On the Impossibility of Dimension Reduction in ℓ_1

Bo Brinkman* Moses Charikar†
Princeton University
35 Olden St.
Princeton, NJ 08540, USA
{brinkman,moses}@cs.princeton.edu

Abstract

The Johnson-Lindenstrauss Lemma shows that any n points in Euclidean space (with distances measured by the ℓ_2 norm) may be mapped down to $O((\log n)/\varepsilon^2)$ dimensions such that no pairwise distance is distorted by more than a $(1+\varepsilon)$ factor. Determining whether such dimension reduction is possible in ℓ_1 has been an intriguing open question. We show strong lower bounds for general dimension reduction in ℓ_1. We give an explicit family of n points in ℓ_1 such that any embedding with distortion δ requires $n^{\Omega(1/\delta^2)}$ dimensions. This proves that there is no analog of the Johnson-Lindenstrauss Lemma for ℓ_1; in fact embedding with any constant distortion requires $n^{\Omega(1)}$ dimensions. Further, embedding the points into ℓ_1 with $1+\varepsilon$ distortion requires $n^{\frac{1}{2}-O(\varepsilon\log(\frac{1}{\varepsilon}))}$ dimensions. Our proof establishes this lower bound for shortest path metrics of series-parallel graphs. We make extensive use of linear programming and duality in devising our bounds. We expect that the tools and techniques we develop will be useful for future investigations of embeddings into ℓ_1.

1. Introduction

Dimension reduction refers to mapping points in a high dimensional space to a space with low dimensions while approximately preserving some property of the original points. We will be interested in dimension reduction techniques that map ℓ_p^d to $\ell_p^{d'}$ and approximately preserve pairwise distances of points. Given metric spaces (M_1, d_1) and (M_2, d_2), an embedding $\sigma : M_1 \to M_2$ is said to be an embedding of M_1 into M_2 with distortion δ if $\forall u, v \in M_1$, $\frac{d_1(u,v)}{\delta} \le d_2(\sigma(u), \sigma(v)) \le d_1(u,v)$.

The fundamental result in this area is the Johnson-Lindenstrauss lemma [17] which shows that any set of n points in Euclidean space can be mapped down to $O((\log n)/\varepsilon^2)$ dimensions such that all distances are distorted by at most $1+\varepsilon$. Moreover, such a mapping can be computed with high probability by simply projecting the set of points on randomly chosen unit vectors. [1]

Metric embeddings have traditionally been studied by functional analysts, and have recently attracted a lot of attention in the theoretical computer science community due to connections to approximation algorithms and the design of efficient algorithms. Dimensionality reduction techniques using the Johnson-Lindenstrauss lemma and closely related methods have recently found numerous algorithmic applications: e.g. approximate searching for nearest neighbors [16, 18, 13], clustering of high dimensional point sets [8, 24], streaming computation [2, 14] and so on. (See the recent survey by Indyk [15].)

The Johnson-Lindenstrauss lemma has proved to be a particularly useful tool since the ℓ_2 norm is a commonly used norm in various settings. A natural question to ask is whether there exists an analogue of the Johnson-Lindenstrauss lemma for other ℓ_p norms. Surprisingly little is known about this question. In particular, the dimension reduction question for the ℓ_1 norm stands out and has attracted attention in several recent surveys on the subject of metric embeddings. This question is interesting both because of its inherent theoretical appeal as well as its potential algorithmic applications. Indyk [15], in his tutorial on algorithmic applications of embeddings in FOCS 01, asks: *"Is there an analog of JL lemma for other norms, especially ℓ_1? This would give a powerful technique for designing approximation algo-*

*Supported by a Frances Upton Fellowship and DOE grant DE-FG02-02ER25540.

†Research supported by NSF ITR grant CCR-0205594, DOE Early Career Principal Investigator award DE-FG02-02ER25540, NSF CAREER award CCR-0237113 and an Alfred P. Sloan Fellowship.

[1] The proof of the original result of Johnson and Lindenstrauss was subsequently simplified by a number of later works: Frankl and Maehara [11], Indyk and Motwani [16], Dasgupta and Gupta [9], Arriaga and Vempala [4] and Achlioptas [1].

rithms for ℓ_1 norms ...". Linial [20], in his article on finite metric spaces at the International Congress of Mathematicians in 2002, says the following about the "*mysterious ℓ_1*": "*We know much less about metric embeddings into ℓ_1, and the attempts to understand them give rise to many intriguing open problems ... What is the smallest $k = k(n, \varepsilon)$ so that every n-point metric in ℓ_1 can be embedded into ℓ_1^k with distortion $< 1 + \varepsilon$? We know very little at the moment, namely $\Omega(\log n) \le k \le O(n \log n)$ for constant $\varepsilon > 0$. The lower bound is trivial and the upper bound is from [26, 27].*"

Known results on dimension reduction: Ball [5] studied upper and lower bounds on the minimum dimension required for *isometric* embeddings in ℓ_p, proving linear lower bounds and quadratic upper bounds. The book by Deza and Laurent [10] gives a very good overview of the results in this area, particularly for isometric embeddings into ℓ_1 and ℓ_2. It is known that dimension reduction is not possible in the ℓ_∞ norm. In general, we need $\Omega(n)$ dimensions to represent a set of n points in ℓ_∞ with any distortion less than 3 [3, 21].

The only known dimensionality reduction theorem for ℓ_1 is due to Indyk [14]. He showed that there is an embedding from ℓ_1^d to $\ell_1^{d'}$ with $d' = (\log 1/\delta)^{O(1/\varepsilon)}$ such that distances do not increase with probability ε and distances do not decrease by a factor $(1+\varepsilon)$ with probability $1-\delta$. Note, however, that with probability $1-\varepsilon$, any distance can increase arbitrarily. In fact this holds for any ℓ_p norm with $p \in [1, 2]$. Kushilevitz, Ostrovsky and Rabani [18] showed a dimension reduction result for the Hamming cube of a different flavor: they give low dimensional embeddings that can distinguish between two specified distance thresholds.

In a recent paper, Charikar and Sahai [7] showed that one cannot hope to use *linear* embeddings for obtaining dimension reduction in ℓ_1. In particular they exhibited a set of $O(n)$ points in ℓ_1^n such that any linear embedding into ℓ_1^d incurs distortion $\Omega(\sqrt{n/d})$. They also constructed low dimensional low distortion embeddings for special classes of ℓ_1 embeddable metrics, including tree metrics and shortest path metrics of outer-planar graphs. The work of [7] introduced and used the notion of a *stretch-limited* embedding in proving lower bounds. We employ and make significant refinements to this proof technique in order to obtain our new results.

Our results: We show strong lower bounds for general dimension reduction in ℓ_1. We give an explicit family of n points in ℓ_1 such that any embedding with distortion δ requires $n^{\Omega(1/\delta^2)}$ dimensions. This proves that there is no analog of the Johnson-Lindenstrauss Lemma for ℓ_1; in fact embedding with any constant distortion requires $n^{\Omega(1)}$ dimensions. Further, embedding the points into ℓ_1 with $1+\varepsilon$ distortion requires $n^{\frac{1}{2}-O(\varepsilon\log(\frac{1}{\varepsilon}))}$ dimensions. Our proof establishes this lower bound for series-parallel graphs, indicating that the low distortion low dimensional embeddings constructed in [7] cannot be extended to this class.

Subsequent to our work, Lee and Naor have provided a more elementary proof of our result[19].

Organization. In Section 1.1, we give an overview of our proof technique. In Section 2, we introduce a family of series-parallel graphs and establish lower bounds on the number of dimensions required to embed them into ℓ_1 with a specified distortion. In Section 4, we show how the lower bound for the graph family implies a lower bound for dimensionality reduction in ℓ_1. We conclude in Section 5 with a discussion of open problems and directions for future work. All omitted proofs appear in the full version of the paper[2].

1.1. Overview of Proof Technique

We first describe *stretch-limited embeddings* introduced in [7].

Definition *(stretch-limited embedding)* A stretch-limited embedding σ of a metric (M, d) is an embedding of (M, d) into $M' = \{\rho_1, \rho_2, \dots, \rho_t\}$, a collection of t line metrics with weights $\{w_1, w_2, \dots, w_t\}$ such that $\sum_{i=1}^t w_i = 1$. The distance function d' is the weighted average of distances in the ρ_i,

$$d'(u, v) \stackrel{def}{=} \sum_{i=1}^{t} w_i |\rho_i(u) - \rho_i(v)|.$$

Furthermore, for all points u and v in the original metric, and $i \in \{1, \dots, t\}$, $|\rho_i(u) - \rho_i(v)| \le s \cdot d(u, v)$. This embedding is said to have stretch s.

In other words, a stretch-s embedding is a convex combination of line metrics where distances in any line metric can not be more than a factor s larger than distances in the original metric.

Claim 1 *The existence of a δ-distortion embedding of a metric M into ℓ_1^s implies the existence of a δ-distortion stretch-limited embedding of M with stretch s.*

Claim 2 *The existence of a δ-distortion stretch-limited embedding of a metric (M, d) with stretch s implies the existence of a $\delta(1+\varepsilon)$-distortion embedding of (M, d) into $\ell_1^{O(s\delta\log(n)/\varepsilon^2)}$.*

Note that stretch-s embeddings are more general than ℓ_1 embeddings with s dimensions: Not only will we allow them to have arbitrary dimension, but we allow them

[2]The full version may be found at `http://www.derandomized.org/`.

to use any convex combination of line metrics, not simply an average. By the results above, stretch-s limited embeddings are a good proxy for embeddings with s dimensions.

Proof Overview

A technique commonly used to prove lower bounds on the distortion for embedding one metric M_1 into $M_2 = \ell_1$ is the following: The basic idea is to find two (non-negative) linear combinations of distances: $\alpha_M = \sum \alpha_{ij} d_M(i,j)$ and $\beta_M = \sum \beta_{ij} d_M(i,j)$ where $\alpha_{ij}, \beta_{ij} \geq 0$. The intuition is that in embedding M_1 into ℓ_1, distances in α tend to expand while distances in β tend to contract. The goal is to prove that $\frac{\alpha_\sigma}{\beta_\sigma} \geq \delta \frac{\alpha_{M_1}}{\beta_{M_1}}$ for some $\delta > 1$, for all embeddings σ from M_1 to the line. This establishes a lower bound of δ on the distortion of embedding M_1 into $M_2 = \ell_1$.

In order to prove lower bounds for dimension reduction in ℓ_1, we adapt this technique. Firstly, low dimensional embeddings seem tricky to reason about. Instead we focus on low-stretch embeddings, exploiting the connection between stretch-limited embeddings and embeddings in low dimensions. Our goal is to prove a lower bound on the stretch s needed to achieve a given distortion δ.

A stretch-limited embedding with distortion δ must satisfy the property that no distance expands and distances contract by at most a factor of δ. By adding up these upper bound and lower bound inequalities (suitably weighted), we get a single inequality for a linear combination of pairwise distances that must be satisfied by the stretch-s limited embedding. Since this inequality is satisfied by a convex combination of line embeddings, it must be satisfied by one line embedding with stretch s. Our goal is to prove that there is no line embedding with stretch s that satisfies the derived inequality. [3]

Charikar and Sahai [7] use this proof technique to prove lower bounds for linear embeddings (although our exposition is a little different). In that case, the restriction to linear embeddings and a careful choice of the inequality on pairwise distances made it possible to prove a lower bound on the stretch s required. How can one prove lower bounds on the stretch for arbitrary (i.e. non-linear) line embeddings? Our innovation is to express the problem of minimizing stretch so as to satisfy the inequality on the pairwise distances as an LP. In general, such an LP formulation is not possible. However, we are able to obtain an LP that minimizes stretch for a carefully chosen family of points in ℓ_1 and a particular set of linear inequalities on pairwise distances. Having obtained the LP formulation, we consider the dual LP and exhibit a dual feasible solution. This establishes a lower bound on the stretch. [4]

One issue that we ignored here is how we select weights for the individual upper and lower bound constraints so as to obtain a combined constraint on a linear combination on pairwise distances. The correct weighting of the individual constraints is crucial to obtaining a good lower bound on the stretch s. In fact, the weights we use could be (and actually are) dependent on the target distortion δ that we would like to reason about. In general, how does one figure out the "best" set of weights to obtain a good bound? Again, LP duality comes to the rescue. We show that one can consider an LP with upper and lower bounds on the average lengths of groups of pairwise distances, instead of a single linear inequality. The dual solution to this LP specifies weights for each of these upper and lower bound constraints (these are simply the values of the corresponding dual variables). These weights (in general, functions of the target distortion δ), can then be used to obtain a single hard constraint. Actually, we will present our proof by directly obtaining this single hard constraint without explaining the origin of the weights we use for the upper and lower bounds on individual distances.

2. The recursive diamond graph

In order to prove our results, we will focus on one particular family of series-parallel graphs which we call the recursive diamond graphs. Newman and Rabinovich previously studied these graphs in [23], and used them to establish a $\sqrt{\log n}$ lower bound for embedding planar graphs into ℓ_2. The order 0 recursive diamond graph is a single edge, with length one. In order to make the

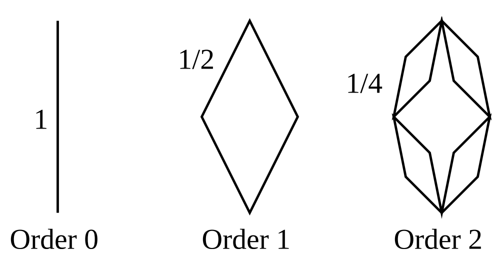

Figure 1. Recursive diamond graphs of different orders

[3] This method for proving a lower bound on the stretch is universal. If for every such valid inequality we can derive on linear combinations of pairwise distances, there is a stretch-s line embedding that satisfies this inequality, then there is a stretch-s embedding that satisfies all the required bounds on pairwise distances [7].

[4] In fact, our dual solution was obtained by examining and extrapolating from actual CPLEX solutions to large LPs.

order k graph from the order $k-1$ graph, replace each edge of length $1/2^{k-1}$ with a four-edge diamond, with edges of length $1/2^k$ (See Figure 1). This is a family of series-parallel graphs with 4^k edges and $\Theta(4^k)$ vertices. Furthermore, the work of Gupta, Newman, Rabinovich and Sinclair [12] shows that this graph can be embedded into ℓ_1 with constant distortion (with many dimensions).

We will need some terminology in order to talk about the graph (see Figure 2).

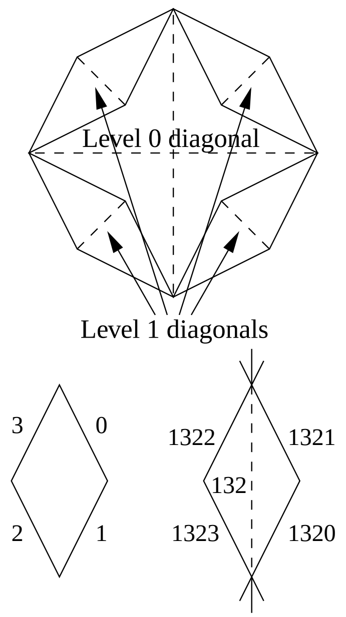

Figure 2. Diamond graph terminology and labels for edges

We will use n to refer to the number of vertices in a given diamond graph, and k to refer to the order (number of levels) of the graph. Each vertex has some k such that it is present in the order k graph, but not in the order $k-1$ graph. We will refer to any vertex as a *level* k vertex if it first appears in the order k graph. When an edge is replaced with a diamond, the two new vertices that are created will be called *siblings*, and we will refer to the pair of siblings as the *diagonal* of this diamond. We will say that it is a level k diagonal if the vertices concerned are level k vertices. Finally, there is a natural parent-child relationship between diamonds of different levels: A diamond is a child of the diamond whose edge it replaces. An ancestor of a diamond is defined in the obvious way, and the ancestors of an edge are the diamonds of each order in which the edge participates.

Every edge in the graph is labeled by a string. A particular diamond has four edges, which we will number 0,1,2 and 3. The label for the ith edge of a diamond is obtained by concatenating i to the label of its ancestor edge. We label diamonds with the label of the parent edge. Also, we use this same label to label the diagonal edge. For a label x, edge e_x denotes the edge labeled by x. Also, f_x denotes the diagonal whose label is x. This leaves the original edge of the graph unlabeled. We will treat it as being "diagonal like" and refer to it as f_*. We will return to the matter of exactly specifying the labeling in a later section.

3. Series-parallel graphs need high dimension for constant distortion

We may now state our first result.

Theorem 1 *A recursive diamond graph on n vertices requires $n^{\Omega(1/\delta^2)}$ dimensions to embed in ℓ_1 with distortion at most δ.*

3.1. Proof

We focus on the n-vertex diamond graph (which has k levels). Consider E, the set of all edges in the graph, and F, the set of all diagonals. We will bound δ by showing that the edges tend to expand while the diagonals tend to contract (refer back to the proof overview).

The δ-distortion constraint

We first develop our key constraint on edge and diagonal lengths imposed by δ. Recall that we have labeled the edges and diagonals. We refer to the length of the edge labeled x as m'_x and the length of the diagonal labeled y as d_y. (We use m'_x here because we later define a variable m_x which is the *signed* length of an edge.)

We assumed w.l.o.g. that our embedding σ is non-expansive and has distortion at most δ. The non-expansive property of σ implies that

$$-\sum_{x\in\{0,1,2,3\}^k} m'_x/2^k \geq -1,$$

while the δ-distortion property implies that

$$\delta\left(d_* + \sum_{i=0}^{k-1}\sum_{y\in\{0,1,2,3\}^i} d_y/2^i\right) \geq k+1.$$

We combine these constraints to get a single constraint (referred to as the δ constraint) that should be hard to satisfy. $\forall\gamma\geq 0$:

$$\delta\left(d_* + \sum_{i=0}^{k-1}\sum_{y\in\{0,1,2,3\}^i} d_y/2^i\right) - \gamma\sum_{x\in\{0,1,2,3\}^k} m'_x/2^k \quad\geq\quad k+1-\gamma.$$

We will eventually optimize γ in order to make this bound as strong as possible.

If this is true for a convex combination of line metrics, then it must be true for at least one of those line metrics. We will show a lower bound on the stretch s which must be incurred by a line metric which satisfies this constraint given values of n, k and δ.

Constraints on edges and diagonals

Before we can write down our LP, we will need a few more constraints. There is a very strong relationship between the length of an edge and the lengths of the diagonals of the edge's ancestor diamonds, and this will give us a second set of constraints.

We first precisely specify the labeling scheme for the edges of the recursive diamond graph. Consider a particular line metric of σ, call it ρ, which satisfies the δ inequality. We know by the above argument that at least one such line metric must exist. The labeling scheme we choose will depend on the particular ρ we are considering, and we will describe how to choose a labeling which satisfies our needs for any given ρ. The rest of this section deals exclusively with ρ, which we can view as an embedding into a single line.

For each edge e, we will designate one end point as the *head* (denoted by $\mathrm{hd}(e)$) and the other as the *tail* (denoted by $\mathrm{tl}(e)$). For every edge e in the graph, we will obtain an expression for $\rho(\mathrm{hd}(e)) - \rho(\mathrm{tl}(e))$. Consider a diamond with parent edge e_p and diagonal edge e_d. The end points of the diagonal edge e_d are labeled as the *top* end point (denoted by $\mathrm{tp}(e_d)$) and the *bottom* end point (denoted by $\mathrm{bt}(e_d)$). This labeling is done such that $\rho(\mathrm{tp}(e_d)) \geq \rho(\mathrm{bt}(e_d))$ (ties are broken arbitrarily).

The edges of the diamond connect the end points of the parent edge e_p to the end points of the diagonal edge e_d. The edge connecting $\mathrm{hd}(e_p)$ to $\mathrm{tp}(e_d)$ is called the 0-edge. The edge connecting $\mathrm{tp}(e_d)$ to $\mathrm{tl}(e_p)$ is called the 1-edge. The edge connecting $\mathrm{bt}(e_d)$ to $\mathrm{tl}(e_p)$ is called the 2-edge. The edge connecting $\mathrm{hd}(e_p)$ to $\mathrm{bt}(e_d)$ is called the 3-edge. Further, $\mathrm{hd}(e_p)$ is considered the *head* for the 0-edge and the 3-edge; for each of these edges, the end point of e_d incident on it is considered the *tail*. $\mathrm{tl}(e_p)$ is considered the *tail* for the 1-edge and the 2-edge; for each of these edges, the end point of e_d incident on it is considered the *head*.

We define the following:

$$
\begin{aligned}
m_x &= \rho(\mathrm{hd}(e_x)) - \rho(\mathrm{tl}(e_x)) \\
d_x &= \rho(\mathrm{tp}(f_x)) - \rho(\mathrm{bt}(f_x)) \\
n_x &= \frac{\rho(\mathrm{tp}(f_x)) + \rho(\mathrm{bt}(f_x))}{2} \\
&\quad - \frac{\rho(\mathrm{hd}(e_x)) + \rho(\mathrm{tl}(e_x))}{2}
\end{aligned}
$$

Note that $|m_x| = m'_x$ is the length of the edge e_x (we allow m_x to be negative when $\mathrm{hd}(e_x) < \mathrm{tl}(e_x)$) in ρ and d_x is the length of the diagonal f_x in ρ. We refer to n_x as the *offset* of the diamond labeled x in ρ.

Now, we can calculate m_{x0}, m_{x1}, m_{x2} and m_{x3} in terms of m_x, d_x and n_x as follows:

Lemma 1

$$
m_{x0} = \frac{m_x}{2} - \frac{d_x}{2} - n_x \qquad m_{x1} = \frac{m_x}{2} + \frac{d_x}{2} + n_x
$$
$$
m_{x2} = \frac{m_x}{2} - \frac{d_x}{2} + n_x \qquad m_{x3} = \frac{m_x}{2} + \frac{d_x}{2} - n_x
$$

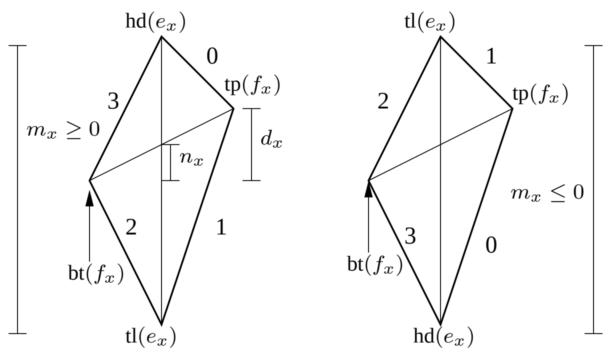

Figure 3. Single dimension of embedded diamond

Using this, one can obtain an expression for m_x in terms of the diagonal lengths and the offsets of the diamonds which are ancestors of e_x. We use $y \sqsubset x$ to denote that y is a prefix of x, and the empty string is a prefix of every string. We also use $|x|$ to denote the length of the string x.

Lemma 2

$$
\begin{aligned}
m_x &= \frac{d_*}{2^{|x|}} + \sum_{y \sqsubset x} S(x_{|y|+1}) \frac{d_y}{2^{|x|-|y|}} \\
&\quad + \sum_{y \sqsubset x} T(x_{|y|+1}) \frac{n_y}{2^{|x|-|y|-1}}
\end{aligned}
$$

where $S(0) = S(2) = -1, S(1) = S(3) = +1$ *and* $T(0) = T(3) = -1, T(1) = T(2) = +1$.

3.2. Grouping edges and diagonals

Before we continue, we would like to remove the dependence on n_y. We noticed in experiments that optimal embeddings had all the n_y set to zero, so we expect removing the n_y should not hurt our bounds much. In fact, we will place our edges and diagonals into groups, and write our constraints in terms of the average distances in these groups. The careful choice of our labeling will cause the n_y terms to cancel out.

In particular, we group edges into 2^k groups of 2^k edges each. Groups are identified with labels in $\{0,1\}^k$. For a group labeled by $z \in \{0,1\}^k$, an edge e_x belongs to the group if $x(mod\ 2) = z$. Here $x(mod\ 2)$ refers to the label obtained by performing a coordinate-wise $mod\ 2$ operation. Similarly, diagonals of level i are grouped into 2^i groups, identified with labels in $\{0,1\}^i$.

$$\begin{aligned}\overline{m_z} &= \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}} |m_x| \\ \overline{d_z} &= \frac{1}{2^i}\sum_{\{x:x(mod\ 2)=z\}} d_x\end{aligned}$$

In other words, $\overline{m_z}$ and $\overline{d_z}$ are the average lengths of their constituent edges and diagonals.

We can immediately rewrite our δ constraint in terms of $\overline{m_z}$ and $\overline{d_z}$ without changing anything.

$$\delta\left(\overline{d_*} + \sum_{i=0}^{k-1}\sum_{y\in\{0,1\}^i}\overline{d_y}\right) - \gamma\sum_{x\in\{0,1\}^k}\overline{m_x} \\ \geq k+1-\gamma$$

Claim 3 *For a group label $z \in \{0,1\}^k$,*

$$\frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}} m_x = \frac{d_*}{2^k} + \sum_{y\sqsubset z} S(z_{|y|+1})\frac{\overline{d_y}}{2^{k-|y|}}$$

Proof Using Lemma 2, the value of the LHS is as follows:

$$\begin{aligned}\text{LHS} &= \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}}\left(\frac{d_*}{2^k} + \sum_{y\sqsubset x}\frac{S(x_{|y|+1})d_y}{2^{k-|y|}}\right. \\ &\quad \left. + \sum_{y\sqsubset x}\frac{T(x_{|y|+1})n_y}{2^{k-|y|-1}}\right) \\ &= \frac{d_*}{2^k} + E_1(z) + E_2(z)\end{aligned}$$

where

$$\begin{aligned}E_1(z) &= \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}}\sum_{y\sqsubset x}\frac{S(x_{|y|+1})d_y}{2^{k-|y|}} \\ E_2(z) &= \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}}\sum_{y\sqsubset x}\frac{T(x_{|y|+1})n_y}{2^{k-|y|-1}}\end{aligned}$$

We now simplify the two expressions $E_1(z)$ and $E_2(z)$.

$$\begin{aligned}E_1(z) &= \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}}\sum_{y\sqsubset x}\frac{S(x_{|y|+1})d_y}{2^{k-|y|}} \\ &= \frac{1}{2^{|y|}}\sum_{y\sqsubset z}\sum_{\{y':y'(mod\ 2)=y\}}\frac{S(z_{|y|+1})d_{y'}}{2^{k-|y|}} \\ &= \sum_{y\sqsubset z}\frac{S(z_{|y|+1})\overline{d_y}}{2^{k-|y|}}\end{aligned}$$

$$E_2(z) = \frac{1}{2^k}\sum_{\{x:x(mod\ 2)=z\}}\sum_{y\sqsubset x}\frac{T(x_{|y|+1})n_y}{2^{k-|y|-1}}$$

$$= \frac{1}{2^{|y|+1}}\sum_{y\ \sqsubset\ z}\sum\frac{(T(z_{|y|+1}) + T(2+z_{|y|+1}))n_{y'}}{2^{k-|y|-1}} \\ = 0$$

The second sum is over $\{y' : y'(mod\ 2) = y\}$ (the subscript was removed to aid typesetting). The last equality follows from the fact that $T(i) + T(2+i) = 0$ for $i \in \{0,1\}$. Substituting the values of $E_1(z)$ and $E_2(z)$ in the expression we derived earlier proves the claim.

□

Lemma 3 *For a group label $z \in \{0,1\}^k$,*

$$\begin{aligned}\overline{m_z} &\geq \frac{d_*}{2^k} + \sum_{y\sqsubset z} S(z_{|y|+1})\frac{\overline{d_y}}{2^{k-|y|}} \\ \overline{m_z} &\geq -\frac{d_*}{2^k} - \sum_{y\sqsubset x} S(z_{|y|+1})\frac{\overline{d_y}}{2^{k-|y|}}\end{aligned}$$

These inequalities result when we replace the m_x in Claim 3 with the $|m_x|$ from the definition of $\overline{m_z}$.

Linear program for minimizing stretch

We have already derived three of the four constraints that we will use in our linear program. All that remains is to provide a lower bound for stretch.

Consider the stretch incurred by an edge e_x in the dimension ρ. For every edge $e_x = (u,v)$, $s \geq |m_x|/d(u,v) \geq 2^k|m_x|$, where d is understood to be the distance function for the original metric. Since $\max_{\{x:x(mod\ 2)=z\}}|m_x| \geq \overline{m_z}$, we conclude that $\forall z \in \{0,1\}^k\ s \geq 2^k\overline{m_z} \Rightarrow s/2^k - \overline{m_z} \geq 0$.

Now we are ready to give our linear program (see Table 1). Note that we will optimize γ later, but that it is constant with respect to the variables of the LP. We provide the names of the dual variables in brackets for reference. We have carefully derived our constraints so that we can see that the solution to our LP is no larger than the minimum stretch needed to embed the recursive diamond graph into ℓ_1.

Table 1. The linear program

$\min s$			
$\delta\left(\overline{d_*}+\sum_{i=0}^{k-1}\sum_{y\in\{0,1\}^i}\overline{d_y}\right)-\gamma\sum_{z\in\{0,1\}^k}\overline{m_z}$		$\geq k+1-\gamma$	$[\mu]$
$\forall z\in\{0,1\}^k$	$\frac{s}{2^k}-\overline{m_z}$	≥ 0	$[p_z]$
$\forall z\in\{0,1\}^k$	$\overline{m_z}+\left(\frac{\overline{d_*}}{2^k}+\sum_{y\sqsubset z}\frac{S(z_{\lvert y\rvert+1})\overline{d_y}}{2^{k-\lvert y\rvert}}\right)$	≥ 0	$[\alpha_z]$
$\forall z\in\{0,1\}^k$	$\overline{m_z}-\left(\frac{\overline{d_*}}{2^k}+\sum_{y\sqsubset z}\frac{S(z_{\lvert y\rvert+1})\overline{d_y}}{2^{k-\lvert y\rvert}}\right)$	≥ 0	$[\beta_z]$

Table 2. The dual linear program

$\max\ (k+1-\gamma)\mu$			
$\forall z\in\{0,1\}^k$	$-\gamma\mu-p_z+\alpha_z+\beta_z$	≤ 0	$[\overline{m_z}]$
	$\sum_{z\in\{0,1\}^k}p_z$	$\leq 2^k$	$[s]$
$\forall y\in\bigcup_{i\in[0,k-1]}\{0,1\}^i$	$\delta\mu+\sum_{v\in\{0,1\}^{k-\lvert y\rvert}}\frac{S((yv)_{\lvert y\rvert+1})(\alpha_{yv}-\beta_{yv})}{2^{k-\lvert y\rvert}}$	≤ 0	$[\overline{d_y}]$
	$\delta\mu+\sum_{z\in\{0,1\}^k}\frac{\alpha_z-\beta_z}{2^k}$	≤ 0	$[\overline{d_*}]$

Dual linear program for the lower bound on stretch

We have formulated an LP minimization problem whose optimum value is a lower bound on the minimum stretch for a δ-distortion embedding. In order to prove our lower bound we give the dual of this LP and a feasible solution. We construct the dual in the normal way (see Table 2).

Next we give our solution for this LP. In fact, our solution is very simple. Every variable is just a constant multiple of μ: $p_x=p_x^*\mu$, $\alpha_x=\alpha_x^*\mu$ and $\beta_x=\beta_x^*\mu$. We will specify the values of these constants, and then maximize μ subject to the constraints of the dual in order to get our bound. For these purposes, we can rewrite the dual LP (see Table 3).

The dual solution

We now give our solution to the dual in Table 4. We use $\|x\|_1$ to denote the number of 1s in the 0-1 string x.

Claim 4 *The values of γ, α_z^*, β_z^* and p_z^* in Table 4 give a feasible solution for our dual LP.*

Proof First check the $\overline{m_z}$ constraint, that $(-\gamma-p_z^*+\alpha_z^*+\beta_z^*)\leq 0$ for all z. We break this into three cases.
Case 1: $\|z\|_1\leq k/2-k/2\delta-1$

$$\begin{aligned}-p_z^*+\alpha_z^*+\beta_z^* &= \delta(k-1-2\|z\|_1)-\\ &\quad -2\delta(k/2-k/2\delta-\|z\|_1)\\ &= \gamma\end{aligned}$$

Case 2: $\|z\|_1\geq k/2+k/2\delta$

$$\begin{aligned}-p_z^*+\alpha_z^*+\beta_z^* &= \delta(2\|z\|_1-k+1)\\ &\quad -2\delta(\|z\|_1-k/2-k/2\delta+1)\\ &= \gamma\end{aligned}$$

Case 3: $k/2-k/2\delta-1<\|z\|_1<k/2+k/2\delta$

$$\begin{aligned}p_z^* &= 0\\ \alpha_z^* &\leq \delta(k-1-2(k/2-k/2\delta))=k-\delta=\gamma\\ \beta_z^* &\leq \delta(2(k/2+k/2\delta-1)-k+1)=k-\delta=\gamma\end{aligned}$$

Since the ranges where α_x^* and β_x^* are positive do not overlap, this proves that the $\overline{m_z}$ constraint is satisfied.

Now let us skip to the $\overline{d_y}$ constraint. In order to prove this, we will use the following lemma:

Lemma 4

$$\forall y\in\bigcup_{i\in[0,k-1]}\{0,1\}^k, v\in\{0,1\}^{k-i-1}$$

$$\begin{aligned}&S((y0v)_{\lvert y\rvert+1})(\alpha_{y0v}^*-\beta_{y0v}^*)+\\ &\quad S((y1v)_{\lvert y\rvert+1})(\alpha_{y1v}^*-\beta_{y1v}^*) = -2\delta\end{aligned}$$

Proof (of Lemma 4) $S((y0v)_{\lvert y\rvert+1})=-1$ and $S((y1v)_{\lvert y\rvert+1})=+1$, so

$$\begin{aligned}&S((y0v)_{\lvert y\rvert+1})(\alpha_{y0v}^*-\beta_{y0v}^*)\\ &+S((y1v)_{\lvert y\rvert+1})(\alpha_{y1v}^*-\beta_{y1v}^*) =\\ &-\alpha_{y0v}^*+\beta_{y0v}^*+\alpha_{y1v}^*-\beta_{y1v}^*.\end{aligned}$$

Since $\|y0v\|_1+1=\|y1v\|_1$, there are three cases.
Case 1: $\|y0v\|_1,\|y1v\|_1\leq k/2-1$

$$\begin{aligned}-\alpha_{y0v}^*+\beta_{y0v}^*+\alpha_{y1v}^*-\beta_{y1v}^* &=\\ -\alpha_i^*+\alpha_{i+1}^* &=\\ -\delta(k-1-2i)+\delta(k-1-2(i+1)) &= -2\delta\end{aligned}$$

Case 2: $\|y0v\|_1,\|y1v\|_1\geq k/2$

$$\begin{aligned}-\alpha_{y0v}^*+\beta_{y0v}^*+\alpha_{y1v}^*-\beta_{y1v}^* &=\\ \beta_i^*-\beta_{i+1}^* &=\\ \delta(2i-k+1)-\delta(2(i+1)-k+1) &= -2\delta\end{aligned}$$

Table 3. Dual with μ factored out

$$\begin{array}{|llll|}\hline \max\ (k+1-\gamma)\mu & & & \\ \forall z\in\{0,1\}^k & \mu\left(-\gamma-p_z^*+\alpha_z^*+\beta_z^*\right) & \le 0 & [\overline{m_z}] \\ & \mu\left(\sum_{z\in\{0,1\}^k} p_z^*\right) & \le 2^k & [s] \\ \forall y\in\bigcup_{i\in[0,k-1]}\{0,1\}^i & \mu\left(\delta+\sum_{v\in\{0,1\}^{k-|y|}}\frac{S((yv)_{|y|+1})(\alpha_{yv}^*-\beta_{yv}^*)}{2^{k-|y|}}\right) & \le 0 & [\overline{d_y}] \\ & \mu\left(\delta+\sum_{z\in\{0,1\}^k}\frac{\alpha_z^*-\beta_z^*}{2^k}\right) & \le 0 & [\overline{d_*}] \\ \hline\end{array}$$

Table 4. The dual solution

$$\begin{array}{|ll|}\hline \gamma = & k-\delta \\ \alpha_x^* = & \begin{cases}\delta(k-1-2\|x\|_1) & \text{if } \|x\|_1\le k/2-1\\ 0 & \text{otherwise}\end{cases} \\ \beta_x^* = & \begin{cases}\delta(2\|x\|_1-k+1) & \text{if } \|x\|_1\ge k/2\\ 0 & \text{otherwise}\end{cases} \\ p_x^* = & \begin{cases}2\delta(k/2-k/2\delta-\|x\|_1) & \text{if } \|x\|_1\le k/2-k/2\delta-1\\ 2\delta(\|x\|_1-k/2-k/2\delta+1) & \text{if } \|x\|_1\ge k/2+k/2\delta\\ 0 & \text{otherwise}\end{cases} \\ \hline\end{array}$$

Case 3: $\|y0v\|_1 = k/2-1, \|y1v\|_1 = k/2$

$$\begin{aligned} -\alpha_{y0v}^*+\beta_{y0v}^*+\alpha_{y1v}^*-\beta_{y1v}^* &= \\ -\alpha_i^*-\beta_{i+1}^* &= \\ -\delta-\delta &= -2\delta \end{aligned}$$

□

Applying lemma 4, we conclude that

$$\begin{aligned} \delta+\sum_{v\in\{0,1\}^{k-|y|}}\frac{S((yv)_{|y|+1})(\alpha_{yv}^*-\beta_{yv}^*)}{2^{k-|y|}} &= \\ \delta-\sum_{v'\in\{0,1\}^{k-|y|-1}}\frac{2\delta}{2^{k-|y|}} &= \\ \delta-\frac{(2^{k-|y|-1})(2\delta)}{2^{k-|y|}} &= 0 \end{aligned}$$

Hence the $\overline{d_y}$ constraints are all satisfied: In fact, they are all tight.

The case for the $\overline{d_*}$ constraint is even simpler because the sign for α_z^* is always positive and the sign for β_z^* is always negative. For every x with

$$\|x\|_1 = l \le k/2-1,$$

pair x with y such that $(x \operatorname{xor} y) = 111\ldots1$ (in other words, y is the bitwise NOT of x). Note that

$$\|y\|_1 = k-l \ge k/2+1,$$

and that

$$\alpha_l^*-\beta_{k-l}^* = \delta(k-1-2l)-\delta(2(k-l)-k+1) = -2\delta.$$

This accounts for all x except where $\|x\|_1 = k/2$. In this case α_* is 0, so we see that the $\overline{d_*}$ constraint is satisfied.

Finally, we return to the s constraint. Recall that our lower bound will be $(1+\delta)\mu$. This constraint is the only one which limits μ, and we will try to make μ as big as we can. Hence, $\mu = 2^k/\sum_z p_z^*$. Via a series of routine, but tedious calculations, we can show that

$$\sum_z p_z^* \le \delta^2(1+\delta)\binom{k}{k/2+k/2\delta}$$

$$\text{Hence, }\ \mu = \frac{2^k}{\sum_z p_z^*} \ge \frac{2^k}{\delta^2(1+\delta)\binom{k}{k/2+k/2\delta}}$$

$$\mathsf{LP}_{\mathsf{dual}} = (1+\delta)\mu \ge \frac{2^k}{\delta^2\binom{k}{k/2+k/2\delta}}$$

Using Stirling's approximation, we get a lower bound of $\Omega\left(\frac{1}{\delta^2}2^{k(1-H(\frac{1}{2}(1+\frac{1}{\delta})))}\right)$. Note that the number of points $n = \theta(2^{2k})$. For large δ, this bound becomes $\frac{1}{\delta^2}n^{\Omega(1/\delta^2)}$. For $\delta = 1+\varepsilon$ where ε is small, the bound becomes $n^{\frac{1}{2}-O(\varepsilon\log(1/\varepsilon))}$.

This concludes the proof of Theorem 1.

□

4. Exponential reduction in dimension is impossible in ℓ_1

So far we have proved that some series-parallel graphs do not admit low distortion, low dimension embeddings. This is in contrast to Gupta, Newman, Rabinovich and Sinclair [12] who prove that series-parallel graphs can be embedded into ℓ_1 with constant distortion (with high dimension). We can go one step further and provide a family of point sets native to ℓ_1 which have the same properties as the recursive diamond graph. This gives our final theorem:

Theorem 2 *There are ℓ_1 metrics of n points which require $n^{\Omega(1/\delta^2)}$ dimensions if only δ distortion is allowed.*

Proof We build our point set with a construction analogous to the construction of the recursive diamond graph. Let the original edge have end points at 0 and 1.

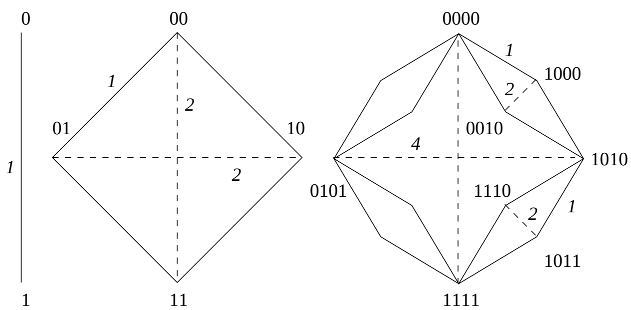

Figure 4. Using the diamond graph to generate our point set

Our "vertices" will be points in $\{0,1\}^i$ (that is, vertices of the hamming-cube). To go from level i to level $i+1$, first double the number of dimensions. The vertices of the parent edge are at the points x and y. Replace them with the points xx (x concatenated with x) and yy. The children will be the points xy and yx. The level-k recursive diamond graph corresponds to a set of $\Theta(4^{k+1})$ points in 2^{k+1} dimensions.

Claim 5 *Every "edge" in a level k point set has length 1.*

Claim 6 *Each diagonal at level i has length 2^{k-i} in the level k point set.*

If we divide all distances by 2^k this point set has exactly the same "edge lengths" and "diagonal lengths" as the recursive diamond graph. Since our constraints only depend on these distances, our lower bound for the recursive diamond graph immediately applies to this point set.

5. Conclusions

We have given the first proof that some point sets in ℓ_1 require a polynomial number of dimensions if only constant distortion is allowed. Our results show the following lower bounds on the dimension-distortion tradeoff for ℓ_1: (1) for any distortion δ, the number of dimensions $d = n^{\Omega(1/\delta^2)}$, and (2) for $\delta = 1+\varepsilon$, $d = n^{\frac{1}{2}-\varepsilon\log(\frac{1}{\varepsilon})}$. For distortion $1+\varepsilon$, the best upper bound on the number of dimensions is $O(n\log n)$ (for constant ε). Our lower bound is weaker because the set of n points that we use embeds isometrically into $O(\sqrt{n})$ dimensions. It should be possible to improve the lower bound using a different construction.

It would be very interesting to devise a dimension reduction scheme for ℓ_1 that requires $d = n^{f(\delta)}$ dimensions in order to guarantee distortion at most δ. Currently, we know of no non-trivial dimension-distortion tradeoff. One avenue for progress on this question is the following: Our lower bounds came from the dual solutions of certain LPs. and the primal solutions give stretch limited embeddings for the recursive diamond graph. Studying these embeddings may give clues for obtaining dimension reduction results for ℓ_1.

Another interesting question is whether our lower bound $n^{\Omega(1/\delta^2)}$ can be improved. This is connected to the following question: How much distortion is required to embed n points in ℓ_1 into ℓ_2? The current upper bound is $O(\log n)$ (by Bourgain [6]), while the best lower bound is $\Omega(\sqrt{\log n})$ (e.g. the Hamming cube with $\log n$ dimensions). It is known that n points in ℓ_2 can be embedded into $\ell_1^{O(\log n)}$ with distortion $(1+\varepsilon)$. If n points in ℓ_1 can be embedded into ℓ_2 with distortion δ, this would give ℓ_1 dimension reduction down to $O(\log n)$ dimensions with distortion δ. Note that $\delta = O(\sqrt{\log n})$ corresponds to a dimension-distortion tradeoff of $d = n^{O((\log\delta)/\delta^2)}$. We know that the recursive diamond graphs, and in fact all planar graphs, embed in ℓ_2 with distortion $O(\sqrt{\log n})$ by the result of Rao [25]. If we wish to significantly improve the $n^{\Omega(1/\delta^2)}$ lower bound, we will need to study non-planar graphs.

Another interesting question is to generalize our dimension reduction lower bound to all l_p for $p \in (1,2)$

References

[1] D. Achlioptas. Database-friendly random projections. In *Proc. 20th PODS*, 2001.

[2] N. Alon, Y. Matias, and M. Szegedy. The Space Complexity of Approximating the Frequency Moments. *JCSS* 58(1): 137-147, 1999

[3] J. Arias-de Reyna and L. Rodriguez-Piazza. Finite metric spaces needing high dimension for lipschitz embeddings in banach spaces. *Israel Journal of Mathematics*, 79:103–111, 1992.

[4] R. Arriaga and S. Vempala. An algorithmic theory of learning: Robust Concepts and Random Projection. In *Proc. 40th FOCS*, pp. 616-623, 1999.

[5] K. Ball. Isometric embedding in l_p-spaces. *European Journal of Combinatorics*, 11:305–311, 1990.

[6] J. Bourgain. On Lipschitz embedding of finite metric spaces into Hilbert space. *Israel Journal of Mathematics.* 52:46-52, 1985.

[7] M. Charikar and A. Sahai. Dimension reduction in the ℓ_1 norm. In *Proc. 43rd FOCS*, pp. 551-560, 2002.

[8] S. Dasgupta. Learning Mixtures of Gaussians. In *Proc. 40th FOCS*, pp. 634-644, 1999.

[9] S. Dasgupta and A. Gupta. An elementary proof of the Johnson-Lindenstrauss lemma. Technical Report TR-99-06, International Computer Science Institute, Berkeley, CA, 1999.

[10] M. Deza and M. Laurent. *Geometry of Cuts and Metrics*. Springer, 1997.

[11] P. Frankl and H. Maehara. The Johnson-Lindenstrauss lemma and the sphericity of some graphs. *J. Comb. Theory, Ser. B*, 44:355–362, 1988.

[12] A. Gupta, I. Newman, Y. Rabinovich and A. Sinclair. Cuts, trees and ℓ_1 embeddings of graphs. In *Proceedings of the 40th annual IEEE conference on foundations of computer science*, 1999.

[13] P. Indyk. Dimensionality reduction techniques for proximity problems. In *Proc. 9th SODA*, pp. 371-378, 2000.

[14] P. Indyk. Stable Distributions, Pseudorandom Generators, Embeddings and Data Stream Computation. In *Proc. 41st FOCS*, pp. 189-197, 2000.

[15] P. Indyk. Algorithmic applications of low-distortion embeddings. In *Proc. 42nd FOCS*, pp. 10-33, 2001.

[16] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proc. 30th STOC* pp. 604–613, 1998.

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

[18] E. Kushilevitz, R. Ostrovsky, and Y. Rabani. Efficient Search for Approximate Nearest Neighbor in High Dimensional Spaces. *SIAM J. Computing*, 30(2): 457-474, 2000.

[19] J. Lee, A. Naor. Embedding the Diamond Graph in L_p and Dimension Reduction in L_1. Personal communication, submitted for publication

[20] N. Linial. Finite Metric Spaces - Combinatorics, Geometry and Algorithms. In *Proc. International Congress of Mathematicians*, 2002.

[21] J. Matousek. On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93:333–344, 1996.

[22] J. Matousek. Embedding finite metric spaces into Euclidean spaces, chapter in *Lectures on discrete geometry*. Graduate Texts in Mathematics, Vol. 212, Springer Verlag, 2002.

[23] I. Newman and Y. Rabinovich A lower bound on the distortion of embedding planar metrics into Euclidean space. In *Proceedings of the eighteenth annual symposium on Computational geometry*, pp. 94-96, 2002.

[24] R. Ostrovsky and Y. Rabani Polynomial Time Approximation Schemes for Geometric k-Clustering. In *Proc. 41st FOCS*, pp. 349-358, 2000.

[25] S. Rao. Small distortion and volume preserving embeddings for planar and Euclidean metrics. In *Proceedings of the Symposium on Computational Geometry,* pages 300-306, 1999.

[26] G. Schechtman. More on embedding subspaces of L_p in ℓ_r^n. *Compositio Math.*, 61(2):159-169, 1987.

[27] M. Talagrand. Embedding subspaces of L_1 into ℓ_1^n. *Proc. Amer. Math. Soc.*, 108(2):363-369, 1990.

Appendix

A. Proofs of claims and lemmas

Proof of Claim 1

Proof Consider an embedding σ into ℓ_1^s. Let ρ_i denote the ith dimension of σ, which we think of as an embedding into a line. Let σ' be the stretch limited embedding with $\rho_i'(u) = s\rho_i(u)$, and let $w_i = \frac{1}{s}$. Let d denote ℓ_1 distance (in whatever number of dimensions is appropriate), and d' denote distance in the host space. Then:

$$\begin{aligned} d'(\sigma'(u), \sigma'(v)) &= \sum_{i=1}^{s} \frac{1}{s} d(\rho_i'(u), \rho_i'(v)) \\ &= \sum_{i=1}^{s} \frac{1}{s} \left(s d(\rho(u), \rho(v)) \right) \\ &= d(\sigma(u), \sigma(v)). \end{aligned}$$

Since distances are identical for σ and σ', their distortions must be equal.

□

Proof of Claim 2

Proof Consider a stretch-s embedding σ as a probability distribution on line metrics, where each line metric ρ_i has probability w_i. To get m dimensions we will sample m line metrics ρ_i from this distribution, and let dimension i be ρ_i/m.

Consider the distance of a particular pair of points u and v in a random ρ_i where i is picked with probability w_i. The expected distance is exactly the distance between u and v in the stretch-limited embedding which in turn is $\in [d(u,v)/\delta, d(u,v)]$. The stretch condition imposes a bound on the variance of this random variable. If we let $m = 8s\delta \log(n)/\varepsilon^2$, the probability that for a particular pairwise distance, the average over m samples is not within $(1 \pm \varepsilon)$ of its expectation is at most $1/n^2$. This follows from a standard Chernoff-Hoeffding bound. Since there are $n(n-1)/2$ pairs of points, $P[\text{relative error is at most } \varepsilon] \geq 1/2$. Hence there exists an embedding in ℓ_1^m with $m = 8s\delta \log(n)/\varepsilon^2$ and distortion at most $\delta(1+\varepsilon)$.[5]

□

Proof of Lemma 1

Proof We will show the calculation for m_{x0} only.

[5] Note that if we used the alternate definition of stretch discussed earlier, the claim can be strengthened to guarantee an embedding into $\ell_1^{O(s \log n/\varepsilon^2)}$.

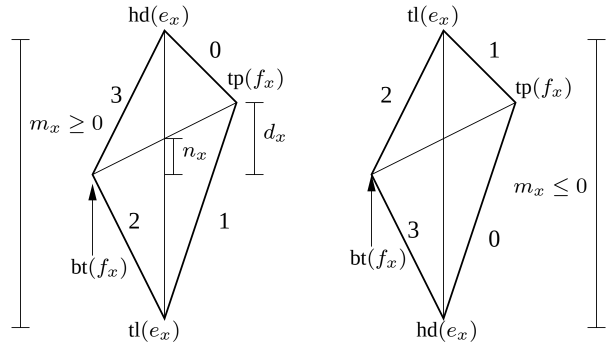

Figure 5. Single dimension of embedded diamond

$$\begin{aligned} m_{x0} &= \rho(\text{hd}(e_{x0})) - \rho(\text{tl}(e_{x0})) \\ &= \rho(\text{hd}(e_x)) - \rho(\text{tp}(f_x)) \\ &= \frac{\rho(\text{hd}(e_x)) - \rho(\text{tl}(e_x))}{2} \\ &\quad + \frac{\rho(\text{hd}(e_x)) + \rho(\text{tl}(e_x))}{2} \\ &\quad - \frac{\rho(\text{tp}(f_x)) - \rho(\text{bt}(f_x))}{2} \\ &\quad - \frac{\rho(\text{tp}(f_x)) + \rho(\text{bt}(f_x))}{2} \\ &= \frac{m_x}{2} - \frac{d_x}{2} \\ &\quad - \frac{\rho(\text{tp}(f_x)) + \rho(\text{bt}(f_x))}{2} \\ &\quad + \frac{\rho(\text{hd}(e_x)) + \rho(\text{tl}(e_x))}{2} \\ &= \frac{m_x}{2} - \frac{d_x}{2} - n_x \end{aligned}$$

The proofs for m_{x1}, m_{x2} and m_{x3} are similar.

□

Proof of Lemma 2

Proof We prove this by induction on $|x|$.

Base Case: Consider $|x| = 0$. In this case, $m_x = d_*$ and the statement is true.

Inductive Step: Suppose the statement is true for all x such that $|x| = i$. Now consider m_{x0}, where $|x| = i$. From Lemma 1, $m_{x0} = \frac{m_x}{2} - \frac{d_x}{2} - n_x$. Using the expression we have for m_x from the inductive hypothesis, we get:

$$
\begin{aligned}
m_{x0} &= \frac{1}{2}\left(\frac{d_*}{2^i} + \sum_{y \sqsubset x} \frac{S(x_{|y|+1})d_y}{2^{i-|y|}}\right. \\
&\quad \left. + \sum_{y \sqsubset x} \frac{T(x_{|y|+1})n_y}{2^{i-|y|-1}}\right) - \frac{d_x}{2} - n_x \\
&= \left(\frac{d_*}{2^{i+1}} + \sum_{y \sqsubset x} \frac{S(x_{|y|+1})d_y}{2^{i+1-|y|}}\right. \\
&\quad \left. + \sum_{y \sqsubset x} \frac{T(x_{|y|+1})n_y}{2^{i+1-|y|-1}}\right) - \frac{d_x}{2} - n_x \\
&= \frac{d_*}{2^{i+1}} + \sum_{y \sqsubset x0} \frac{S(x_{|y|+1})d_y}{2^{i+1-|y|}} \\
&\quad + \sum_{y \sqsubset x0} \frac{T(x_{|y|+1})n_y}{2^{i+1-|y|-1}}
\end{aligned}
$$

Thus the statement holds for m_{x0} as well. Similarly, we can show that the statement holds for m_{x1}, m_{x2} and m_{x3}. By induction, the statement of the lemma is true.

□

Proof of Lemma 3

Proof

$$
\begin{aligned}
\overline{m_z} &= \frac{1}{2^k} \sum_{\{x:x(mod\ 2)=z\}} |m_x| \\
&\geq \frac{1}{2^k} \sum_{\{x:x(mod\ 2)=z\}} m_x
\end{aligned}
$$

Using Claim 3, we get the first inequality we need to prove. Also,

$$
\begin{aligned}
\overline{m_z} &= \frac{1}{2^k} \sum_{\{x:x(mod\ 2)=z\}} |m_x| \\
&\geq -\frac{1}{2^k} \sum_{\{x:x(mod\ 2)=z\}} m_x
\end{aligned}
$$

Again, using Claim 3 gives the second inequality.

□

Proof of Claim 5

Proof We prove this by induction on the level of the point set.
Base case: The original edge in the point set is between 0 and 1.
Inductive step: By the inductive hypothesis, the end points x and y of an edge at level i has length 1. The four child edges at level $i+1$ are (xx, xy), (xx, yx), (yy, xy) and (yy, yx). Since $d(x, x) = d(y, y) = 0$ and $d(x, y) = 1$, $d(xx, xy) = d(xx, yx) = d(yy, xy) = d(yy, yx) = 1$.

□

Proof of Claim 6

Proof Again, we proceed by induction on the level of the point set.
Base case: The level 0 diagonal in the level 1 point set is from 01 to 10, which has length 2.
Inductive step: In the level j graph a level $i < j$ diagonal between points x and y has length 2^{j-i}. In the level $j+1$ graph these points are replaced with xx and yy. $d(xx, yy) = 2d(x, y) = 2^{j+1-i}$. The new diagonals in the level $j+1$ graph are at level j. A given new diagonal with parents x and y has end points xy and yx, and $d(xy, yx) = 2d(x, y) = 2 = 2^{j+1-j}$ by claim 5.

□

B. Determining weights for individual constraints via duality

Consider the LP we used for minimizing the stretch of a distortion δ embedding subject to the derived constraint on edges and diagonal lengths. The LP is presented in Table 5. In contrast to the LP we used in the proof in the main text, this LP uses weights $\lambda, \lambda_0, \ldots, \lambda_{k-1}, \gamma$ for the bounds on the average diagonal lengths and average edge lengths. We describe a technique to determine the *optimal* values for these weights.

In order to do this, we consider instead, a closely related LP (see Table 6) in which the combined constraint is replaced by individual constraints for the average edge length and average diagonal lengths at each level. A priori, it is not clear whether the bound on stretch obtained by this LP is a valid lower bound on the stretch for a δ distortion embedding. This is because our argument of the validity of the previous LP used the fact that we had a single combined constraint. This allowed us to conclude that there exists a single stretch-limited line embedding that satisfies this constraint. Nevertheless, we will prove that the bound produced by this new LP can be obtained from the previous LP by setting the weights appropriately. (The use of $\lambda, \lambda_0, \ldots, \lambda_{k-1}, \gamma$ for weights in Primal1 as well as dual variables for the constraints in Primal2 is deliberate.)

Consider now the dual to Primal2, given in Table 7. We look at any feasible solution dual solution to Dual2

Table 5. Primal1: The linear program with a combined constraint ($\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ are constants)

$\min s$			
$\delta\left(\lambda\overline{d_*} + \sum_{i=0}^{k-1}\lambda_i \sum_{y\in\{0,1\}^i}\overline{d_y}\right) - \gamma\sum_{z\in\{0,1\}^k}\overline{m_z}$		$\geq \lambda + \sum_{i=0}^{k-1}\lambda_i - \gamma$	$[\mu]$
$\forall z \in \{0,1\}^k$	$\frac{s}{2^k} - \overline{m_z}$	≥ 0	$[p_z]$
$\forall z \in \{0,1\}^k$	$\overline{m_z} + \left(\frac{\overline{d_*}}{2^k} + \sum_{y \sqsubset z}\frac{S(z_{\vert y\vert+1})\overline{d_y}}{2^{k-\vert y\vert}}\right)$	≥ 0	$[\alpha_z]$
$\forall z \in \{0,1\}^k$	$\overline{m_z} - \left(\frac{\overline{d_*}}{2^k} + \sum_{y \sqsubset z}\frac{S(z_{\vert y\vert+1})\overline{d_y}}{2^{k-\vert y\vert}}\right)$	≥ 0	$[\beta_z]$

Table 6. Primal2: The linear program with separate constraints ($\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ are dual variables)

$\min s$			
	$\sum_{z\in\{0,1\}^k}\overline{m_z}$	≤ 1	$[\gamma]$
	$\delta\overline{d_*}$	≥ 1	$[\lambda]$
$\forall i \in \{0,\ldots,k-1\}$	$\delta\sum_{y\in\{0,1\}^i}\overline{d_y}$	≥ 1	$[\lambda_i]$
$\forall z \in \{0,1\}^k$	$\frac{s}{2^k} - \overline{m_z}$	≥ 0	$[p_z]$
$\forall z \in \{0,1\}^k$	$\overline{m_z} + \left(\frac{\overline{d_*}}{2^k} + \sum_{y \sqsubset z}\frac{S(z_{\vert y\vert+1})\overline{d_y}}{2^{k-\vert y\vert}}\right)$	≥ 0	$[\alpha_z]$
$\forall z \in \{0,1\}^k$	$\overline{m_z} - \left(\frac{\overline{d_*}}{2^k} + \sum_{y \sqsubset z}\frac{S(z_{\vert y\vert+1})\overline{d_y}}{2^{k-\vert y\vert}}\right)$	≥ 0	$[\beta_z]$

and prove that this is a valid lower bound on the value of Primal1. In order to do this, we look at the values of the dual variables $\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ and use them as the values of the weights in Primal1.

Consider the dual to Primal1, given in Table 8. Note that $\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ are constants whose values are the same as the values of the corresponding variables in a specific feasible solution to Dual2. We claim that there exists a solution to Dual1 whose value is equal to the feasible solution to Dual2. In order to see this, we simply set $\mu = 1$ and use the same values for the rest of the variables as in the feasible solution to Dual2. It is easy to see that all the feasibility constraints are satisfied and the value of the two solutions is identical. This implies that Dual2 actually gives a lower bound on Primal1 for an appropriate setting of weights $\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$; moreover these weights are simply the values of dual variables in Dual2. This is in fact how we determined the weights used in the LP we presented in the main text.

Table 7. Dual2: The dual linear program for the primal with separate constraints ($\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ are variables)

$$
\begin{array}{lrll}
\max\ \lambda + \sum_{i=0}^{k-1} \lambda_i - \gamma & & & \\
\forall z \in \{0,1\}^k & -\gamma - p_z + \alpha_z + \beta_z & \leq 0 & [\overline{m_z}] \\
 & \sum_{z \in \{0,1\}^k} p_z & \leq 2^k & [s] \\
\forall y \in \bigcup_{i \in [0,k-1]} \{0,1\}^i & \delta\lambda_i + \sum_{v \in \{0,1\}^{k-|y|}} \frac{S((yv)_{|y|+1})(\alpha_{yv} - \beta_{yv})}{2^{k-|y|}} & \leq 0 & [\overline{d_y}] \\
 & \delta\lambda + \sum_{z \in \{0,1\}^k} \frac{\alpha_z - \beta_z}{2^k} & \leq 0 & [\overline{d_*}]
\end{array}
$$

Table 8. Dual1: The dual linear program for the combined constraint primal ($\lambda, \lambda_0, \ldots \lambda_{k-1}, \gamma$ are constants, μ is a variable)

$$
\begin{array}{lrll}
\max\ \left(\lambda + \sum_{i=0}^{k-1} \lambda_i - \gamma\right) \mu & & & \\
\forall z \in \{0,1\}^k & -\gamma\mu - p_z + \alpha_z + \beta_z & \leq 0 & [\overline{m_z}] \\
 & \sum_{z \in \{0,1\}^k} p_z & \leq 2^k & [s] \\
\forall y \in \bigcup_{i \in [0,k-1]} \{0,1\}^i & \delta\lambda_i\mu + \sum_{v \in \{0,1\}^{k-|y|}} \frac{S((yv)_{|y|+1})(\alpha_{yv} - \beta_{yv})}{2^{k-|y|}} & \leq 0 & [\overline{d_y}] \\
 & \delta\lambda\mu + \sum_{z \in \{0,1\}^k} \frac{\alpha_z - \beta_z}{2^k} & \leq 0 & [\overline{d_*}]
\end{array}
$$