the lemma we shall first prove the following claim.

Claim 4.5 *Let $q \leq n$ be an integer, Q a $q \times q$ real valued matrix, $0 \leq \phi, \theta < 1/2$, and $\mu \geq 0$. Suppose that for at least $q - \phi n$ of the rows i in Q, there exists a value r_i such that for at least $q - \theta n$ of the entries $Q[i,j]$ we have $|Q[i,j] - r_i| \leq \mu$, and that an analogous claim holds for the columns of Q. Then there exists a single value t, such that for all but at most $(2\phi + 3\theta)n \cdot q$ of the entries $Q[i,j]$, we have $|Q[i,j] - t| \leq 3\mu$.*

Proof: The proof of Claim 4.5 follows the same lines as the proof of Claim 3.3. Here we say that a row i is *dense* if there exists a value r_i such that for at least $q - \theta n$ of the entries $Q[i,j]$ we have $|Q[i,j] - r_i| \leq \mu$. We similarly define dense columns. We say that an entry $Q[i,j]$ is *row-representative* if its row is dense and $|Q[i,j] - r_i| \leq \mu$. We similarly defined *column-representative* entries. Then, similarly to the proof of Claim 3.3, we obtain that all but at most $2(\phi + \theta)n \cdot q$ of the entries in Q are both row-representative and column-representative.

We then look at a row i that contains at least $q - 2(\phi + \theta)n$ entries that are both row-representative and column-representative. For each such entry $Q[i,j]$, we have that $|Q[i,j] - r_i| \leq \mu$. We also know that for all but at most θn of the entries $Q[k,j]$ (in the j'th column), $|Q[i,j] - Q[k,j]| \leq 2\mu$. Hence, for all but at most $2(\phi + \theta)n \cdot q + q \cdot \theta n = (2\phi + 3\theta)n \cdot q$ of the entries $Q[k,j]$ we have $|Q[k,j] - r_i| \leq 3\mu$. ■

Proof of Lemma 4.4: Let M_S denote the submatrix of M induced by S. Consider any $i \in S$ that is not an (α, λ)-effective separator. We claim that there exists a value r_i such that in the row (and similarly the column) that corresponds to i there at most $2\alpha n$ entries $M_S[i,j]$ such that $|M_S[i,j] - r_i| > \lambda/2$. To see why this is true, let us order the entries in the i'th row in increasing order. Assume for simplicity that all entries are distinct (the argument can be easily modified to work with non distinct values). Consider the first entry $M_S[i,j]$ in this order such that there are exactly αn entries that are smaller than $M_S[i,j]$. Then there must be at most αn entries that are larger by more than λ from $M_S[i,j]$ (otherwise, there would be more than $(\alpha n)^2$ pairs j, ℓ such that $|M[i,j] - M[i,\ell]| > \lambda$, and thus i would be an (α, λ)-effective separator). But this implies that for $r_i = M_S[i,j] + \lambda/2$, there are at most $2\alpha n$ entries $M_S[i,\ell]$ such that $|M_S[i,\ell] - t_i| > \lambda/2$.

The corollary follows by applying Claim 4.5 with $Q = M_S$, $\phi = \beta$, $\theta = 2\alpha$, and $\mu = \lambda/2$, . ■

At this point we slightly depart from the structure of the analysis in the exact case. We shall need the following definition.

Definition 4.7 (Incorrect Points) *Let $U \subseteq [n]$ be such that there exists an ultrametric tree T_U that is δ-consistent with M on U (and so, in particular, D_U is well defined). Let $i \notin U$, and define:*

$$B_i^{\lambda_1,\lambda_2} \stackrel{\text{def}}{=} \{j \notin U : \textit{ there is no } \lambda_1\textit{-separator for } j \textit{ and } i \textit{ in } U, \textit{ and } M[i,j] < D_U(i,j) - \lambda_2\}. \quad (2)$$

Then point i is $(\alpha, \lambda_1, \lambda_2)$-incorrect *with respect to D_U, if $\left|B_i^{\lambda_1,\lambda_2}\right| > \alpha n$.*

Roughly speaking, a point i is incorrect with respect to D_U if there are many points j (from which i is not separated by U) such that $M[i,j]$ differs significantly from $D_U(i,j)$, and in particular, is smaller. (We note that when $M[i,j]$ is significantly larger, then i and j are a violating pair.)

We now show that if the number of inconsistent points, violating pairs and incorrect points is small, then M is ϵ-close to an approximate ultrametric for the appropriate constants.

Lemma 4.6 *Let U be such that there exists an ultrametric tree T_U that is δ-consistent with M on U, and furthermore, there are at most $\frac{\epsilon}{4}n$ points that are not 3δ-consistent with T_U and at most $\frac{\epsilon}{4}n^2$ pairs of of violating points with respect to T_U. If the number of $(\frac{\epsilon}{4}, \lambda_1, \lambda_2)$-incorrect points in $[n] \setminus U$ is at most $\frac{\epsilon}{4}n$, where $\lambda_1 \geq 6\delta$, then M is ϵ-close to being a $\max\{\lambda_1, \lambda_2\}$-ultrametric.*

Proof: We show that on all but at most ϵn^2 pairs of points i, j, we have $|M[i,j] - D_U(i,j)| \leq \max\{\lambda_1, \lambda_2\}$. Since D_U is determined by an ultrametric tree, the lemma follows.

Let A denote the set of $(\frac{\epsilon}{4}, \lambda_1, \lambda_2)$-incorrect points in $[n] \setminus U$, and for each $i \in A$, let $B_i = B_i^{\lambda_1,\lambda_2}$. Let us go over all pairs i, j:

1. For every pair $i, j \in U$: $|M[i,j] - D_U(i,j)| = |M[i,j] - T_U(i,j)| \leq \delta$.
   Similarly, for every $i \in U$, $j \in \Gamma_U^{3\delta}$: $|M[i,j] - D_U(i,j)| = |M[i,j] - T_U(i,j)| \leq 3\delta$.
2. For every pair of points $i, j \in \Gamma_U^{3\delta}$ that are not violating and $j \notin B_i$:
   (a) If i and j are λ_1-separated by U: then by Lemma 4.3, $|M[i,j] - D_U(i,j)| \leq 5\delta$ (since $\lambda_1 \geq 6\delta$).
   (b) Otherwise: $D_U(i,j) - \lambda_2 \leq M[i,j] \leq D_U(i,j) + 5\delta$. The first inequality follows from the definition of B_i. For the second inequality, note that there exists a point k such that $D_U(i,j) = \max\{D_U(i,k), D_U(j,k)\}$ (since $i, j \notin U$). Therefore, since i and j are not violating and $i, j \in \Gamma_U^{3\delta}$, we get $M[i,j] \leq \max\{M[i,k], M[j,k]\} + 2\delta \leq \max\{D[i,k] + 3\delta, D[j,k] + 3\delta\} + 2\delta = D_U(i,j) + 5\delta$
3. For all other pairs, the difference between M and D_U might be larger, but we can bound their number as follows.

(a) The number of pairs i, j such that $i \notin \Gamma_U^{3\delta}$ is at most $\frac{\epsilon}{4}n^2$.
(b) The number of violating pairs with respect to U is at most $\frac{\epsilon}{4}n^2$.
(c) The number of pairs $i \notin A$ and $j \in B_i$ is at most $\frac{\epsilon}{4}n^2$ (since for each $i \notin A$, $|B_i| \leq \frac{\epsilon}{4}n$).
(d) The number of pairs i, j such that $i \in A$, is at most $\frac{\epsilon}{4}n^2$.

■

Our algorithm only checks for inconsistent points and violating pairs of points. Therefore, we can not apply the above lemma as it is, but have to bound the number of incorrect points. In order to do so, we introduce the notion of *useful points*. As we shall see, the two types of points are related, and we are able to bound the number of incorrect points by bounding the number of useful points.

Definition 4.8 (Useful Points) *We say that a point $i \notin U$ is (α, λ)-useful with respect to U, if one of the following conditions holds:*

1. *There are at least $(\alpha n)^2$ pairs of points that are violating with respect to $U \cup \{i\}$.*
2. *Let $C_i \stackrel{\text{def}}{=} \{j : \forall k \in U,\ M[j,i] < M[j,k]\ \}$ be the set of points that are closer to i than to any point in U. Then there are at least $(\alpha n)^2$ pairs of points $j, \ell \in C_i$, such that $M[j,\ell] \geq \max\{M[j,i], M[\ell,i]\} - \lambda$, while for every $k \in U$, $M[j,\ell] < \max\{M[j,k], M[\ell,k]\} - \lambda$.*

Intuitively, a useful point is such that its addition to U either causes many violations, or actually brings D_U closer to M on many pairs of points (and so makes fewer points incorrect).

Lemma 4.7 *There exists constants $c_1, \cdots, c_7$ and $d_1, \cdots, d_4$ such that $c_6, c_7 \leq \frac{1}{4}$ and $d_3 \geq 6\delta$, for which the following holds. Let U be such that there exists an ultrametric tree T_U that is δ-consistent with M and furthermore:*

- *The number of $(c_1\epsilon, d_1\delta)$-effective separators with respect to U is at most $c_2\epsilon n$;*
- *The number of points that are not 3δ-consistent with respect to U is at most $c_3\epsilon n$.*

If the number of $(c_4\epsilon, d_2\delta)$-useful points with respect to U is less than $c_5\epsilon n$, then the number of $(c_6\epsilon, d_3\delta, d_4\delta)$-incorrect points with respect D_U is at most $c_7\epsilon n$.

Proof: Assume, contrary to the claim, that the number of $(c_6\epsilon, d_3\delta, d_4\delta)$-incorrect points with respect to D_U is greater than $c_7\epsilon n$. We show that the number of $(c_4\epsilon, d_2\delta)$-useful points with respect to U is at least $c_5\epsilon n$, in contradiction to the premise of the lemma. For ease of the presentation, we sometimes drop the parameters, and simply refer to incorrect and useful points. Along the way we introduce constraints on the relations between the different constants $c_1, \cdots, c_7$ and $d_1, \cdots, d_4$. At the end of the proof we verify that all these constraints can be satisfied simultaneously.

Let A denote the set of incorrect points that are 3δ-consistent with T_U. The number of such points is at least $(c_7 - c_3)\epsilon n \geq c_5\epsilon n$. For each point $i \in A$, we show that either i is useful, or there exist at least $c_5\epsilon n$ other useful points (that are close to i).

We start by making several observations concerning each $i \in A$. Consider the set $B_i = B_i^{d_3\delta, d_4\delta}$ as defined in Equation (2). For each $j \in B_i$, consider the point $k \in U$ that is closest to j, so that

$D_U(j,k) = M[j,k]$. Since i is 3δ-consistent with T_U, we also have that $|D_U(i,k) - M[i,k]| \leq 3\delta$. By definition of D_U, $D_U(i,j) = \max\{D_U(i,k), D_U(j,k)\}$ and so $|D_U(i,j) - \max\{M[j,k], M[i,k]\}| \leq 3\delta$. Now, by definition of B_i, we have that $|M[i,k] - M[j,k]| \leq d_3\delta$ and so $|D_U(i,j) - M[j,k]| \leq (d_3+3)\delta$. Since (again by definition of B_i), $M[i,j] < D_U(i,j) - d_4\delta$, we obtain that for every $j \in B_i$ and $k \in U$,

$$M[i,j] < M[j,k] - (d_4 - d_3 - 3)\delta \tag{3}$$

Furthermore, since for every $j \in B_i$, there is no $d_3\delta$-separator for i and j in U, then for every pair $j, \ell \in B_i$ there is no $2d_3\delta$-separator in U. Let us apply Lemma 4.4 using the fact that the number of $(c_1\epsilon, d_1\delta)$-effective separators is at most $c_2\epsilon n$. If we set $d_1 = 2d_3$, we obtain that there exists a star-shaped tree T_{B_i}, such that for all but at most $(6c_1\epsilon + 2c_2\epsilon) \cdot n \cdot |B_i|$ of the pairs of points $j, \ell \in B_i$,

$$|M[j,\ell] - 2h(T_{B_i})| \leq 3d_3\delta,$$

where $h(T_{B_i})$ is the height of T_{B_i}. We say that such pairs are *representative* with respect to B_i. Since $|B_i| \geq c_6\epsilon n$, if $(6c_1 + 2c_2) < c_6/32$, then the number of non-representative pairs is at most $\frac{1}{32}|B_i|^2$.

Let $\hat{B}_i \stackrel{\text{def}}{=} \{j \in B_i :\ M[i,j] < 2h(T_{B_i}) - 3d_3\delta - 2\delta\}$. Roughly speaking, $\hat{B}_i$ is the subset of points in B_i that are significantly closer to i than to each other. We consider two cases.

1. $|\hat{B}_i| \geq \frac{1}{2}|B_i|$: Then for every representative pair $j, \ell \in \hat{B}_i$,

$$\begin{aligned} M[j,\ell] &\geq 2h(T_{B_i}) - 3d_3\delta \\ &> \max\{M[j,i], M[\ell,i]\} + 2\delta \end{aligned}$$

   That is, j and ℓ are a violating pair with respect to $U \cup \{i\}$. The number of such pairs is at least

$$|\hat{B}_i|^2 - \frac{1}{32}|B_i|^2 \geq \frac{1}{4}|B_i|^2 - \frac{1}{32}|B_i|^2 > \frac{1}{5}|B_i|^2 \geq \frac{1}{5}(c_6\epsilon n)^2 \geq (c_4\epsilon n)^2,$$

   where the last inequality is correct if $c_4^2 \leq c_6^2/5$. Thus, the point i is useful (of the first type).

2. $|\hat{B}_i| < \frac{1}{2}|B_i|$. Let $\tilde{B}_i \stackrel{\text{def}}{=} B_i \setminus \hat{B}_i$, so that $|\tilde{B}_i| > \frac{1}{2}|B_i|$. In this case, for every representative pair $j, \ell \in \tilde{B}_i$,

$$\begin{aligned} M[j,\ell] &\leq 2h(T_{B_i}) + 3d_3\delta \\ &\leq \min\{M[j,i], M[\ell,i]\} + (6d_3 + 2)\delta \end{aligned}$$

   By Equation (3), for every such pair, and for every $k \in U$,

$$M[j,\ell] \leq \min\{M[j,k], M[\ell,k]\} + (7d_3 + 5 - d_4)\delta \tag{4}$$

   Let

$$\tilde{B}_i^j \stackrel{\text{def}}{=} \{\ell \in \tilde{B}_i :\ j \text{ and } \ell \text{ are a representative pair }\}.$$

   We say that a point $j \in \tilde{B}_i$ is a *good partner* with respect to $\tilde{B}_i$, if $|\tilde{B}_i^j| > \frac{3}{4}|\tilde{B}_i|$. By a simple counting argument (using the fact that the number of non-representative pairs is at most $\frac{1}{32}|B_i|^2$), we get that the number of good partners in $\tilde{B}_i$ is at least $\frac{1}{2}|\tilde{B}_i| \geq \frac{1}{4}|B_i| \geq c_5\epsilon n$. We now show that every good partner $j \in \tilde{B}_i$ is useful (of the second type).

Consider any point $\ell \in \tilde{B}_i^j$. By Equation (4) for every $k \in U$, $M[j,\ell] < M[\ell,k]+(7d_3+5-d_4)\delta$. Hence, if $d_4 > 7d_3+5$, then for every $k \in U$, $M[j,\ell] < M[\ell,k]$. Therefore, all points $\ell \in \tilde{B}_i^j$, are closer to j than to any point in U.

Furthermore, if $d_4 > (7d_3+5+d_2)$, then for every $k \in U$, and every representative pair $\ell,\ell' \in \tilde{B}_i^j$, $M[\ell,\ell'] < \max\{M[\ell,k], M[\ell',k]\} - d_2\delta$. On the other hand, for every such pair (by definition of representative pairs), $M[\ell,\ell'] \geq \max\{M[\ell,j], M[\ell',j]\} - 6d_3\delta$, and so for $d_3 \leq d_2/6$, we have $M[\ell,\ell'] \geq \max\{M[\ell,j], M[\ell',j]\} - d_2\delta$. The number of representative pairs in $\tilde{B}_i^j$ is at least

$$|\tilde{B}_i^j|^2 - \frac{1}{32}|B_i|^2 > \left(\frac{3}{4}\right)^2 |\tilde{B}_i|^2 - \frac{1}{32}|B_i|^2 > \frac{9}{16}\cdot\frac{1}{4}|B_i|^2 - \frac{1}{32}|B_i|^2 > \frac{1}{10}(c_6\epsilon n)^2 \geq (c_4\epsilon n)^2,$$

if $c_4^2 \leq c_6^2/10$. Therefore, j is a useful point (of the second type).

In order to finish the proof, we go over all constraints introduced above, and check that there exists a consistent setting of the constants. We have the following constraints:

- $c_5 \leq c_7 - c_3$, $(2c_2+6c_1) < c_6/32$, $c_5 \leq c_6/4$, $c_4^2 \leq c_6^2/10$.
- $d_1 = 2d_3$, $d_4 > (7d_3+5+d_2)$, $d_2 \geq 6d_3$.

We set:

- $c_6, c_7 = 2^{-2}$, $c_5, c_4, c_3 = 2^{-4}$, $c_2 = 2^{-7}$, $c_1 = 2^{-10}$.
- $d_3 = 6$, $d_1 = 12$, $d_2 = 36$, $d_4 = 84$.

■

Proof of Theorem 4: The proof of the theorem will follow similar lines to those of the proof of Theorem 2. If M is a δ-ultrametric, then by Lemma 4.1 and Lemma 4.2 it always passes the test. We thus turn to the second part of the theorem.

As in the proof of Theorem 2, we view U as being selected in phases. Here too there are $p = \Theta(1/\epsilon^2)$ phases, and in each phase, $s' = \Theta(\ln p/\epsilon)$ points are selected. In what follows, all constants are as in Lemma 4.7. Similarly to what was argued in the proof of Theorem 2, as long as the number of $(c_1\epsilon, d_1\delta)$-effective separators is at least $c_2\epsilon n$, or the number of $(c_4\epsilon, d_2\delta)$-useful points is at least $c_5\epsilon n$, either an effective separator or a useful point will be selected in the next phase with high probability. If a useful point that creates at least $(c_4\epsilon n)^2$ violations is selected (that is, of the first type of useful points), then we are done, as the algorithm will reject with high probability in Step 5 of the algorithm. Otherwise, by the definitions of effective separators and of useful points, after at most $p = 1/(c_1\epsilon)^2 + 1/(c_4\epsilon)^2 = \Theta(1/\epsilon^2)$ phases in which either an effective separator or a useful point (of the second type) is selected, the number of effective separators must be less than $c_2\epsilon n$, and the number of useful points (of the second type) must be less than $c_5\epsilon n$.

If there is no tree T_U that is δ-consistent with M on U, then the algorithm will reject in Step 2 of the algorithm. If such a tree is found in Step 2 but the number of points that are not 3δ-consistent with T_U is at least $c_3\epsilon n$, then with high probability the algorithm will reject in Step 4 of the algorithm. Otherwise, we can apply Lemma 4.7 and obtain that the number of $(c_6\epsilon, d_3\delta, d_4\delta)$-incorrect points with respect D_U is at most $c_7\epsilon n$. Hence, if M is ϵ-far from being a $d_4\delta$-ultrametric, then there must be at least $\frac{\epsilon}{4}n^2$ violating pairs, or else (since $c_6, c_7 \leq \frac{1}{4}$ and $d_3 \geq 6\delta$), we could apply Lemma 4.6 and obtain a contradiction. ■

5 Testing Tree Metrics

In this section we describe how to modify the testing algorithm for ultrametrics so that it can be applied to (general) tree metrics. We start with a definition of tree metrics.

Definition 5.1 (Tree Metrics) *We say that an $n \times n$ matrix M is a* tree metric *(or an* additive metric*), if there exists a tree T with positive weights on the edges, for which the following holds:*

1. *There exists a mapping ϕ from $[n]$ into the nodes of T.*
2. *All internal nodes in the tree to which no $i \in [n]$ is mapped to, have degree greater than 2.*
3. *For every $i, j \in [n]$, $T(\phi(i), \phi(j)) = M[i, j]$.*

In the above definition we allow ϕ to be many-to-one, so that M may actually be a pseudo-metric. However, with a slight abuse of terminology we refer to M as being a tree metric. In Section 7 we show that testing the stricter property, in which the embedding ϕ must be one-to-one, requires $\Omega(\sqrt{n})$ queries (for a constant ϵ).

We show:

Theorem 6 *There exists an algorithm for testing whether a matrix M is a tree metric. The algorithm takes a sample of size $O(\log(1/\epsilon)/\epsilon^3)$, and has query complexity and running time that are at most quadratic in the sample size.*

Constructing a Tree Metric. Similarly to the case of ultrametrics, there are efficient procedures for constructing a tree T that is consistent with a tree metric M. Furthermore, one of the known procedures [22] is iterative. For our purposes, the important aspect of this procedure is that when adding a point j to a tree T, there is a unique choice for the point of departure of j from T, and the distance of j to this point is also uniquely determined.

Consistent Tree and Point. The definition of a consistent tree (Definition 3.3) and of consistent points (Definition 3.4), are adapted to this case in a straightforward manner, and Γ_U denotes the set of points consistent with T_U.

The Scaffold Partition. Let $U \subset [n]$ be such that there exists a tree T_U that is consistent with M on U. Here we partition the points in Γ_U according to their points of departure from the scaffold T_U. Namely, two points in Γ_U belong to the same class in the partition $\mathcal{P}_U$ if and only if they have the same point of departure from T_U. Note that as opposed to the ultrametric case, two points in the same class may have different distances according to M from the points in U. As in the ultrametric case, if M is in fact an additive metric, then classes correspond to subtrees with respect to T_U.

The Scaffold Distance. We define the scaffold distance D_U similarly to the way it was defined for ultrametrics (Definition 3.6). In particular, for each point $i \in \Gamma_U$, let $d_U(i)$ be the distance between i and its point of departure from T_U. For all inconsistent points we may select an arbitrary point of departure and an arbitrary distance to this point. For completeness, for each $i \in U$, i itself is defined as its point of departure from T_U, and $d_U(i) = 0$. Then for every $i, j \in [n]$, define:

$$D_U(i,j) = d_U(i) + d_U(j) + T_U(p_i, p_j)$$

where p_i and p_j are the points of departure of i and j respectively, and $T_U(p_i, p_j)$ is their distance in the tree T_U. (Note that we slightly abuse notation, since p_i and p_j may not exist as nodes in the tree T_U).

Hence, here too if M is a tree metric, then for every pair of points $i, j \in \Gamma_U$ that belong to different classes in $\mathcal{P}_U$, $M[i,j] = D_U(i,j)$, and for every pair i, j that belong to the same class, $M[i,j] \leq D_U(i,j)$.

VIOLATING PAIRS. Violating pairs are defined the same as in the ultrametric case (Definition 3.7).

THE TESTING ALGORITHM. Testing general tree metrics is essentially the same as testing ultrametrics. Here too the algorithm selects a uniform sample U of $\tilde{O}(1/\epsilon^3)$ points, and tries to construct a tree T_U that is consistent with M on U. It then selects an additional sample of $O(1/\epsilon)$ pairs of points, and checks for inconsistent points and violating pairs. The required modifications in the analysis are provided below, and we start with the definition of separators.

SEPARATORS. Separated and non-separated pairs are defined as in the ultrametric case (Definition 3.8). The definition of separators is modified as follows.

Definition 5.2 (Separators) *Let U be such that there exists a tree T_U that is consistent with M on U. We say that a point $k \in \Gamma_U$ is a* separator *with respect to U for a non-separated pair of points $i, j \in \Gamma_U$, if either one of the following holds:*

1. *Both i and j are consistent with $\Gamma_{U \cup \{k\}}$ and they are separated with respect to $U \cup \{k\}$.*
2. *Either i or j is inconsistent with $\Gamma_{U \cup \{k\}}$.*

The definition of effective separators (Definition 3.10), remains as is (given the above definition of separators).

The main difference in the analysis of the algorithm is in the proof of a variant of Lemma 3.2 presented below.

Lemma 5.1 *Let $C \subseteq [n] \setminus U$ be a class in $\mathcal{P}_U$, and let p_C be the common point of departure of the points in C from T_U. If there are at most βn points in C that are α-effective separators with respect to U, then there exists a subtree T_C such that:*

1. *The root of T_C is the point p_C.*
2. *For at most $(3\beta + 4\alpha)n \cdot |C|$ of the pairs $i, j \in C$, we have $T_C(i,j) \neq M[i,j]$.*
3. *For each $i \in C$, we have $T_C(i, p_C) = d_U(i)$.*

Proof: For each pair of points $i, j \in C$, we say that i and j are *compatible* (with respect to U) if j is consistent with $T_{U \cup \{i\}}$ (which is equivalent to i being consistent with $T_{U \cup \{j\}}$). Otherwise, they are *incompatible*. Let Q be a $|C| \times |C|$ matrix that is defined as follows. For every compatible pair $i, j \in C$,

$$Q[i,j] \stackrel{\text{def}}{=} \frac{D_U(i,j) - M[i,j]}{2} = \frac{d_U(i) + d_U(j) - M[i,j]}{2} \tag{5}$$

For any incompatible pair, $Q[i,j] = 0$. What does $Q[i,j]$ mean? If $i, j \in C$ are compatible, then $Q[i,j]$ is simply the distance between p_C and the point of departure of j from $T_{U \cup \{i\}}$.

PROPERTIES OF Q. Observe that if i and j are compatible, then $0 \le Q[i,j] \le \min(d_U(i), d_U(j))$. It follows from the definition of separators that if $i \in C$ is not a separator (with respect to U) for $j, \ell \in C$, then both j and ℓ are compatible with i, and $Q[i,j] = Q[i,\ell]$.

Hence, if i is not an α-effective separator with respect to U, then the number of pairs of entries $Q[i,j] \neq Q[i,\ell]$ is at most $(\alpha n)^2$. Similarly to what we showed in the proof of Lemma 3.2, it follows that all but at most αn of the entries in the i'th row (column) in Q have the same value r_i.

We can now appeal to Claim 3.3 and obtain that all but at most $(2\beta + 3\alpha) \cdot n \cdot |C|$ of the entries in Q have the same value t. For each $i \in C$, let $d_i = d_U(i) - t$. By Equation (5), for every compatible pair $i, j \in C$:

$$M[i,j] = d_U(i) + d_U(j) - 2Q[i,j]$$

Thus, if i, j are compatible and $Q[i,j] = t$, then $M[i,j] = d_i + d_j$. Therefore, for all but at most $(2\beta + 3\alpha) \cdot n \cdot |C|$ of the compatible pairs $i, j \in C$, we have $M[i,j] = d_i + d_j$.

DEFINING T_C. Intuitively, we would now like to simply set $T_C(i,j) = d_i + d_j$. In this case the subtree T_C will be a star, such that the center of the star is connected by an edge of length t to the point p_C. Each point $i \in C$ is a leaf connected by an edge of length d_i to the center of the star.

However, the difficulty with this definition is that some of the d_i's may be negative. To address this issue, we do the following.

1. For each pair $i, j \in C$ such that $d_i \ge 0$ and $d_j \ge 0$: set $T_C(i,j) = d_i + d_j$.
2. If either $d_i < 0$ or $d_j < 0$: set $T_C(i,j) = d_U(i) + d_U(j)$.

The subtree T_C corresponds to a tree that consists of two stars with an edge of length t connecting the centers of these stars. Every i for which $d_i \ge 0$ is a leaf of the first star and is connected to the center of this star with an edge of length d_i. (If $d_i = 0$ then i resides at the center of the star.) Every i for which $d_i < 0$ is a leaf of the second star, and is connected to the center of this star by an edge of length $d_U(i)$. The center of the second star is the point p_C. See Figure 3.

BOUNDING THE DIFFERENCES BETWEEN M AND T_C. We now count the pairs on which M and T_C differ.

1. Compatible pairs i, j for which $d_i \ge 0$ and $d_j \ge 0$: In this case, for compatible pairs i, j for which $Q[i,j] = t$, we have $T_C(i,j) = d_i + d_j = M[i,j]$. As discussed above, the number of compatible pairs i, j such that $Q[i,j] \neq t$ is at most $(2\beta + 3\alpha) \cdot n \cdot |C|$. Thus, in this case, for all but at most $(2\beta + 3\alpha) \cdot n \cdot |C|$ of the compatible pairs $i, j \in C$, $T_C(i,j) = M[i,j]$.

2. Compatible pairs i, j for which $d_i < 0$: Recall that for each compatible pair $i, j \in C$, $Q[i,j] \le \min(d_U(i), d_U(j))$. Thus, if $d_i = d_U(i) - t < 0$, then $d_U(i) < t$, and so $Q[i,j] < t$. In particular $Q[i,j] \neq t$, and so we already counted these pairs in Item 1.

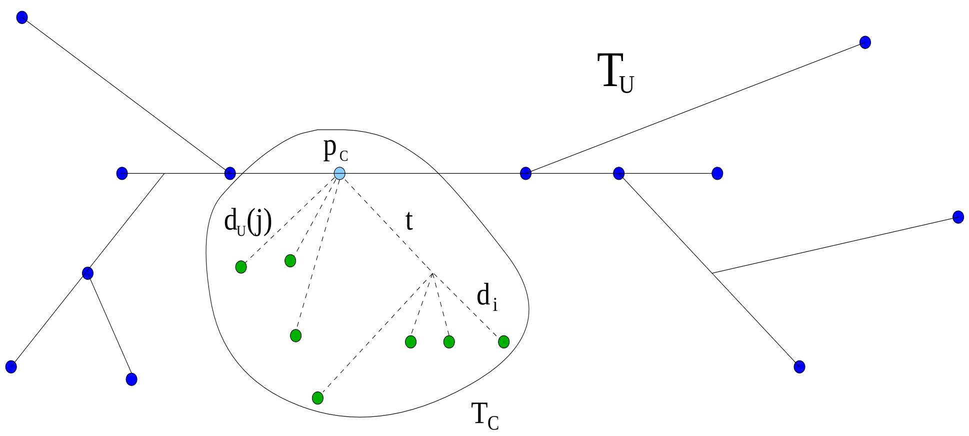

Figure 3: The tree T_C and its relation to the scaffold T_U.

3. Incompatible pairs i, j: We show that the number of incompatible pairs is at most $(\beta + \alpha) \cdot n \cdot |C|$. By definition of separators, for every $i, j, \ell \in C$, if either j or ℓ is incompatible with i, then i is a separator for j, ℓ. Hence, for each i that is not an α-effective separator, the number of points $j \in C$ that are not compatible with i is at most αn. To see why this is true observe first that if $|C| \leq \alpha n$ then the claim holds trivially. If $|C| > \alpha n$ and there are more than αn points $j \in C$ that are not compatible with i, then i would separate more than $\alpha n \cdot |C - 1| \geq (\alpha n)^2$ pairs of points (in contradiction to i not being an α-effective separator). Assuming that the number of α-effective separators is at most βn, we get that the total number of incompatible pairs is at most $\beta n \cdot |C| + \alpha n \cdot |C|$, as claimed.

The total number of pairs on which M and T_C differs is hence at most $(3\beta + 4\alpha) \cdot n \cdot |C|$. ■

Correctness of the Algorithm. The remainder of the proof of correctness of the algorithm proceeds essentially as the proof of Theorem 2 (where here we set $\alpha = \frac{\epsilon}{16}$ and $\beta = \frac{\epsilon}{12}$). Here too, with probability at least $5/6$ over the choice of U, either there is no tree T_U that is consistent with M on U, or such a tree exists but the number of α-effective separators with respect to U is at most βn. We can show that in the latter case, if M is ϵ-far from being a tree metric, then there are either more than $\frac{\epsilon}{4}n$ points that are inconsistent with U, or more than $\frac{\epsilon}{4}n^2$ violating pairs (thus causing the algorithm to reject with high probability). Assuming in contradiction that there are at most $\frac{\epsilon}{4}n$ points that are inconsistent with U, and at most $\frac{\epsilon}{4}n^2$ violating pairs, we can show that there exists a tree metric M' that disagrees with M on at most ϵn^2 entries. The matrix M' is defined the same as in the proof of Theorem 2 with the appropriate modified definition of D_U. Here though, for non-separated pairs of points i, j, we do not need to take the minimum between $D_U(i, j)$ and $T_C(i, j)$, since T_C was already defined so that $T_C(i, j) \leq D_U(i, j)$ for every $i, j \in C$.

Note that here too the "natural" algorithm that takes a sample of $\tilde{O}(1/\epsilon^3)$ points and checks whether it is possible to construct a tree that is consistent with these points, is a testing algorithm for tree metrics.

6 Testing Euclidean Metrics

For any two points $x, y \in \Re^d$, we denote by $\text{dist}(x,y)$ the Euclidean distance between x and y. That is, if $x = (x_1, \ldots, x_d)$ and $y = (y_1, \ldots, y_d)$, then $\text{dist}(x,y) \stackrel{\text{def}}{=} \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$.

An *embedding* of a set $U \subseteq [n]$ in d-dimensional Euclidean space is a mapping $\phi : U \to \Re^d$. The *dimension* of an embedding ϕ, denoted $\dim(\phi)$, is the dimension of the subspace in which the set of points $\{\phi(i)\}_{i \in U}$ resides.

Definition 6.1 (Euclidean Metrics) *Let M be an $n \times n$ matrix. If there is an embedding $\phi : [n] \to \Re^d$ such that $\text{dist}(\phi(i), \phi(j)) = M[i,j]$ for every $i, j \in [n]$, then we say that M is a d-*dimensional Euclidean metric.

In the above definition we allow ϕ to be many-to-one, so that M may actually be a pseudo-metric. However, with a slight abuse of terminology (and in particular so as not to confuse with pseudo-Euclidean metrics), we refer to M as being a Euclidean metric. In this section we describe an algorithm for testing whether a matrix M is a d-dimensional Euclidean metric as defined above, for any given integer d. In Section 7 we show that testing the stricter property, in which the embedding ϕ must be one-to-one, requires $\Omega(\sqrt{n})$ queries (for constant ϵ).

The basic underlying idea of the algorithm has appeared in various forms in our other algorithms as well. The idea is that a small sample from $[n]$ induces certain constraints that must be satisfied in case the tested matrix has the desired property.

Definition 6.2 (Consistent Embedding) *For a given matrix M and a subset $U \subseteq [n]$, we say that an embedding $\phi : U \to \Re^d$ is* consistent *with M on U, if $\text{dist}(\phi(i), \phi(j)) = M[i,j]$ for every $i, j \in U$. When $U = [n]$ we simply say that the embedding is consistent with M.*

If $U \subseteq [n]$ is such that there exists an embedding $\phi : U \to \Re^d$ that is consistent *with M on U, then we say that U is d-*embedable *with respect to M.*

Our testing algorithm is based on the following fact.

Fact 2 (Unique Embeddings) *Let M be an $n \times n$ matrix and let $U \subset [n]$ be a d-embedable subset with respect to M. For any set $S \subseteq [n]$, if there exists an embedding $\phi' : U \cup S \to \Re^d$ such that:*

1. *ϕ' is an extension of ϕ (that is, $\phi'(i) = \phi(i)$ for every $i \in U$);*
2. *ϕ' is consistent with M on $U \cup S$.*
3. *$\dim(\phi') = \dim(\phi)$;*

then the embedding ϕ' is unique*. Furthermore, for every $j \in S$, $\phi'(j)$ can be computed using only ϕ and the values $M[i,j]$ for every $i \in U$.*

The above fact implies that if M is a d-dimensional Euclidean metric, and $\phi : U \to \Re^d$ has dimension d (that is, the points $\{\phi(i)\}_{i \in U}$ are in general position), then there exists a unique embedding $\phi' : [n] \to \Re^d$ that is an extension of ϕ and is consistent with M.

Given a matrix M and a subset U that is d-embedable with respect to M, there is a straightforward iterative procedure for constructing an embedding $\phi : U \to \Re^d$ that is consistent with M on U.

The first point $i_1 \in U$ is mapped to $(0, \ldots, 0)$, the second point i_2 is mapped to $(M[i_1, i_2], 0, \ldots, 0)$, and in general, each new point is mapped to the lowest dimensional subspace possible. This procedure can be applied to any matrix M and subset U, but will of course fail if for the selected U is not d-embedable with respect to M.

In the above description we have ignored the issue of precision. As we shall see later (in Corollary 8), it will suffice to just solve the corresponding decision problem (i.e., does there exist such an embedding) which can be done in polynomial time. However, it will be instructive to think of the above (infinite-precision) procedure for sake of the presentation.

We next introduce two useful definitions. In both definitions, M is an $n \times n$ matrix.

Definition 6.3 (Consistent and Strongly Consistent Points) *Let $U \subset [n]$ and let $\phi : U \to \Re^d$ be an embedding of U that is consistent with M, and is derived by the iterative procedure mentioned above. We say that a point $j \notin U$ is* consistent *with U if there exists an extension $\phi' : U \cup \{j\} \to \Re^d$ of ϕ that is consistent with M. We say that j is* strongly consistent *with U if $\dim(\phi') = \dim(\phi)$.*

We denote the set of points in $[n] \setminus U$ that are consistent with U by Γ_U, and those that are strongly consistent by $\overline{\Gamma}_U$.

If M is a d-dimensional Euclidean metric, then all points are consistent with U, for every subset U. Thus, if the procedure for extending ϕ to some point j fails, we have evidence that M is not a d-dimensional Euclidean metric. Note that if j is strongly consistent with U then it is necessarily consistent with U. The implication in the other direction only holds when the dimension of ϕ is d, and in this case $\overline{\Gamma}_U = \Gamma_U$.

Definition 6.4 (Violating Pairs) *Let $U \subset [n]$ be d-embedable with respect to M, and let $\phi : U \to \Re^d$ be the embedding obtained by applying the iterative procedure mentioned earlier. For each point $j \in \overline{\Gamma}_U$, let $\phi'(j)$ be as determined by the unique extension of ϕ to $S = U \cup \{j\}$. We say that a pair of points $i, j \in \overline{\Gamma}_U$ are a* violating pair *with respect to U if $\mathrm{dist}(\phi'(i), \phi'(j)) \neq M[i, j]$.*

By Fact 2, if M is a d-dimensional Euclidean metric, then there are no violating pairs with respect to any subset U. Observe that the definition of violating pairs is applicable only to points that are strongly consistent with U. If a point j is consistent with U but not strongly consistent, then the extension ϕ' is not unique. Once again, if $\dim(\phi) = d$, then ϕ' is uniquely defined for all consistent points, and so in this case the above definition is applicable to all pairs of points in Γ_U.

Lemma 6.1 *Let $U \subset [n]$ be a subset for which there exist an embedding $\phi : U \to \Re^d$ that is consistent with M on U. If M is ϵ-far from being a d-dimensional Euclidean metric, then there are either more than $\frac{\epsilon}{2}n$ points that are not strongly consistent with respect to U or more than $\frac{\epsilon}{2}n^2$ violating pairs (of strongly consistent points).*

Recall that if $\dim(\phi) = d$ then we may exchange *not strongly consistent* in the above lemma, with *not consistent*.

Proof: Assume contrary to the claim that there are at most $\frac{\epsilon}{2}n$ points that are not strongly consistent with respect to U, and at most $\frac{\epsilon}{2}n^2$ violating pairs. We next show that there exists a d-dimensional Euclidean metric M' that differs from M on at most ϵn^2 entries. But this contradicts our assumption on M.

For each pair $i, j \in [n]$, we set $M'[i,j] = \mathrm{dist}(\phi'(i), \phi'(j))$ where $\phi' : [n] \to \Re^d$ is defined as follows. For each $i \in U$, let $\phi'(i) = \phi(i)$, and for each $i \in \overline{\Gamma}_U$, let $\phi'(i)$ be as determined by the unique extension of ϕ to $S = U \cup \{j\}$. For each point $i \in [n] \setminus (U \cup \overline{\Gamma}_U)$ we set $\phi'(i)$ arbitrary. Thus, M' and M differ on at most $\frac{\epsilon}{2}n^2$ violating pairs of points (both in $\overline{\Gamma}_U$), and on at most $\frac{\epsilon}{2}n^2$ pairs of points i, j such that either i or j are not strongly consistent with U. ■

Suppose that the algorithm was provided with a subset U for which $\phi : U \to \Re^d$ is consistent with M and has dimension d. By Lemma 6.1 (and the sentence just following the lemma), the algorithm could test whether M is a d-dimensional Euclidean metric (or ϵ-far from being such a metric) as follows: The algorithm would uniformly sample $4/\epsilon$ pairs of points and check that all points selected are consistent with U, and that all pairs of points are non-violating. Clearly, if M is a d-dimensional Euclidean metric, then the algorithm always accepts. On the other hand, by Lemma 6.1, if M is ϵ-far from being a d-dimensional Euclidean metric, then the probability that the sample contains no inconsistent point and no violating pair is at most $(1 - \frac{\epsilon}{2})^{4/\epsilon} < e^{-2} < 1/3$.

Since the algorithm is not provided with such a subset U, it tries to construct it in $d+1$ phases. The algorithm starts with $U = \{1\}$ and $\phi(1) = (0, \ldots, 0)$, and in each phase it selects a new sample of points. If the sample contains a point that is consistent with U but is not strongly consistent, it adds the point to U and extends ϕ to be defined on it (so that the dimension of ϕ increases). After d such phases, the algorithm has a subset U and an embedding $\phi : U \to \Re^d$ with dimension d as desired, and it can proceed as described above. (If at any phase an inconsistent point is selected then the algorithm can clearly reject). But what if at some phase all points selected are strongly consistent? Then the algorithm simply checks that all pairs are non-violating with respect to U.

Test Euclidean(d)

1. Let $U = \{1\}$, and $\phi(1) = (0, \ldots, 0) \in \Re^d$. (Thus, ϕ initially has dimension 0).
2. While $\dim(\phi) \leq d$ (and the algorithm has not yet rejected or accepted), do:
   (a) Uniformly and independently select $s = O(\log(d)/\epsilon)$ pairs of points.
   (b) If any of the points selected is not consistent with U, then reject.
   (c) Otherwise (all points are consistent), if there exists a point j in the sample that is not strongly consistent with U: then add j to U and extend ϕ to be defined on $U \cup \{j\}$. (This step is applicable only as long as $\dim(\phi) < d$).
   (d) Otherwise (all points are strongly consistent): if any of the pairs of points is violating with respect to U, then reject. Otherwise, accept.

Theorem 7 *The above algorithm is a testing algorithm for Euclidean metrics.*

Note that when the algorithm rejects it provides *evidence* that M is not a d-dimensional Euclidean metric (in the form of a subset of points for which there is no d-dimensional embedding that is consistent with M).

Proof: If M is a d-dimensional Euclidean metric then it is clearly accepted by the algorithm. Thus, assume M is ϵ-far from being a d-dimensional Euclidean metric. Consider any *fixed* phase in the algorithm. By Lemma 6.1, there must either be more than $\frac{\epsilon}{2}n$ points in $[n]$ that are not strongly consistent with U, or there must be more than $\frac{\epsilon}{2}n^2$ violating pairs of strongly consistent points. In the first case, for $s \geq \frac{6\ln(d+1)}{\epsilon}$, the probability that the sample of $2s$ points does not contain any point that is not strongly consistent with U is at most $(1-\frac{\epsilon}{2})^{2s} < e^{-s\epsilon} = \frac{1}{6(d+1)}$. Since there are at most $d+1$ such phases, the probability that there exists a phase in which there are more than $\frac{\epsilon}{2}n$ points in $[n]$ that are not strongly consistent with U, but the algorithm does not select such a point, is at most $\frac{1}{6}$. (Note that if we reach the last phase, then a point that is not strongly consistent with U is not consistent with U, and so when selected will cause the algorithm to reject). In the latter case (there are more than $\frac{\epsilon}{2}n^2$ violating pairs), for $s \geq \frac{12}{\epsilon}$, the probability that no violating pair is selected is at most $(1-\frac{\epsilon}{2})^s < e^{-s\epsilon/2} = \frac{1}{6}$. Thus, the probability that the algorithm rejects is at least $1-(1/6+1/6) = 2/3$. ■

As a direct corollary to Theorem 7 we get.

Corollary 8 *Let the "natural" testing algorithm be the algorithm that simply selects a uniform sample of $O(d \log d/\epsilon)$ points from $[n]$ and accepts if and only if the sample selected is d-embedable with respect to M. Then the natural algorithm is a testing algorithm for d-dimensional Euclidean metrics.*

Deciding whether the sample S is d-embedable with respect to M can be done in polynomial time as follows. For our convenience, we renumber the points in S so that $S = \{0,\ldots,m-1\}$. We are thus asking whether there exist d-dimensional vectors $v^0,\ldots,v^{m-1}$ such that $\mathrm{dist}(v^i,v^j) = M[i,j]$ for every $i,j \in S$. Since we may assume without loss of generality, that v^0 is the all-0 vector, the problem can be rephrased as deciding whether there exist $m-1$ vectors, such that the inner product between v^i and v^j equals $Q[i,j]$, where $Q[i,j] \stackrel{\mathrm{def}}{=} \frac{1}{2}(M^2[i,0]+M^2[j,0]-M^2[i,j])$. Thus our problem reduces to deciding whether the matrix Q is positive semi-definite and has rank at most d. The first task can be performed by computing the characteristic polynomial of the matrix, and approximating its roots to check whether they are all positive. The second task is done by Gaussian elimination.

Proof: If M is d-dimensional Euclidean metric then clearly the natural algorithm always accepts. We next show that if M is ϵ-far from being a d-dimensional Euclidean metric then the natural algorithm rejects with probability at least $2/3$. To see why this is true, observe that every sample that causes the algorithm Test Euclidean to reject necessarily causes the natural algorithm to reject (since if in any phase there is either an inconsistent point or a violating pair, then the complete sample is not d-embedable with respect to M). ■

7 Lower Bounds

We say that an $n \times n$ matrix M is a *proper* d-dimensional Euclidean metric, if there exists an embedding $\phi : [n] \rightarrow \Re^d$ that is consistent with M *and is one-to-one*. We define *proper* tree metrics in an analogous manner. In this section we show the following lower bound.

Theorem 9 *Any algorithm for testing proper d-dimensional Euclidean metrics requires $\Omega(\sqrt{n})$ queries. Similarly, any algorithm for testing proper tree metrics requires $\Omega(\sqrt{n})$ queries. These bounds hold for testing algorithms that are allowed two-sided error probability.*

The Lower Bound Idea. Before we give the formal argument for our lower bounds, we describe the basic idea which is common to both bounds. Consider a matrix M that is defined as follows.

For $i = 1, \ldots, \frac{n}{2}$ and $j = 1, \ldots, \frac{n}{2}$, let: $M[2i-1, 2j-1] = M[2i-1, 2j] = M[2i, 2j-1] = M[2i, 2j] = |j - i|$. Thus, in the Euclidean case, the embedding $\phi : [n] \to \Re$ that maps each pair of points $\{2i-1, 2i\}$ to the integer $i \in \Re$ is consistent with M. For an illustration, see Figure 4. Similarly, in the tree metric case, the set $[n]$ can be mapped consistently with M to the tree T which is a path of $\frac{n}{2}$ nodes $\{1, \ldots, \frac{n}{2}\}$, where node i is connected to node $i+1$ by an edge of weight 1 (and $2i-1, 2i \in [n]$ are both mapped to node i). In other words, M is a 1-dimensional Euclidean metric and also a tree metric that corresponds to a path.

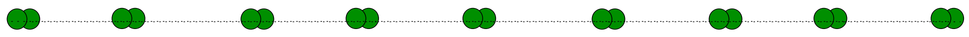

Figure 4: An embedding consistent with the matrix M. A pair of points is mapped to each integer on the line in the range $\{1, \ldots, \frac{n}{2}\}$.

Clearly, M is not a proper Euclidean metric. We next show that M is actually $\Omega(1)$-far from being a proper d-dimensional Euclidean metric, for any d. It can similarly be shown that M is $\Omega(1)$-far from being a proper tree metric. Consider any 3 disjoint pairs of points, $\{2i-1, 2i\}$, $\{2j-1, 2j\}$, and $\{2k-1, 2k\}$. Assume without loss of generality that $i < j < k$. Then for any $x \in \{2i-1, 2i\}$, $y \in \{2j-1, 2j\}$ and $z \in \{2k-1, 2k\}$,

$$M[x, z] = M[x, y] + M[y, z].$$

Consider any one-to-one embedding ϕ that maps the 6 points $\{2i-1, 2i, 2j-1, 2j, 2k-1, 2k\}$ to $\Re^d$ (for any d). Then it is easy to verify, that necessarily for some $a, b \in \{i, j, k\}$, $a \neq b$, and for some $x \in \{2a-1, 2a\}$ and $y \in \{2b-1, 2b\}$,

$$\mathrm{dist}(\phi(x), \phi(y)) \neq M[x, y]. \tag{6}$$

Now consider an auxiliary undirected graph G_ϕ over the vertex set $\{1, \ldots, \frac{n}{2}\}$, such that there is an edge between vertices a and b if an only if Inequality 6 holds for some $x \in \{2a-1, 2a\}$ and $y \in \{2b-1, 2b\}$. Then we know that in G_ϕ, for every three vertices, at least two are connected by an edge. That is, there is no independent set of size 3. Thus, by Turán's Theorem [21], the number of edges in G_ϕ is $\Omega(n^2)$. By definition of G_ϕ this means that the distance that is induced by ϕ between pairs of points in $[n]$ disagrees with M on $\Omega(n^2)$ entries. Since this holds for any one-to-one embedding ϕ, we get that M is $\Omega(1)$-far from being a proper Euclidean metric.

Finally we give a lower bound on the number of queries required by the "natural" testing algorithm. While this does not imply a lower bound for every testing algorithm, it provides intuition to the difficulty of the problem. The natural algorithm takes a uniform sample of points from $[n]$ and tries to construct a one-to-one embedding of the points in $\Re^d$. If it succeeds, then it accepts, and otherwise it rejects. Note that as long as the algorithm does not select both $2i-1$ and $2i$ for some $1 \leq i \leq \frac{n}{2}$, then it is possible to embed the sample in $\Re$. By the well-known *Birthday Paradox*, if the number of points selected is sufficiently smaller than $\sqrt{n}$, then with high probability no such pair $2i-1$ and $2i$ is selected. A similar argument holds for the natural testing algorithm for proper tree metrics.

Generalizing the Lower Bounds. In order to generalize the lower bounds to any testing algorithm, we do the following. Consider first the problem of testing proper Euclidean metrics. We describe two families of matrices, such that in one family all matrices are proper Euclidean metrics, while in the other family all matrices are $\Omega(1)$-far from being proper Euclidean matrices. However, it is not possible to distinguish with sufficient success probability between a matrix selected uniformly in the first family, and a matrix selected in the second family, using less than $c\sqrt{n}$ queries, for some constant $c < 1$. Since our lower bound argument is very similar to other known lower bound proofs (*cf.* [15, 2]), we only provide a sketch.

The two families of matrices are determined by actual embeddings of $[n]$ into $\Re^d$. The first family consists of all one-to-one mappings from $[n]$ to two parallel lines, each containing $n/2$ equally spaced positions. We may also think of the second family as a mapping to two parallel lines with equally spaced positions. Here though the range of each mapping in the family consists of only half the positions: For each position, either two points are mapped to this position, or two points are mapped to the "parallel position". For an illustration, see Figure 5.

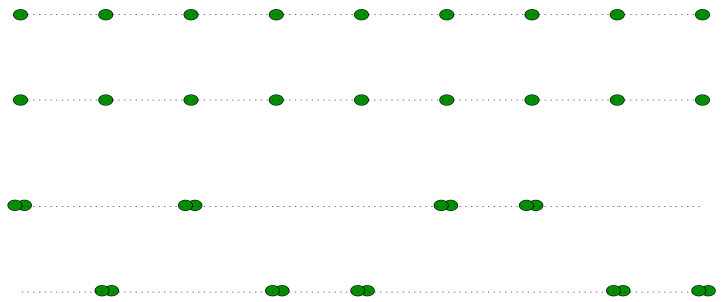

Figure 5: An illustration of the lower bound construction for Euclidean metrics.

By definition, all matrices in the first family are proper 2-dimensional Euclidean metrics. Note that in the second family there are at least $n/4$ pairs of points mapped to one of the lines. Thus, we can prove, as we did previously, that every matrix in the second family is $\Omega(1)$-far from being a Euclidean metric. (The fact that the positions to which the pairs are mapped to, are not equally spaced, is immaterial to the proof).

Now consider two "query answering" processes that can interact with a testing algorithm while constructing a random matrix M. The first process answers the algorithm's queries while constructing a uniformly selected matrix in the first family, and the second process does so by constructing a uniformly matrix in the second family. This is done in the following manner. In either case the process maintains a partial mapping of those points i which appeared in queries performed by the algorithm (that is, queries concerning entries $M[i, j]$). Given a new query $M[i, j]$, if i is not yet positioned (mapped), then both processes select a position, and map i to this position (a similar selection is done for j if it is not yet positioned). The processes then answer the query consistently with the mapping they have. The two processes thus differ only in the way they select a position for a new point i.

1. The first process uniformly selects a vacant position on one of the two parallel lines, and places the point in that position. It is easy to verify that this is equivalent to selecting a position in the following manner: The process first selects a pair of parallel positions according to the distribution induced by selecting a uniform vacant position. Namely, suppose that there are n_1 parallel pairs of positions that are both vacant, and n_2 in which one position is vacant and one is occupied. Then with probability $\frac{2n_1}{2n_1+n_2}$ a pair of the first type is selected, and with probability $\frac{n_2}{2n_1+n_2}$ a pair of the second type is selected. (Among each type the selection is uniform). If both parallel positions selected are vacant, then the process selects one of the two with equal probability. If only one is vacant, then it places the point in that position.

2. The second process selects a pair of parallel positions according to the same distribution. If both positions are vacant, it too selects one of the two with equal probability. However, if one is occupied, then it positions the new point in the same position.

Hence, as long as no pair of parallel positions is selected twice, the distribution on the position of the new point (or points) is the same for both processes (and hence the distribution on the answer to the query $M[i,j]$ is the same). It is easy to verify that for a sufficiently small constant $c < 1$, if less than $c\sqrt{n}$ queries are performed, then the probability that a parallel pair of positions is selected twice is very small. The lower bound follows (where the details are similar to those in [15]).

The lower bound for proper tree metrics follows the same lines, and uses a similar choice of families of matrices. In the first family all matrices are determined by a "comb-tree" (see Figure 6), where the points in $[n]$ are mapped both to the "root" and to the "tip" of each "comb tooth". In the second family, the trees have "missing teeth" and pairs of points are mapped either to the root of a missing tooth or to the tip of an existing tooth. The only slight technicality that did not arise in the the Euclidean case, is that here some matrices from the second family are not very far from being proper tree metrics (since relatively few pairs of points are mapped to the "base of the comb"). However, the probability that a uniformly selected matrix in the second family will correspond to such a tree is negligible.

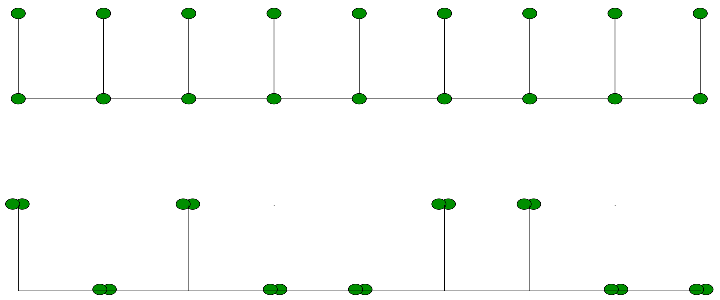

Figure 6: An illustration of the lower bound construction for general tree metrics.

Further Research.

First, we suspect that our algorithms are not optimal in terms of their dependence on ϵ (and d in the case of Euclidean metrics), and a more thorough study of this issue is in place.

Second, we conjecture that our algorithms for testing general tree metrics can be extended to deal with the approximate versions of this property in a similar way to what we show for ultrametrics. This problem though may be more difficult, as exhibited by the hardness results described above.

Acknowledgments

We are greatly in debt to Martin Farach-Colton and Michael Bender for their participation at the initial stages of this work. We would like to thank Madhu Sudan, Bernard Chazelle, and Michelle Goemans for their help.

References

[1] R. Agarwala, V. Bafna, M. Farach, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy (fitting distances by tree metrics). *SIAM Journal on Computing*, 28(3):1073–1085, 1999.

[2] N. Alon, S. Dar, M. Parnas, and D. Ron. Testing of clustering. In *Proceedings of the Forty-First Annual Symposium on Foundations of Computer Science*, pages 240–250, 2000.

[3] N. Alon, E. Fischer, M. Krivelevich, and M Szegedy. Efficient testing of large graphs. In *Proceedings of the Fortieth Annual Symposium on Foundations of Computer Science*, pages 645–655, 1999.

[4] N. Alon, M. Krivelevich, I. Newman, and M Szegedy. Regular languages are testable with a constant number of queries. In *Proceedings of the Fortieth Annual Symposium on Foundations of Computer Science*, pages 656–666, 1999.

[5] J-P Barthélemy and A. Guénoche. *Trees and Proximity Representations.* Wiley, New York, 1991.

[6] L. Cavalli-Sforza and A. Edwards. Phylogenetic analysis models and estimation procedures. *American Journal of Human Genetics*, 19:233–257, 1967.

[7] J. Culberson and P. Rudnicki. A fast algorithm for constructing trees from distance matrices. *Information Processing Letters*, pages 215–220, 1989.

[8] A. Czumaj, C. Sohler, and M. Ziegler. Property testing in computational geometry. In *Proceedings of the 8th Annual European Symposium on Algorithms (ESA)*, pages 155–166, 2000. Lecture Notes in Computer Science edited by M. Paterson, Springer-Verlag, Berlin.

[9] W. H. E. Day. Computational complexity of inferring phylogenies from dissimilarity matrices. *Bulletin of Mathematical Biology*, 49(4):461–467, 1987.

[10] A. Dress and V. von Haessler (Eds.). *Trees and Hierarchical Structures.* Springer Verlag, 1987. Lecture Notes in Bio-Mathematics.

[11] F. Ergun, S. Kannan, S. R. Kumar, R. Rubinfeld, and M. Viswanathan. Spot-checkers. In *Proceedings of the Thirty-Second Annual ACM Symposium on the Theory of Computing*, pages 259–268, 1998.

[12] M. Farach, S. Kannan, and T. Warnow. A robust model for finding optimal evolutionary trees. *Algorithmica*, 13(1/2):155–179, 1995.

[13] J. Felsenstein. Numerical methods for inferring evolutionary trees. *Quarterly Review on Biology*, 57(4):379–404, 1982.

[14] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998.

[15] O. Goldreich and D. Ron. Property testing in bounded degree graphs. In *Proceedings of the Thirty-First Annual ACM Symposium on the Theory of Computing*, pages 406–415, 1997. To appear in *Algorithmica*.

[16] S. Kannan, E. Lawler, and T. Warnow. Determining the evolutionary tree. *Journal of Algorithms*, 21(1):26–50, 1996.

[17] M. Křivánek. The complexity of ultrametric partitions on graphs. *Information Processing Letters*, 27:265–270, 1988.

[18] Dana Ron. Property testing. To appear in the Handbook on Randomization. Currently available from: `http://www.eng.tau.ac.il/~danar`, 2000.

[19] R. Rubinfeld and M. Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.

[20] P. H. A. Sneath and R. R. Sokal. *Numerical Taxonomy.* W. H. Freeman, 1973.

[21] P. Turán. On an extremal problem in graph theory (in Hungarian). *Mat. Fiz. Lapok*, 48:436–452, 1941.

[22] M. S. Waterman, T. F. Smith, M Singh, and W. A. Beyer. Additive evolutionary trees. *Journal of Theoretical Biology*, 64:199–213, 1977.