# Testing Properties of Sets of Points in Metric Spaces

Krzysztof Onak*

Massachusetts Institute of Technology, Cambridge MA 02139, USA

**Abstract.** Given query access to a set of points in a metric space, we wish to quickly check if it has a specific property. More precisely, we wish to distinguish sets of points that have the property from those that need to have at least an $\varepsilon$ fraction of points modified to achieve it.
We show one-sided error testers that immediately follow from known characterizations of metric spaces. Among other things, we give testers for tree metrics and ultrametrics which are optimal among one-sided error testers. Our tester for embeddability into the line is optimal even among two-sided error testers, and runs in sublinear time. We complement our algorithms with several lower bounds. For instance, we present lower bounds for testing dimensionality reduction in the $\ell_1$ and $\ell_\infty$ metrics, which improve upon lower bounds given by Krauthgamer and Sasson (SODA 2003). All our lower bounds are constructed by using a generic approach.
We also look at the problem from a streaming perspective, and give a method for converting each of our property testers into a streaming tester.

## 1 Introduction

Many real-world data sets are sets of points in a metric space. If the metric space is complicated or, like high-dimensional spaces in many applications, expensive to deal with, then a natural question is that of finding a simplified representation of the input set of points. In many cases, we are not as much interested in the actual points as in the distances between them. We may then consider mapping the data set to a simpler space so that the distances between the points are either exactly or approximately preserved.

The best example of a tool that allows for such transformation is the Johnson-Lindenstrauss lemma (see [1] and [2]). It states that for any $\varepsilon > 0$, there exists a mapping of a set of $n$ points in $\ell_2$ into $\ell_2^{O(\log(n)/\varepsilon^2)}$ with multiplicative distortion $1+\varepsilon$. For instance, if small distortion is acceptable, and we have an algorithm that runs in time exponential in the dimension, then by using the Johnson-Lindenstrauss lemma, we may get a polynomial-time approximation algorithm.

Another possible approach is to take advantage of profound properties of our data sets in constructing an embedding into a simpler space. But how can one

* Supported by an Akamai Presidential Fellowship and NSF grant 0514771.

efficiently discover such properties? This problem was addressed in a variety of settings by Parnas and Ron [3] and Krauthgamer and Sasson [4], who focused on constructing testers for multiple metric properties. By using those testers, one may check if a data set or a metric is close to a specific property, and if that turns out to be the case, try to use the property to construct a nice embedding into a simpler space. We continue this line of research. In particular, we follow and generalize the model of Krauthgamer and Sasson. We describe existing models and previous results on them in more detail later in this section.

### 1.1 Property Testing

In *property testing* (see [5,6]), one is interested in checking if the input (for instance, a set of points) has a specific property. We, however, do not attempt to answer this question exactly. Instead, we try to quickly distinguish, by reading a small part of the input, between sets of points that have the property, and sets of points that are significantly different from any set that has the property. We assume some notion of the distance from the property. Usually, the distance is defined as the minimum fraction of the input that needs to be modified to achieve the property. If the distance of an input $x$ from the property is at least $\varepsilon$, then we say that $x$ is *$\varepsilon$-far* from the property. We now define what a tester is.

**Definition 1.** *A* (two-sided error) tester *for a property $\mathcal{P}$ is an algorithm that accepts an input that has property $\mathcal{P}$ with probability at least $2/3$, and rejects with probability at least $2/3$ every input that is $\varepsilon$-far from $\mathcal{P}$. Moreover, if the tester never rejects any input that has property $\mathcal{P}$, we say that such a tester has* one-sided error*.*

Note that one-sided error testers only reject an input if they find evidence that it does not have a given property. Traditionally, the main quantities minimized in property testing are the query complexity and the running time of a tester.

### 1.2 Considered Models and Previous Results

*The Model of Parnas and Ron.* Parnas and Ron [3] assumed that the input metric on $n$ points was given as an $n \times n$ matrix of distances between each pair of points. The distance to a property was in their setting defined as the minimum number of matrix entries that must be modified to achieve the property. They showed one-sided error testers for verifying if the input metric embeds into $\ell_2^d$, if it is a tree metric, an ultrametric or an approximate ultrametric. Their testers choose a random subset of points of size independent of the size of the metric, and check if the metric restricted to them has the property. We consider almost the same set of properties in a different setting. Unfortunately, in our setting the numbers of queries must depend on the size of the metric.

It is also worth mentioning that Abraham et al. [7] considered a related notion of embeddings that preserve all but a small fraction of distances.

*The Model of Krauthgamer and Sasson.* The problem of testing a dimension of a set of points was stated by Krauthgamer and Sasson [4]. They assumed that their input is a set of points in a given metric space. What differentiates their model from the model of Parnas and Ron is that the distance from a property equals the minimum fraction of points, rather than the minimum fraction of distances, that must be modified. Krauthgamer and Sasson asked if given a set of points in $\ell_p^k$, we can efficiently determine if it isometrically embeds into $\ell_p^d$, for some fixed $d$. They showed that for $p = 2$, it suffices to read a random subset of points of size $O(d/\varepsilon)$ to find with constant probability a certificate that the entire set does not embed into $\ell_p^d$, provided it is $\varepsilon$-far from embeddability into $\ell_p^d$. Furthermore, Krauthgamer and Sasson showed that for $p = 1$, any tester for embeddability of a set of points in the $\ell_1$ metric into $\ell_1^d$ must query $\Omega(\sqrt[4]{n})$ points. They also gave a lower bound of $\Omega(\sqrt{n/\Delta})$ for testing embeddability of a set of points in $\ell_2^m$ with distortion $\Delta$ into $\ell_2^d$, and a lower bound of $\Omega(\min\{\sqrt{n}, \sqrt{m/\log m}\})$ for testing if a set of points in $\ell_2^m$ can be perturbed by $\delta > 0$ so that it isometrically embeds into $\ell_2^d$.

*Our Generalized Model.* We also assume that the input is a set of points. We, however, take a more general look at the model proposed by Krauthgamer and Sasson. As opposed to them, we do not assume anything about the metric space the input set of points lies in. Instead, our testers have query access to a distance oracle for the underlying metric space. The distance to a property is defined as follows throughout the whole paper.

**Definition 2.** *We say that a set $S$ of points is* $\varepsilon$-far *from a property if any subset of $S$ of more than $(1-\varepsilon)|S|$ points does not have the property.*

### 1.3 Considered Properties

For completeness, we first recall the notion of a metric space.

**Definition 3.** *Let $\mathcal{M}$ be a pair $\langle S, \delta\rangle$, where $S$ is a set of points and $\delta$ is a function from a pair of points in $S$ to $\mathbb{R}_{\geq 0}$, the set of non-negative reals. We say that $\mathcal{M}$ is a* metric space *if for any $x, y, z \in S$, $\delta(x, y) = \delta(y, x)$, $\delta(x, y) = 0$ iff $x = y$, and $\delta(x, z) \leq \delta(x, y) + \delta(y, z)$.*

Properties that we test are easy to express via embeddings.

**Definition 4.** *A metric space $\mathcal{M}_1 = \langle S_1, \delta_1\rangle$ is* embeddable *(or* embeds*) into a metric space $\mathcal{M}_2 = \langle S_2, \delta_2\rangle$ if there exists a mapping $f$ from $S_1$ to $S_2$ that preserves distances, i.e. for any pair $x$ and $y$ of points in $S$, it holds that*

$$\delta_1(x, y) = \delta_2(f(x), f(y)).$$

We now define what a tree metric and an ultrametric are. Note that every ultrametric is a tree metric.

**Definition 5.** *A metric $\mathcal{M}$ is a* tree metric *if it can be embedded into the shortest-path metric of some weighted tree.*

*Alternatively, $\mathcal{M} = \langle S, \delta \rangle$ is a tree metric if it meets the following 4-point condition:*

$$\forall x, y, z, w \in S, \quad \delta(x, y) + \delta(z, w) \leq \max\{\delta(x, z) + \delta(y, w), \delta(x, w) + \delta(z, y)\}.$$

**Definition 6.** *A metric $\mathcal{M}$ is an* ultrametric *if there exists a weighted rooted tree $T$ of all leaves at the same distance from the root, and $M$ embeds into the shortest-path metric of $T$ with all points in $\mathcal{M}$ mapped to points corresponding to the leaves of $T$.*

*Alternatively, $\mathcal{M} = \langle S, \delta \rangle$ is an ultrametric if it meets the following 3-point condition:*

$$\forall x, y, z \in S, \quad \delta(x, y) \leq \max\{\delta(x, z), \delta(y, z)\}.$$

### 1.4 Our Results

**Embeddability into the Line:** We show an optimal $O(\sqrt{n/\varepsilon})$-query one-sided tester. We prove that any tester, including two-sided testers, must query $\Omega(\sqrt{n/\varepsilon})$ points even if the set of points is a subset of $\ell_1^2$. This improves upon the $\Omega(\sqrt[4]{n})$ lower bound of Krauthgamer and Sasson [4].

**Tree Metrics and Ultrametrics:** We exhibit one-sided error testers of query complexity $O(n^{2/3}\varepsilon^{-1/3})$. The testers are optimal among one-sided tester.

**Embeddability into $\ell_1^2$ and $\ell_2^d$:** We exhibit one-sided testers of query complexity $O(n^{5/6}\varepsilon^{-1/6})$ and $O(n^{(d+2)/(d+3)}\varepsilon^{-1/(d+3)})$, respectively. Note that the spaces $\ell_1^2$ and $\ell_\infty^2$ are isometric.

**Dimension Reduction Testing in $\ell_1$ and $\ell_\infty$:** We strengthen the results of Krauthgamer and Sasson [4] on the dimension reduction. We show that any one-sided tester for testing if a set of points in $\ell_1^{d+1}$ embeds into $\ell_1^d$ must query $\Omega(n^{d/(d+1)}\varepsilon^{-1/(d+1)})$ points for sufficiently small $\varepsilon$. For the analogous setting in $\ell_\infty$, we prove a lower bound of $\Omega(n^{2^{d-1}/(2^{d-1}+1)}\varepsilon^{-1/(2^{d-1}+1)})$.

**Embeddability into the Line with Distortion:** In any metric $\ell_p$, for every $\delta > 0$, there exists a dimension $d$, and a constant $C > 1$ such that a one-sided error tester for embeddability of points in $\ell_p^d$ into the line with distortion at most $C$ must query $\Omega(n^{1-\delta})$ points.

Most of our lower bounds only apply to one-sided testers, but all known testers for these and related problems have one-sided error. Due to the space limitation, many of our results are not included in this version of the paper.

### 1.5 A Streaming Perspective

*The Model.* We also take a look at the testing problem from a streaming perspective (see [8] for a survey on streaming algorithms). For our purposes, a *streaming algorithm* is an algorithm that takes an input stream, and computes a result in one pass over the input. A streaming algorithm can read the *entire* input, but only once. The main quantity that is minimized in streaming is the space complexity.

Feigenbaum et al. [9] considered a model that combines streaming and property testing. A *streaming tester* takes an input stream, and accepts with probability at least $2/3$, if the input has a given property, and rejects with probability $2/3$, if the input does not have the property.

*Our Results.* We first show that the exact verification of properties considered by us requires at least $\Omega(n)$ bits of space. This lower bound can easily be overcome for most of properties that we consider by using stream testers. In particular, we show that for each property that we had an algorithm that used $O(n^{(d-1)/d}\varepsilon^{-1/d})$ samples in the property testing approach, there is a streaming tester that needs space to keeps only $O(n^{(d-2)/(d-1)}\varepsilon^{-1/(d-1)})$ points. For instance, for embeddability into the line, this gives a streaming tester that keeps only $O(1/\varepsilon)$ points.

### 1.6 Our Techniques

*Testers via Small Subspace Characterizations.* There are several properties (see for instance [10, 11]) that can be characterized by a property that holds for any subset of points of size of at most $c$, for some constant $c$. In this case, we can create a one-sided tester that looks for a small subset of at most $c$ points that do not have the property. Using this approach we get a tester for ultrametrics which is optimal among one-sided testers.

Moreover, to build an efficient tester for embeddability into $\ell_1^2$, we use the algorithm of Edmonds [12] to check if a collected sample embeds into $\ell_1^2$. In the case of testing for embeddability into the line and for being a tree metric, this general approach does not yield an optimal tester, but we prove that with respect to some fixed number of points, it suffices to find a smaller group of points, and therefore, we can improve the query complexity of the testers. For instance, in testing for tree metrics, it essentially suffices to find a triple, not a quadruple of points of a specific property, and therefore, the query complexity improves.

*Lower Bounds for Property Testing.* All our lower bounds follow from the same approach. We construct a gadget, and make several copies of it. Any subset of points that does not contain an entire copy of the gadget has the property, but the whole set of points is far from the property. This implies that a one-sided tester must read an entire copy of the gadget to reject the input. To construct such gadgets in $\ell_1$, we use a theorem of Hadlock and Hoffman [13]. We show and use an analogue of this result in $\ell_\infty$.

*Streaming Testers.* All our lower bounds follow from a simple application of the set disjointness lower bound [14, 15]. As for algorithms, we notice that whenever a standard property tester looks for a $k$-tuple of points to find evidence that the input does not have a property, a streaming tester may draw only the first $k-1$ points of the tuple in the stream, and it will notice the $k$-th complementing point, when it reads it. An improvement follows from the fact that finding $(k-1)$-tuples is easier than finding $k$-tuples.

## 2 Two Simple Probability Facts

We use two probability facts throughout the paper. Suppose that a set contains many disjoint groups of elements, and by selecting elements of the set at random, we wish to draw at least one of the groups entirely. The facts below specify what number of samples is sufficient and what number of samples is necessary. We omit their proof in this version of the paper.

**Fact 7 (Upper bound).** *Let $S$ be a set of $n$ items where some of them constitute $g$ disjoint groups of size $k$ each. It suffices to select* $\min\left\{\frac{2n}{g^{1/k}}, n\right\} = O\left(\frac{n}{g^{1/k}}\right)$ *items at random to draw at least one of the groups entirely with constant probability.*

**Fact 8 (Lower bound).** *Let $S$ be a set of $n$ items where some of them constitute $g$ disjoint groups of size $k$ each. The probability that by we draw at least one group entirely by choosing at random $q$ items from $S$ is not greater than $g \cdot (q/n)^k$.*

## 3 Testing via a Small Subset Characterization

Some properties $P$ can be expressed as a condition that says that there exists a constant $c$ such that a metric space $\mathcal{M}$ has property $P$ if and only if every subspace of $\mathcal{M}$ of at most $c$ points has a computable property $P'$. Apart from the alternative definitions of a tree metric and an ultrametric, we list here the following two examples:

- A metric spaces $M$ embeds into $\ell_1^2$ (or equivalently into $\ell_\infty^2$) if and only if each subset of $M$ of at most 6 points embeds into $\ell_1^2$ (Bandelt and Chepoi [10]).
- A metric spaces $M$ embeds into $\ell_2^d$ if and only if each subset of $M$ of at most $d+3$ points embeds into $\ell_2^d$ (Menger [11]).

All properties of this form yield testers of sublinear query complexity.

**Theorem 9.** *Let $c$ be a constant such that a set $S$ of points has a property $P$ if and only if every subset of $S$ at most $c$ points has a computable property $P'$. There exists a one-sided error tester for $P$ that queries $O\left(n^{1-1/c}\varepsilon^{-1/c}\right)$ points. The tester finds with constant probability evidence that $S$ does not have $P$, provided $S$ is $\varepsilon$-far from having it.*

*Proof.* Let $S$ be $\varepsilon$-far from $P$. This implies that any subset of $S$ of at least $(1-\varepsilon)n$ does not have the property $P$.

Let $S_0$ be equal to $S$. As long as $|S_i| > (1-\varepsilon)n$, we inductively define $S_{i+1}$ and $T_{i+1}$ as follows. Since $S_i$ does not have the property $P$, there exists a subset of $S_i$ of at most $c$ points that does not have the property $P'$. Let $T_{i+1}$ be any such subset, and let $S_{i+1} = S_i \backslash T_{i+1}$. Eventually, we have at least $\varepsilon n/c$ disjoint groups, each of size at most $c$ such that any of them proves that the set does not have $P$.

By Fact 7 it suffices to draw $O\left(n^{1-1/c}\left(\frac{c}{\varepsilon}\right)^{1/c}\right) = O\left(n^{1-1/c}\varepsilon^{-1/c}\right)$ random elements to entirely draw with constant probability at least one of these groups, and hence to discover that $S$ does not have $P$. Then, because $P'$ is computable, it suffices to verify that $P'$ holds for every subset of at most $c$ points. $\square$

Theorem 9 and the aforementioned characterizations yield sublinear-query testers. Their running time can be improved, by checking if the whole sample subset has the given property. One can check if a metric on $n$ points is a tree metric or an ultrametric in $O(n^2)$ time [16], and check if it embeds into $\ell_1^2$ in $O(n^2 \log^3 n)$ time [12]. We summarize all the results in the corollary below.

**Corollary 10.** *There are sublinear-query one-sided error testers if the input set of points*

- *spans a tree metric (query complexity: $O(n^{3/4}\varepsilon^{-1/4})$, time $O(n^{3/2}\varepsilon^{-1/2})$),*
- *spans an ultrametric (query complexity: $O(n^{2/3}\varepsilon^{-1/3})$, time $O(n^{4/3}\varepsilon^{-2/3})$),*
- *embeds into $\ell_1^2$ (query complexity: $O(n^{5/6}\varepsilon^{-1/6})$, time $O(n^{5/3}\varepsilon^{-1/3}\log^3 n)$),*
- *embeds into $\ell_2^d$ (query complexity: $O(n^{(d+2)/(d+3)}\varepsilon^{-1/(d+3)})$).*

# 4 Improved Testers

## 4.1 Testing Tree Metrics

We now show a slightly more efficient algorithm for testing if a metric spanned by a set of points is a tree metric. Recall that Corollary 10 gave us a tester of query complexity $O(n^{3/4}\varepsilon^{-1/4})$. The reason behind the complexity is that the tester looks for quadruples of points. The lemma below implies that it really suffices to look for triples of points, and we can therefore improve the query complexity to $O(n^{2/3}\varepsilon^{-1/3})$. We omit the proof in this version of the paper.

**Lemma 11.** *Let $S = \{x, y, s, t\}$ be a subset of four points in a metric space that spans a non-tree submetric. Let $p$ be an arbitrary point in the same metric space. There exists a subset $S'$ of $S$ of size 3 such that $S' \cup \{p\}$ spans a non-tree submetric as well.*

**Corollary 12.** *There is a one-sided error tester for being a tree metric that queries only $O(n^{2/3}\varepsilon^{-1/3})$ points and runs in $O(n^{4/3}\varepsilon^{-2/3})$ time.*

### 4.2 Testing Embeddability into the Line

We now show an optimal tester for embeddability into the line. The query complexity of the tester is $O(\sqrt{n/\varepsilon})$. Note that this significantly improves on $O(n^{3/4}\varepsilon^{-1/4})$, the query complexity given by Corollary 10.

**Theorem 13.** *There is a one-sided error tester for isometric embeddability into the line that queries $O(\sqrt{n/\varepsilon})$ points.*

*Proof.* Consider first the following algorithm. Query $O(1/\varepsilon)$ random points. If all points in the sample are identical, accept the input. Otherwise, let $p$ and $q$ be the first two different drawn points in the sample. Place $p$ and $q$ on the line at distance $\delta(p,q)$. Now for any other point $r$ in the set, the placement of $p$ and $q$ uniquely determines the position of $r$ on the line, provided the subspace $\{p,q,r\}$ embeds into the line. Draw $O(\sqrt{n/\varepsilon})$ new points, and if for any point $r$ in the new sample, the subspace $\{p,q,r\}$ does not embed into the line, reject the input. Otherwise, place all the points from the sample on the line with respect to $p$ and $q$, and verify if all the pairwise distances on the line equal the distances in the original metric. If at least one of them is different, reject. Otherwise, accept.

We assume that $\varepsilon \geq 1/n$, since every set of points is either embeddable into the line, or is $1/n$-far from this property. This implies in particular that $1/\varepsilon = O(\sqrt{n/\varepsilon})$, and thus, the query complexity of the algorithm is $O(\sqrt{n/\varepsilon})$.

Let us prove that the above algorithm works. Clearly, it can only reject inputs that are not embeddable into the line. Suppose that an input is accepted by the above algorithm with probability at least $2/3$. We show that the input is $\varepsilon/2$-close to a set embeddable into the line. The input can be accepted in two different steps of the algorithm, and it must be accepted with probability at least $1/6$ in one of them. If it is accepted with probability at least $1/6$ because all points in the first sample are identical, the set must be $\varepsilon/2$-close to an input that consists of $n$ copies of a single point. Suppose now that it passes the other two tests with probability at least $1/6$ for arbitrary $p$ and $q$ fixed in the first phase of the algorithm. Let $S'$ be the maximum size subset of the input set such that each point $r$ in $S'$ embeds into the line with respect to $p$ and $q$, and all pairwise distances for the points in $S'$ are preserved in this embedding. We claim that $|S'| \geq (1-\varepsilon/2)|S|$, i.e., there is a subset of the input of size $(1-\varepsilon/2)n$ that isometrically embeds into the line. Firstly, the fraction of points in $S$ that do not embed with respect to $p$ and $q$ must be smaller than $\varepsilon/4$, since the constant hidden in the big-Oh notation is sufficiently large to detect every fraction greater than $\varepsilon/4$ of these points with probability greater than $5/6$. Secondly, the fraction of points of $S$ in $S \backslash S'$ that embed with respect to $p$ and $q$ also cannot be to large. Denote the set of those points by $U$. Suppose that $|U| \geq \varepsilon n/4$. Let $X_i = S' \cup U$, and iteratively create $X_i$ as follows. As long as $|X_i| > |S'|$, there is a pair of points $(a_i, b_i)$ in $X_i$ such that the distance between $a_i$ and $b_i$ changes after embedding into the line with respect to $p$ and $q$. We create $X_{i+1}$ by removing these two points from $X_i$. If $|U| \geq \varepsilon n/4$, there are at least $\varepsilon n/8$ such disjoint pairs of points, and by Fact 7, we find such a pair with probability greater than $5/6$. Hence the size of $T$ must be less than $\varepsilon n/4$, and the size of $S'$ is at least

$(1-\varepsilon/2)\cdot n$. Therefore, the input is $\varepsilon/2$-close to an input embeddable into the line, which finishes the proof of the correctness of the algorithm. $\square$

One can show that there is an algorithm that for a set of $s$ points, checks in time $O(s(T+\log s))$ if it exactly embeds into the line or not, where $T$ is the time complexity of computing the distance between two points.

**Corollary 14.** *There is a one-sided error tester for isometric embeddability into the line that queries $O(\sqrt{n/\varepsilon})$ points and runs in $O(\sqrt{n/\varepsilon}(T+\log n))$ time, where $T$ is the time necessary to compute the distance between two points.*

# 5 Lower Bounds

We give a number of lower bounds for testing. All of them follow from the same approach. We create a constant size gadget that is repeated several times. Until we read entirely at least one of the copies of the gadget, the subset of points has a considered property. At the same time the whole input is far from the property. A one-sided tester must therefore read an entire copy of the gadget, which requires many queries. One can also show that each of our lower bounds can be transformed into an $\Omega(\sqrt{n/\varepsilon})$ lower bound for two-sided testers.

## 5.1 A Lower Bound for Testing Dimension Reduction in $\ell_1$

A say that a set of points in $\ell_p^m$ is *$d$-dimensional* if it isometrically embeds into $\ell_p^d$. We now present a general lower bound for one-sided error testers, which shows that a $d$-dimensionality tester with one-side error must query many points for small $\varepsilon$ and large $d$. To prove the lower bound, we make use of a nonembeddability lemma by Hadlock and Hoffman [13]. They showed that to embed a tree metric into $\ell_1$ one needs exactly $\lceil k/2 \rceil$ dimensions, where $k$ is the number of leaves in the underlying tree. Here we only make use of the nonembeddability part of their result.

**Lemma 15 (Hadlock and Hoffman [13]).** *Let $\mathcal{M}=(S,\delta)$ be a tree metric of $k\geq 3$ leaves. $\mathcal{M}$ does not embed into $\ell_1^m$ for any $m<k/2$.*

**Theorem 16.** *Any one-sided tester for $d$-dimensionality must query $\Omega(n^{d/(d+1)}\cdot \varepsilon^{-1/(d+1)})$ points for $\varepsilon<1/(2d+2)$, even if the host space is $\ell_1^{d+1}$.*

*Proof.* A one-sided tester for inputs that are $\varepsilon$-far from $d$-dimensionality needs to detect with constant probability evidence of non-$d$-dimensionality. In our case, it must read with constant probability a subset of points that is not $d$-dimensional.

We will exhibit a $d+1$-dimensional set that is hard for one-sided testers. Before we pass this set to the tester, we randomly shuffle the list of the points. Thus we can assume that the tester reads random points from the set. (Bar-Yossef et al. [17] conduct an interesting analysis of testers for the properties that do not depend on the order of the elements in the input.)

We will define a set of points in $\ell_1^{d+1}$, that will not be $d$-dimensional. Let $\mathbf{e}_i$, $1 \leq i \leq d+1$, be the unit vector in $\mathbb{R}^{d+1}$ of the $i$-th coordinate equal to 1 and all the others equal to 0. Also define $\mathbf{1}$ and $\mathbf{0}$ to be the vectors of ones and zeros in all coordinates, respectively.

We construct an input set $S$ as follows. Let $p = \varepsilon n$. First, we add $n - p(2d+2)$ copies of $\mathbf{0}$. Then, for each $1 \leq i \leq p$, we add the following group $G_i$ of $2d+2$ points:

- $u_i = 3i \cdot \mathbf{1}$,
- $v_{ij} = 3i \cdot \mathbf{1} - \mathbf{e}_j$, for each $1 \leq j \leq d+1$,
- $w_{ij} = 3i \cdot \mathbf{1} + \mathbf{e}_j$, for each $1 \leq j \leq d$.

Note that each $G_i$ is the shortest-path metric of the unweighted star of $2d+1$ leaves. Thus, by Lemma 15, $G_i$ is not $d$-dimensional. To turn $S$ into a $d$-dimensional set, we need to remove at least one point from each $G_i$, therefore $S$ is $\varepsilon$-far from $d$-dimensionality. On the other hand, if we remove at least one point $v_{ij}$ for each $1 \leq i \leq p$, we get a $d$-dimensional set. Since all the points $v_{ij}$, for fixed $i$, are symmetric in terms of the distance to the other points, we can assume without loss of generality that we remove $v_{i,d+1}$ for each $i$. We can define a distance-preserving embedding $f$ of the remaining points into $\ell_1^d$:

$$f(x) = \begin{cases} \mathbf{0}, & \text{if } x = \mathbf{0}; \\ 3\frac{d+1}{d} i \cdot \mathbf{1}, & \text{if } x = u_i; \\ 3\frac{d+1}{d} i \cdot \mathbf{1} - \mathbf{e}_j, & \text{if } x = v_{ij}; \\ 3\frac{d+1}{d} i \cdot \mathbf{1} + \mathbf{e}_j, & \text{if } x = w_{ij}. \end{cases}$$

One can easily check that this embedding does preserve all the distances.

Moreover, this implies that to find evidence that $S$ is not $d$-dimensional, the tester needs to read all the $v_{ij}$ for some $i$. If the tester queries $q$ points and finds evidence with constant probability, it follows from Fact 8 that $p\left(\frac{q}{n}\right)^{d+1} = \Omega(1)$, which implies that $q = \Omega\left(\frac{n^{d/(d+1)}}{\varepsilon^{1/(d+1)}}\right)$. □

# 6 Streaming Testers

## 6.1 A Linear Lower Bound for the Exact Property Verification

We now give a sketch of how to prove a lower bound for the exact verification of properties in the streaming model. We omit many technical details. Each of our lower bounds for property testing can easily be turned into a lower bound for exactly checking a property in streaming. For each of those lower bounds, we design a size-$k$ gadget for some constant $k$. Whenever an entire copy of the gadget is present in the input, the input does not have the property. We can also break each of the gadgets into two parts of the same, or almost the same size such that when only one of the halves is present, it does not contradict the property. We start from an input that has $n/k$ copies of the gadget. One of the halves of each gadget is assigned to Alice, and the other one to Bob. Alice picks her set of points by selecting an arbitrary subset of her halves of gadgets. So does Bob. If Alice and Bob picked halves that compose to an entire copy of the gadget, the union of their sets of points does not have the property. Otherwise, it does. Clearly, we can now use any streaming algorithm for the exact property verification to give a protocol for set disjointness on the set $\{1, \ldots, n/k\}$. Alice first simulates the algorithm on her set of points, passes the intermediate state to Bob, and Bob continues the simulation on his set of points. In the worst case, Alice must pass at least $\Omega(n/k)$ bits to Bob, so the amount of space used by the streaming algorithm is at least $\Omega(n/k)$. We state a corollary for embeddability into the line.

**Lemma 17.** *The exact verification of embeddability into the line requires $\Omega(n)$ bits of space in the streaming model.*

### 6.2 A Lower Bound for Streaming Testers

The above approach can easily be modified to give a lower bound for streaming testers. Instead of $n/k$ different copies of the gadget, we now only have $1/(\varepsilon k)$ different copies, but we always repeat each of them $\varepsilon n$ times. Because of this, whenever the subsets of $\{1, \ldots, 1/(\varepsilon k)\}$ chosen by Alice and Bob intersect, there are $\varepsilon n$ copies of the gadget, which makes the set of points $\varepsilon$-far from the property. By the same argument as before, we get a lower bound of $\Omega(1/(\varepsilon k))$ bits of space. In particular, the following lower bound holds for embeddability into the line.

**Lemma 18.** *A streaming tester for embeddability into the line must use $\Omega(1/\varepsilon)$ bits of space.*

### 6.3 Algorithms

Note that if there is a property tester of query complexity $T$, then there is a streaming tester that keeps only $T$ points. It collects $T$ random points when it goes over the stream, and at the end simulates the property tester on the sample. Here, we show that the number of points kept can be decreased.

All our property testing algorithms look for a $k$-tuple of points that is used as (a part of) a certificate that the input does not have a property. There are always at least $\Omega(\varepsilon n/k)$ such $k$-tuples, if the input is $\varepsilon$-far from a property. The improvement comes from the fact that it suffices to draw the first $k-1$ points of one of the $k$-tuples, and then, going over the stream, check for each point if it complements a $k$-tuple. By Fact 7, we only need to collect $O(n^{(k-2)/(k-1)}\varepsilon^{-1/(k-1)})$ sample points from the stream as opposed to $O(n^{(k-1)/k}\varepsilon^{-1/k})$ samples in the property testing model.

Moreover, for testing embeddability into the line (testing tree metrics), we need two different fixed points (one fixed point). We can use for that the first two different points (the first point) of the stream. For embeddability into the line, we get the following lemma.

**Lemma 19.** *There is a one-sided error streaming tester for embeddability into the line that stores $\Omega(1/\varepsilon)$ points.*

**Acknowledgments.** The author would like to thank Alexandru Andoni and Ronitt Rubinfeld for useful comments on an early version of the paper.

## References

1. Johnson, W.B., Lindenstrauss, J.: Extensions of Lipschitz mappings into a Hilbert space. In: Conference in Modern Analysis and Probability (New Haven, 1982). Volume 26 of Comtemporary Mathematics., Providence, RI, American Mathematical Society (1984) 189–206
2. Indyk, P.: Algorithmic applications of low-distortion geometric embeddings. In: Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science. (2001) 10–33
3. Parnas, M., Ron, D.: Testing metric properties. Information and Computation **187**(2) (2003) 155–195
4. Krauthgamer, R., Sasson, O.: Property testing of data dimensionality. In: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms. (2003) 18–27
5. Rubinfeld, R., Sudan, M.: Robust characterizations of polynomials with applications to program testing. SIAM Journal on Computing **25**(2) (1996) 252–271
6. Goldreich, O., Goldwasser, S., Ron, D.: Property testing and its connection to learning and approximation. J. ACM **45**(4) (1998) 653–750
7. Abraham, I., Bartal, Y., Chan, H.T.H., Dhamdhere, K., Gupta, A., Kleinberg, J.M., Neiman, O., Slivkins, A.: Metric embeddings with relaxed guarantees. In: FOCS. (2005) 83–100
8. Muthukrishnan, S.: Data streams: algorithms and applications. Found. Trends Theor. Comput. Sci. **1**(2) (2005) 117–236
9. Feigenbaum, J., Kannan, S., Strauss, M., Viswanathan, M.: Testing and spot-checking of data streams. Algorithmica **34**(1) (2002) 67–80
10. Bandelt, H.J., Chepoi, V.: Embedding metric spaces in the rectilinear plane: a six-point criterion. Discrete & Computational Geometry **15**(1) (1996) 107–117
11. Menger, K.: Untersuchungen uber allgemeine Metrik. Mathematische Annalen **100** (1928) 75–163
12. Edmonds, J.: Embedding into $\ell_\infty^2$ is easy, embedding into $\ell_\infty^3$ is NP-complete. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. (2007) 522–531
13. Hadlock, F., Hoffman, F.: Manhattan trees. Utilitas Mathematica **13** (1978) 55–67
14. Kalyanasundaram, B., Schnitger, G.: The probabilistic communication complexity of set intersection. SIAM J. Discrete Math. **5**(4) (1992) 545–557
15. Razborov, A.A.: On the distributional complexity of disjointness. In: ICALP. (1990) 249–253
16. Waterman, M.S., Smith, T.F., Singh, M., Beyer, W.A.: Additive evolutionary trees. Journal of Theoretical Biology **64** (1977) 199–213
17. Bar-Yossef, Z., Kumar, R., Sivakumar, D.: Sampling algorithms: lower bounds and applications. In: Proceedings on 33rd Annual ACM Symposium on Theory of Computing. (2001) 266–275