A Treehouse with Custom Windows: Minimum Distortion Embeddings into Bounded Treewidth Graphs

Amir Nayyeri* Benjamin Raichel†

Abstract

We describe a $(1+\varepsilon)$-approximation algorithm for finding the minimum distortion embedding of an n-point metric space X into the shortest path metric space of a weighted graph G with m vertices. The running time of our algorithm is

$$m^{O(1)} \cdot n^{O(\omega)} \cdot (\delta_{opt}\Delta)^{\omega \cdot (1/\varepsilon)^{\lambda+2} \cdot \lambda \cdot (O(\delta_{opt}))^{2\lambda}}$$

parametrized by the values of the minimum distortion, δ_{opt}, the spread, Δ, of the points of X, the treewidth, ω, of G, and the doubling dimension, λ, of G.

In particular, our result implies a PTAS provided an X with polynomial spread, and the doubling dimension of G, the treewidth of G, and δ_{opt}, are all constant. For example, if X has a polynomial spread and δ_{opt} is a constant, we obtain PTAS's for embedding X into the following spaces: the line, a cycle, a tree of bounded doubling dimension, and a k-outer planar graph of bounded doubling dimension (for a constant k).

*School of Electrical Engineering and Computer Science; Oregon State University; nayyeria@eecs.oregonstate.edu; http://web.engr.oregonstate.edu/~nayyeria/.

†Department of Computer Science; University of Texas at Dallas; Richardson, TX 75080, USA; benjamin.raichel@utdallas.edu; http://utdallas.edu/~bar150630/.

1 Introduction

If (X, d_X) and (Y, d_Y) are two metric spaces, an ***embedding*** of X into Y is an injective map $f : X \to Y$, with the ***expansion*** e_f and the ***contraction*** c_f defined as follows.

$$e_f = \max_{\substack{x,x' \in X \\ x \neq x'}} \frac{d_Y(f(x), f(x'))}{d_X(x, x')}, \quad c_f = \max_{\substack{x,x' \in X \\ x \neq x'}} \frac{d_X(x, x')}{d_Y(f(x), f(x'))}.$$

Then, the ***distortion*** of f is defined as $\delta_f = e_f \cdot c_f$.

A low distortion embedding of X into Y is evidence that X is similar to (a subset of) Y. Therefore, computing embeddings is crucial for measuring the similarity of geometric objects in applications like pattern matching and pattern recognition. As is the case considered in this paper, in such settings X and Y are often finite point sets, sampled in some fashion from the objects being compared. In other areas of computer science, such as machine learning, low distortion embeddings are often used as a means to "simplify" the input metric, where in such cases Y is often considered to be a continuous host space such as $\mathbb{R}^d$. Moreover, data visualization requires low distortion embeddings into Euclidean spaces of dimension three or less.

Due to the wide range of applications, metric embeddings have been extensively studied in the last few decades, resulting in a rich array of results. Some notable results include: any n-point metric embeds into $\ell_2^{O(\log n)}$ with $O(\log n)$ distortion [Bou85], any n-point Euclidean metric embeds into $\ell_2^{O(\log n/\varepsilon^2)}$ with $1 + \varepsilon$ distortion [JL84], and any n-point metric embeds into a distribution of tree metrics with $O(\log n)$-distortion [Bar96, FRT03]. For a lot more results and applications we refer the interested reader to the survey papers by Matousek and Indyk [Mat13, IM04, Ind01].

Finite metrics into weighted graphs

Here, we seek approximation algorithms for embedding an n-point metric space (X, d_X) into a metric space (V_G, d_G), where $G = (V_G, E_G)$ is a given weighted undirected graph and d_G is its shortest path metric. One commonly considered case is when G is simply a weighted path, and we first briefly review results for this special case.

Embedding into a path. Perhaps the most well studied case is when X is the shortest path metric of an unweighted graph and G is further restricted to be an unweighted path. Badiou et al. [BDG+05] show that the problem is APX-hard, and describe an $O(\delta)$-approximation algorithm as well as an $n^{O(\delta)}$ exact algorithm, where δ is the minimum distortion. Fellows et al. [FFL+13] improved their exact algorithm to an $O(n\delta^4(2\delta + 1)^{2\delta})$ time exact algorithm. For embedding a general metric space into an unweighted path, Badiou et al. [BCIS05] provided an $O(\Delta^{3/4}\delta^{O(1)})$-approximation algorithm, where Δ is the spread of X. They also show it is hard to approximate the minimum distortion within any polynomial factor, even if X is a weighted tree with polynomial spread.

Recently, Nayyeri and Raichel [NR15] gave two improved results for embedding into weighted paths. First, when X is also a weighted path of the same cardinality (i.e. the embedding is bijective), a $\delta^{O(\delta^2 \log^2 D)} n^{O(1)}$ time exact algorithm was given, where D is the largest edge weight in X. The second result in [NR15], allowed X to be a general metric space (of potentially smaller cardinality), and gave a $\Delta^{O(\delta^2)}(mn)^{O(1)}$ time $O(1)$-approximation algorithm, where m is the number of vertices of the path, and Δ is the spread of X. Theorem 1.1 below, extends this approach to handle embedding into a much more general class of graphs than weighted paths. Moreover, it improves the $O(1)$ factor of approximation to $(1+\varepsilon)$. Moving to these more general settings is a far more challenging endeavor.

Embedding into more general graphs. The problem already becomes significantly more involved when G is a tree instead of a path.[1] Kenyon et al. [KRS04] developed an exact $O(n^2 \cdot 2^{\lambda^{\delta^3}})$ time algorithm to compute an optimal bijective embedding from an unweighted graph into an unweighted tree of maximum degree λ. Fellows et al. [FFL+13] furthered this result by developing a $n^2 \cdot |V(T)| \cdot 2^{O((5\delta)^{\lambda^{\delta+1}} \cdot \delta)}$ time algorithm for embedding an unweighted graph into an unweighted tree with maximum degree λ.

For more general graphs than trees, however, the authors are unaware of any similar results. Thus our main result, theorem 1.1 below, represents a significant step forward as it implies a $(1+\varepsilon)$-approximation algorithm for embedding general metric spaces into more general graphs, facilitated by parameterizing on the treewidth and doubling dimension.

Our Result

Theorem 1.1. *Let (X, d_X) be an n-point metric space with spread Δ. Let $G = (V_G, E_G)$ be a weighted graph of treewidth ω and doubling dimension λ, with m vertices. Let δ_{opt} be the minimum distortion of any embedding of X into V_G. There is a*

$$m^{O(1)} \cdot n^{O(\omega)} \cdot (\delta_{opt}\Delta)^{\omega \cdot (1/\varepsilon)^{\lambda+2} \cdot \lambda \cdot (O(\delta_{opt}))^{2\lambda}}$$

time $(1+\varepsilon)$-approximation algorithm for computing the minimum distortion embedding of X into G.

In particular, Theorem 1.1 yields a PTAS provided an X with polynomial spread, and the doubling dimensions of G, the treewidth of G, and δ_{opt}, are all constant. For example, if X has a polynomial spread and δ_{opt} is a constant, we obtain PTAS's for embedding X into the following spaces: the line, a cycle, a tree of bounded doubling dimension, and a k-outer planar graph of bounded doubling dimension (for a constant k). Note that for the line, a natural reduction discussed in [NR15], allows us to embed into the continuous space rather than a discrete subset.

2 Preliminaries

Subspaces. Let $G = (V_G, E_G)$ be a graph, and let $U \subseteq V_G$. We denote by $G[U]$ the ***induced*** subgraph by U. We use $\mathsf{diam}(U)$ to denote the diameter of $G[U]$, the maximum distance between any pair of vertices in $G[U]$. Note that the distance is in the induced subgraph, for example, if U is composed of two non-adjacent vertices then its diameter is not finite.

Given a metric space (X, d_X), and $x \in X$, the ball $B(x, R)$ (of radius R with center x) is a subset of X composed of all points in X that are at distance at most R from x. More generally, for a subset $X' \subseteq X$, we use the notation $B(X', R)$ to denote the set of all points in X that are at distance at most R from X', i.e. the union of balls centered at the points in X'.

[1] See references [BIS07, BDH+08, CDN+10] for results about embedding into trees. We emphasize that the goal in these papers is different as they look for the best tree that X can be embedded into, whereas in our setting the tree, G, is given in the input.

Tree decomposition and treewidth. A ***tree decomposition*** of a graph $G = (V_G, E_G)$ is a pair $(V_\mathcal{T}, \mathcal{T})$, where $V_\mathcal{T} = \{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_k\}$ is a family of subsets of V_G that are called ***bags***, and $\mathcal{T}$ is a tree whose vertex set is $V_\mathcal{T}$, with the following properties:

(1) $\bigcup_{\mathcal{B} \in V_\mathcal{T}} \mathcal{B} = V_G$.

(2) For each edge $(u, v) \in E_G$ there is a bag $\mathcal{B} \in V_\mathcal{T}$ that contains both u and v.

(3) For any $v \in V_G$, any pair $\mathcal{B}_i, \mathcal{B}_j \in V_\mathcal{T}$, and any $\mathcal{B}_t \in V_\mathcal{T}$ that is on the unique $\mathcal{B}_i$-to-$\mathcal{B}_j$ path in $\mathcal{T}$, if $v \in \mathcal{B}_i \cap \mathcal{B}_j$ then $v \in \mathcal{B}_t$.

The ***width*** of a tree decomposition is the size of its largest bag minus one. The ***treewidth*** of a graph is the minimum width among its valid tree decompositions. For example, a tree has treewidth one, a k-outerplanar graph has treewidth $O(k)$, and a planar graph with n vertices has treewidth $O(\sqrt{n})$ (implied by the separator theorem). The treewidth ω of a graph can be exactly computed in $\omega^{O(\omega^3)} n$ time by the algorithm of Bodlaender [Bod93]. In this paper, we use the following result to estimate the treewidth up to a constant factor. See Bodlaender et al. [BDD+13] for a survey of algorithms on treewidth computation.

Lemma 2.1 (Bodlaender et al. [BDD+13], Theorem I). *Let G be a graph with m vertices and treewidth ω. There exists a $2^{O(\omega)} m$ time algorithm to return a tree decomposition of G of width $O(\omega)$.*

Doubling Dimension. The ***doubling dimension*** of a metric space (X, d_X) is the smallest constant λ such that, for any $R \in \mathbb{R}^+$, any ball of radius R in X can be covered by at most 2^λ balls of radius $R/2$. An n-point metric space is ***doubling*** if its doubling dimension is a constant (independent of n). Doubling dimension was first introduced by Assouad [Ass83]. We find the following observation of Gupta et al. [GKL03] helpful in this paper.

Lemma 2.2 (Gupta et al. [GKL03], Proposition 1.1). *Let (Y, d_Y) be a metric with doubling dimension λ, and let $Y' \subseteq Y$. If all pairwise distances in Y' are at least ℓ, then any ball of radius R in Y contains at most $\left(\frac{2R}{\ell}\right)^\lambda$ points of Y'.* [2]

Total and partial maps. Let A and B be two sets. A partial map f from A to B is denoted by $f : A \nrightarrow B$. The domain of f, denoted by $\mathrm{Dom}(f)$, is the set of all $a \in A$ for which $f(a)$ is defined. So, $\mathrm{Dom}(f) \subseteq A$. The Image of f, denoted by $\mathrm{Im}(f)$, is the set of all $b \in B$ such that $b = f(a)$ for some $a \in A$. So, $\mathrm{Im}(f) \subseteq B$. In the special case that $A = \mathrm{Dom}(f)$, we call f a total map, or simply a map, and we denote it by $f : A \to B$.

Distortion. Let (X, d_X) and (Y, d_Y) be two metric spaces. An ***embedding*** of X into Y is an injective map $f : X \to Y$. The ***expansion*** e_f and the ***contraction*** c_f of f are defined as follows.

$$e_f = \max_{\substack{x,x' \in X \\ x \neq x'}} \frac{d_Y(f(x), f(x'))}{d_X(x, x')}, \quad c_f = \max_{\substack{x,x' \in X \\ x \neq x'}} \frac{d_X(x, x')}{d_Y(f(x), f(x'))}.$$

The ***distortion*** of f is defined as $\delta_f = e_f \cdot c_f$. It follows by definition that distortion is invariant under scaling of either of the sets. An embedding is ***non-contracting*** if the contraction is ≤ 1. The following lemma (whose variants can be found in previous papers (e.g., [KRS04])) allows us to restrict our attention to non-contracting embeddings of expansion $\leq \delta$, where δ is a known value.

[2] Note that λ in their paper is the doubling constant, whereas in this paper it denotes the doubling dimension.

Lemma 2.3 (Nayyeri and Raichel [NR15], Lemma 3.2). *Let (X, d_X) and (Y, d_Y) be finite metric spaces of sizes n and m, respectively. Then the problem of finding an embedding of X into Y with minimum distortion reduces to solving $(mn)^{O(1)}$ instances of the following problem: given a real value $\delta \geq 1$, compute a non-contracting embedding of X into Y with expansion at most δ, or correctly report that no such embedding exists.*

In this paper, we consider embedding an n-point metric space (X, d_X) into an m-point metric space (V_G, d_G) induced by the shortest path metric of a weighted graph $G = (V_G, E_G)$. Assume that for any pair $x, y \in X$, $1 \leq d_X(x, y) \leq \Delta$, and $\min_{x,y \in X} d_X(x, y) = 1$. Δ will be called the ***spread*** of X. Also assume, $|V_G| = m \geq n$. Let λ denote the doubling dimension of G.

Note that m, the size of G, can be much larger than n, the size of X. The following lemma breaks the problem of embedding X into G into m independent problems of embedding X into smaller subgraphs of G. As a result, we obtain a more efficient algorithm particularly when m is much larger than n.

Lemma 2.4. *There exists a ball B of radius $2\delta\Delta$ centered at a vertex of G, such that X embeds into $G[B]$ with distortion δ.*

Proof: Let $f : X \to V_G$ be a non-contracting embedding of expansion δ, and let $v \in V_G$ be any vertex that f maps onto. As the expansion is δ the image of f is contained in $B(v, \delta\Delta)$. Let $u, u' \in V_G$ be any two vertices that f maps onto, so $u, u' \in B(v, \delta\Delta)$. Since the expansion is δ, we have $d_G(u, u') \leq \delta\Delta$, thus, all vertices of the shortest u-to-u' path are inside the ball $B(v, 2\delta\Delta)$. Therefore, $B(v, 2\delta\Delta)$ contains all vertices that f maps onto. It also contains the shortest path between every pair of vertices in the image of f. So, the restriction of f gives is an embedding of X into $G[B(v, \delta\Delta)]$ of distortion δ. ☐

Remark 2.5. *In light of Lemma 2.4, we assume that G has diameter at most $4\delta\Delta$. Our final algorithm tries all possibilities for B at the price of a factor of m in the running time.*

3 Overview of the Approach

Suppose we know there is an embedding f of X into G of distortion at most δ. To find f naively we could guess all possible embeddings, which can be viewed as having a list for each vertex in G of the possible vertices from X that can map onto it, where each guess of f consists of choosing one item from each of these lists. One issue with this approach is that selecting an item from a list tells you no information about how the other members of X should map into G, resulting in blindly trying all possible combinations of items from the lists (like putting a jigsaw puzzle together in the dark). Therefore, for each vertex v in G rather than only guessing which point in X maps onto v, we guess an approximate view of what the image of all points in X look like when sitting at the vertex v. As we are concerned with relative approximations, points whose image is far from v, do not need to be "seen" as clearly. Specifically, if a point maps within a ball centered at v with radius $O(\delta^{s+1})$, then we need to know its location up to a factor of $O(\delta^s)$. This motivates the definition of *focused views* from Section 4.

So now rather than having a list of possible points at each vertex, we have a list of possible focused views. This is potentially problematic as previously the list sizes where only $|X|$, but as focused views are more expressive, the sizes of these lists have grown significantly. Fortunately, one can avoid enumerating all focused views and hence keep manageable list sizes, and this is the focus of Section 5. Roughly speaking, this reduction is achieved by finding an approximate image of a net of X rather than all of X, where each point not in the net maps to the same region as its nearest neighbor in the net. This is the most technically challenging portion of the paper, and while this section and the ideas it contains are crucial for our approach, this section can be viewed as giving a black box list size reduction.

At this point we know focused views result in larger list sizes, so what is their benefit? If v is a cut vertex in G, then a focused view not only tells us about the local neighborhood of v, but also for points in X whose image is further away, it tells us which component of $G \setminus v$ they map to. That is, a cut in G induces a "cut" in X, implying that dynamic programming may be possible. The remaining issue is that in order to extract f we need a focused view at each vertex in G, and while in a tree every (non-leaf) vertex induces a cut, this is not the case for more general graphs. This is where tree decomposition comes into play, as each bag in a tree decomposition induces a cut. Hence instead of guessing focused views "centered" around each vertex in G, we guess focused views "centered" around the collection of vertices in each bag in the decomposition.

Many elements of this approach appeared in [NR15], though in a much rougher form, when the authors considered the much simpler case of embedding into a subset of the real line. In particular, handling the much more general class of graphs considered in this paper not only required a number of new elements and insights[3], but also would not have been possible without significant simplification, restructuring, and a better overall understanding of the approach, as described above. This better understanding also gives a strict improvement over the previous paper even for the case of the real line, as now we have a $1+\varepsilon$ rather than a (large) constant approximation.

Naturally the above is only a rough sketch, and the details are far more involved. Therefore, in the different parts of the argument below we provide further intuition in line about the specific details of how the above story is put into action.

4 Focused Views

In this section we formally define focused views and related terms, and prove some basic facts about them. As discussed in Section 3, a focused view of a subset $U \subseteq V_G$, assigns every $x \in X$ to a corresponding approximate location in V_G. The accuracy of this approximate location varies, being highly accurate near U, and less accurate as one moves away from U. Naturally, for any embedding f there is at least one corresponding focused view, though conversely a focused view may not comply with any feasible f.

Let $\rho \in \mathbb{R}$ and $\rho \geq \max(1/(\delta-1), 4)$. Also, let $U \subseteq V_G$, and let $S = \lceil \log_\delta \Delta \rceil + 1$. A collection of maps $F = \{f_0, f_1, \ldots, f_{S-1}\}$, where for each $0 \leq s < S$, $f_s : X \to 2^{V_G}$ with the following properties is called a ***focused view*** at U. For each $0 \leq s < S$, and each $x \in X$, $f_s(x)$ is either

(1) a subset of V_G of diameter less than δ^s that intersects $B(U, \rho\delta^{s+1})$, or

(2) it is $V_G \backslash B(U, \rho\delta^{s+1})$.

U is called the ***center*** of F, and ρ is the ***span*** of the view. For each $0 \leq s < S$, we call f_s the ***scale s view*** of F. A point x is ***visible*** at scale s if condition (1) holds for it, and ***invisible*** at scale s otherwise. Each scale s view provides an estimate for the image of x. Intuitively, these estimates are more accurate when s is smaller if the image of x is sufficiently close to the center.

[3] This includes netting in G in addition to X, having a parameter allowing us to fine tune the radii of views to match the desired accuracy, generalizing the notion of the center of a view both with respect to the number of components it induces and its size, and more. In general, these new pieces are too technical in nature to be understood in the context of the above high level overview, though they will become clear later in the paper.

In particular, if x is visible at scale zero then $f_0(x)$ is a point. A possibly more accurate estimation for the image of x can be provided by taking into account estimates of all scales, we denote this estimation by $F(x)$, defined below.

$$F(x) = \bigcap_{0 \le s < S} f_s(x).$$

We say F is ***a focused view of*** an embedding f, if $f(x) \in F(x)$ (equivalently, for any $x \in X$, and any $0 \le s < S$, $f(x) \in f_s(x)$). Conversely, f is called an ***extension*** F.

An embedding is ***feasible*** if it is non-contracting and its expansion is at most δ. A focused view is ***feasible*** if it can be extended to a feasible embedding. Ideally, we would like to work with feasible focused views. However, verifying if a focused view is feasible locally seems to be impossible. Instead, we define the weaker notion of plausibility. Intuitively, a focused view, F, is plausible if one cannot conclude it is not feasible by locally examining it. Formally, F is ***plausible*** if for all $x, y \in X$, there are $u \in F(x)$ and $v \in F(y)$ such that:

$$1 \le d_G(v, u)/d_X(x, y) \le \delta.$$

Note that the plausibility of F in particular implies that for all $x \in X$, $F(x) \ne \emptyset$. In turn, it is implied that every $x \in X$ is visible at scale $S-1$, because $V_G \backslash B(U, \rho\delta^S) = \emptyset$ by Lemma 2.4 as $\rho \ge 4$. Furthermore, we now show that a sufficiently large choice of ρ guarantees that once a point is visible at some some s, it remains visible for all larger scales.

Lemma 4.1. *Let F be a plausible focused view at U of span $\rho \ge 1/(\delta - 1)$. For any $x \in X$, and any $0 \le s \le s' < S$, if x is visible under F at scale s then it is also visible at scale s'.*

Proof: We show that for any $0 < s < S$, if x is visible at scale $s-1$ then it is visible at scale s. The statement of the lemma will be implied by induction. The vertex subset $f_{s-1}(x)$ has diameter δ^{s-1}, and it intersects $B(U, \rho\delta^s)$. Therefore, it is completely inside $B(U, \rho\delta^s + \delta^{s-1})$. Since $\rho \ge 1/(\delta - 1)$, we have $\rho\delta^s + \delta^{s-1} \le \rho\delta^{s+1}$, so, $f_{s-1}(x) \subseteq B(U, \rho\delta^{s+1})$, or equivalently $f_{s-1}(x)$ does not intersect $V_G \backslash B(U, \rho\delta^{s+1})$. As F is plausible, $F(x) \ne \emptyset$, and hence x is visible at scale s. □

Plausibility, of a focused map can be easily checked in polynomial time by computing $F(x)$ for all $x \in X$, and testing the plausibility condition for each pair of $x, y \in X$, by considering every choice of $u \in F(x)$ and $v \in F(y)$. Consequently, we have the following lemma.

Lemma 4.2. *Let F be focused view at $U \subseteq V_G$. There is a $n^{O(1)} \cdot (\delta\Delta)^{O(\lambda)}$ time algorithm to decide whether F is plausible or not (provided that $|V_G| = (\delta\Delta)^{O(\lambda)}$).*

4.1 Aggregating focused views

We wish to aggregate several focused views to recover a nearly optimal embedding. Here we show this recovery is possible, provided a set of focused views which yields a sufficient amount of non-contradictory information.

Let $\mathcal{F}$ be a collection of plausible focused views. We say that $\mathcal{F}$ is a ***complete set of focused views*** if for each $x \in X$, we have:

(1) $\bigcap_{F \in \mathcal{F}} F(x) \ne \emptyset$, and

(2) there is an $F = \{f_0, f_1, \ldots, f_{S-1}\} \in \mathcal{F}$ at a subset $U \in V_G$, such that $f_0(x) \in U$.

A complete set of focused views imply a unique embedding of X into G. We define $h_{\mathcal{F}}$ to be this unique embedding, the aggregation of all scale zero views.

$$h_{\mathcal{F}} = \bigcup_{\{f_0,\ldots,f_{S-1}\}\in\mathcal{F}} f_0. \tag{1}$$

We call $h_{\mathcal{F}}$ the ***global view*** of $\mathcal{F}$, and we show that $h_{\mathcal{F}}$ is a nearly optimal embedding. First, we show that $h_{\mathcal{F}}$ is an embedding.

Lemma 4.3. *Let $\mathcal{F}$ be a complete set of focused views, and let $h_{\mathcal{F}}$ be its global view. We have that $h_{\mathcal{F}}$ is an embedding of X into V_G.*

Proof: Property (2) ensures that $h_{\mathcal{F}}$ acts on every point in X. Property (1) ensures that for each $x \in X$ and $F, F' \in \mathcal{F}$ if $x \in \mathrm{Dom}(f_0)$ and $x \in \mathrm{Dom}(f'_0)$ then $f_0(x) = f'_0(x)$, thus, $h_{\mathcal{F}}(x)$ is a well-defined function.

It remains to show that $h_{\mathcal{F}}$ is one-to-one: for any $x, y \in X$, $x \neq y$ implies $h_{\mathcal{F}}(x) \neq h_{\mathcal{F}}(y)$. Since $\mathcal{F}$ is a complete set of focused views it contains a focused view $F = \{f_0, f_1, \ldots, f_{S-1}\} \in \mathcal{F}$ at $U \in V_G$, such that $f_0(x) \in U$. Note that $f_0(x)$ is a point, so $h_{\mathcal{F}}(x) = f_0(x)$. If y is visible at scale zero, then $f_0(y) \neq f_0(x)$ because of plausibility of F, thus, $h_{\mathcal{F}}(x) \neq h_{\mathcal{F}}(y)$. Otherwise, $f_0(y) = V_G \backslash B(U, \rho\delta)$. By Property (2) of a complete set of focused views we have $h_{\mathcal{F}}(y) \in f_0(y)$, in particular, $h_{\mathcal{F}}(y) \neq f_0(x) = h_{\mathcal{F}}(x)$. □

Next, we show that $h_{\mathcal{F}}$ has bounded distortion.

Lemma 4.4. *Let $\mathcal{F}$ be a complete set of focused views of span ρ, and let $h_{\mathcal{F}}$ be its global view. We have that $h_{\mathcal{F}}$ is an embedding with distortion at most $(\rho+1)/(\rho-1)\delta$.*

Proof: Let $x, y \in X$, and let $h = h_{\mathcal{F}}$. We show

$$\left(1 - \frac{1}{\rho}\right) \leq \frac{d_G(h(x), h(y))}{d_X(x, y)} \leq \left(1 + \frac{1}{\rho}\right) \cdot \delta.$$

Let $U \subseteq V_G$ be such that there is $F \in \mathcal{F}$ at U with $f_0(x) \in U$. Such U and F exists as $\mathcal{F}$ is a complete set of focused views. Thus, we have $h(x) = f_0(x)$. Let $0 \leq s < S$ be the smallest scale, in which y is visible. If $s = 0$, then $h(y) = f_0(y)$, and the plausibility of F implies

$$1 \leq \frac{d_G(h(x), h(y))}{d_X(x, y)} \leq \delta.$$

Otherwise, because F is plausible, there is a $v \in f_s(y)$ such that

$$1 \leq \frac{d_G(h(x), v)}{d_X(x, y)} \leq \delta. \tag{2}$$

Since $f_s(y)$ has diameter less than δ^s,

$$d_G(h(y), v) \leq \delta^s. \tag{3}$$

On the other hand, y is not visible at scale $s-1$, so, $h(y) \notin B(U, \rho\delta^s)$.

$$d_G(h(x), h(y)) \geq \rho\delta^s. \tag{4}$$

Combining (3) and (4), we obtain

$$d_G(h(y), v) \le \frac{1}{\rho} \cdot d_G(h(x), h(y)).$$

Using the triangle inequality, we obtain

$$d_G(h(x), v) \le d_G(h(x), h(y)) + d_G(h(y), v) \le \left(1 + \frac{1}{\rho}\right) \cdot d_G(h(x), h(y)), \tag{5}$$

and

$$d_G(h(x), v) \ge d_G(h(x), h(y)) - d_G(h(y), v) \ge \left(1 - \frac{1}{\rho}\right) \cdot d_G(h(x), h(y)). \tag{6}$$

We combine (2) and (5) to bound the expansion.

$$\frac{d_G(h(x), h(y))}{d_X(x, y)} \le \left(1 + \frac{1}{\rho}\right) \cdot \frac{d_G(h(x), v)}{d_X(x, y)} \le \left(1 + \frac{1}{\rho}\right) \cdot \delta.$$

Similarly, we combine (2) and (6) to bound the contraction.

$$\frac{d_G(h(x), h(y))}{d_X(x, y)} \ge \left(1 - \frac{1}{\rho}\right) \cdot \frac{d_G(h(x), v)}{d_X(x, y)} \ge \left(1 - \frac{1}{\rho}\right).$$

□

4.2 Pairwise consistency

A nearly optimal embedding can be deduced from a complete set of focused views as promised by Lemma 4.4, therefore, all we need is such a complete set. To compute this complete set our algorithm proceeds in three steps: (i) choose appropriate set of centers for our focused views, (ii) find a relatively small list of focused views at each chosen center, and (iii) extract a complete set by picking one element from each list. Step (iii) is the most challenging step, and to accomplish this step we will need a stronger notion of consistency between the focused view than what is provided by property (1) of a complete set of views, and this notion of consistency is described below. Naturally, ensuring this stronger notion of consistency, requires an appropriate choice of the centers and list in steps (i) and (ii), and this is discussed in the next section.

We say that F and F' at U and U', respectively, are ***consistent***, if for each $x \in X$, and each $0 \le s < S$ the following two properties hold.

(1) If x is visible under F and not visible under F' at scale s then $f_s(x)$ is disjoint from $B(U', \rho\delta^{s+1})$. Similarly, if x is visible in F' and not visible in F at scale s then $f'_s(x)$ is disjoint from $B(U, \rho\delta^{s+1})$.

(2) if x is visible at scale s under both F and F' then $f_s(x) = f'_s(x)$.

Note that (1) and (2) are much stronger than what we needed for a pair of focused views in a complete set. For example, (2) requires that the estimates of F and F' of x at a given scale s be *exactly* the same if x is visible at scale s under both of them. On the other hand, F and F' can be in a complete set if their estimations for x are merely not completely disjoint.

Consistency, of a pair of focused views can be easily checked in polynomial time by checking conditions (1) and (2) for each $x \in X$ and $0 \le s < S$ in $|V_G|^{O(1)}$ time, leading to the following lemma.

Lemma 4.5. *The consistency of two focused views can be tested in $n^{O(1)} \cdot (\delta\Delta)^{O(\lambda)}$ time (provided that $|V_G| = (\delta\Delta)^{O(\lambda)}$).*

The following lemma shows that the pairwise consistency of the elements of a set of focused views into a cover of V_G implies the completeness of the set.

Lemma 4.6. *Let $\mathcal{U}$ be a family of subsets of V_G that covers V_G. Let $\mathcal{F}$ be a collection of plausible focused views, one at each member of $\mathcal{U}$. If every pair of focused view in $\mathcal{F}$ are consistent, then $\mathcal{F}$ is a complete set of focused views.*

Proof: First, we show that for any $0 \leq s < S$ and $x \in X$ there is an $F \in \mathcal{F}$ under which x is visible at scale s. This statement, in particular, implies that x is visible at scale zero under some focused view, the second property of a complete set of focused views. Since F is plausible, every $x \in X$ is visible at scale $S-1$ under F. So, let $0 \leq s < S-1$. By the induction hypothesis, there is $F' = \{f'_0, f'_1, \ldots, f'_{S-1}\}$, under which x is visible at scale $s+1$. Let $U \in \mathcal{U}$ be any a subset of V_G that intersects $f'_{s+1}(x)$, and let $F = \{f_0, f_1, \ldots, f_{S-1}\} \in \mathcal{F}$ be the focused view at U. Such a U exists because $\mathcal{U}$ covers V_G. The consistency of F and F' implies that $f_{s+1}(x) = f'_{s+1}(x)$. The plausibility of F implies that $f_s(x)$ intersects $f_{s+1}(x)$. But, $f_{s+1}(x)$ is completely inside $B(U, \rho\delta^{s+1})$, as it intersects U, and its diameter is less than δ^{s+1}. Therefore, $f_s(x)$ must intersect $B(U, \rho\delta^{s+1})$, that is x is visible under F at scale s.

To show the first property of a complete set of focused views, let $x \in X$, and let $F = \{f_0, f_1, \ldots, f_{S-1}\} \in \mathcal{F}$ be a focused view at U under which x is visible at scale zero. By Lemma 4.1, x is visible at all scales under F. For any other focused view $F' = \{f'_0, f'_1, \ldots, f'_{S-1}\}$, and any scale $0 \leq s < S$, we have that $f'_s(x) \supseteq f_s(x)$, since if x is visible at scale s under F', $f'_s(x) = f_s(x)$ (the second property of consistency), and if x is not visible then $f'_s(x) \supseteq f_s(x)$ (the first property of consistency). Hence, all estimations for the image of x contain $f_0(x)$, so, their intersection is not empty. □

5 Enumerating focused views

Given an appropriate set of centers, we wish to compute focused views at these centers that can be aggregated into nearly optimal embeddings. In this section, we show how to compute a relatively short list of focused views at each center that contains a focused view of *every* feasible embedding. In particular, for any feasible f, each of our lists contains a focused view of f at its corresponding center. We build our focused views so that views of a single f at different centers are consistent, ensuring these lists contain a complete set of focused views, which our algorithm in the next section describes how to extract.

Lemma 5.1. *For any $U_1, \ldots, U_k \subseteq V_G$, and $\rho \geq 1$, there is an algorithm to compute a list of $n^{|U_i|} \cdot (\delta\Delta)^{|U_i| \cdot (1+\log_\delta \rho) \cdot \lambda \cdot \left(O(\rho\delta^2)\right)^\lambda}$ focused views of span ρ at each U_i that contains a focused view of every feasible embedding in $k \cdot n^{\alpha+1} \cdot (\delta\Delta)^{\alpha \cdot (1+\log_\delta \rho) \cdot \lambda \cdot \left(O(\rho\delta^2)\right)^\lambda}$ time, where $\alpha = \max_{1 \leq i \leq k} |U_i|$. Moreover, any pair of focused views of a feasible f computed by this algorithm are consistent.*

To build the candidate lists of focused views around each U_i described in the above lemma, we first build a relatively small list of choices for f_s for each s, and then combine these lists. Recall that f_s provides a δ^s accurate image of every $x \in X$ at distance $O(\delta^{s+1})$ of U_i. We show that f_s can be deduced if a $\delta^s/3$ accurate image of a $\delta^{s-1}/3$-net, $X_{\geq s-1}$, in X is provided around U_i. We then determine $f_s(x)$ for each x by looking at the image of its nearest neighbor z in $X_{\geq s-1}$. To limit the possibilities for the image of any point, we also use a fixed set of bins of diameter $\delta^s/3$ defined over the vertices of G. The list of possible choices for f_s is then constructed by enumerating all possible mappings onto U_i, and for each mapping, all possible assignments of the net points to the set of bins which intersect the visible range of U_i.

5.1 Nets and Bins

Nets on X. Let $(x_1, x_2, \ldots, x_n)$ be the Gonzalez permutation of X computed as follows. The point x_1 is an arbitrary point in X. For every $2 \le i \le n$, the point $x_i \in X\backslash\{x_1, \ldots, x_{i-1}\}$ is the farthest point from the set $\{x_1, \ldots, x_{i-1}\}$. For each $0 \le s < S$, the set $\boldsymbol{X_{\ge s}}$ is composed of the points in the maximal prefix of $(x_1, x_2, \ldots, x_n)$, in which the mutual distances of the points are at least $\delta^s/3$. Note that for any $s \le 0$, $X_{\ge s} = X$. The scale of a point $x \in X$, is the largest s such that $x \in X_{\ge s}$. For any point $x \in X$, and any $0 \le s < S$, we denote by $\boldsymbol{nn_s(x)}$ the nearest point of $X_{\ge s}$ to x. For example, if $x \in X_{\ge s}$ then $nn_s(x) = x$.

For the remainder of the paper we assume that we have precomputed the sets $X_s = X_{\ge s}\backslash X_{\ge s+1}$, for all $s \in \{0, 1, \ldots, S-1\}$. Additionally, we also precompute for each $x \in X$ and for each $0 \le s < S$, the nearest neighbor of x in $X_{\ge s}$. This can all be done in $(Sn)^{O(1)}$ time. We find the following implication of Lemma 2.2 helpful in the paper.

Lemma 5.2. *Any ball B of radius R in X contains at most $\left(\frac{6R}{\delta^{s-1}}\right)^\lambda$ points from $X_{\ge s}$.*

Proof: Let $f : X \to V$ be a non-contracting distortion δ embedding. The image $f(B)$ will be contained in a ball B' of radius δR of G. In particular, we have

$$|f(X_{\ge s}) \cap B'| \ge |X_{\ge s} \cap B|.$$

Additionally, since f is non-contracting the distance between any pair of points in $f(X_{\ge s})$ is at least $\delta^s/3$. Lemma 2.2 then implies that

$$|f(X_{\ge s}) \cap B'| \le \left(\frac{6\delta R}{\delta^s}\right)^\lambda = \left(\frac{6R}{\delta^{s-1}}\right)^\lambda.$$

□

Bins in G. We use predefined bins to estimate the images of the net points. A ***family of bins*** $\{\sqcup_0, \sqcup_1, \ldots, \sqcup_{S+1}\}$ of ***density*** α and ***accuracy*** β in G is a family of partitions of V_G with the following properties:

(1) For each $0 \le s < S$ and any bin $b \in \sqcup_s$, $G[b]$ is a subgraph with diameter smaller than δ^s/β.

(2) For each $0 \le s < S$ and any $D \ge 1$, any diameter D subgraph of G intersects at most $(O(\beta D \delta^{-s}))^\alpha$ bins of scale s.

Each element in $\sqcup_s$ is called a ***bin*** of scale s. In particular, a bin of scale zero contains exactly one vertex.

Lemma 5.3. *For any $\beta \ge 1$, there is a family of bins of density λ and accuracy β in G that can be computed in $m^{O(1)}$ time.*

Proof: For each $0 \leq s < S$, let $V_{\geq s}$ be composed of the elements of the longest prefix of the Gonzalez permutation of V_G, in which the mutual distances between vertices is at most $\delta^s/(2\beta)$. For each $v \in V_{\geq s}$ let the scale s bin b_v centered at v be the set of all vertices in V_G that are closer to v than any other vertex in $V_{\geq s}$ (break the ties arbitrarily). By the definition of $V_{\geq s}$, for each $u \in b_v$ we have $d_G(u, v) < \delta^s/(2\beta)$, thus, the diameter of b_v is at most δ^s/β. Let $\sqcup_s$ be the set of all scale s bins. The argument above ensures property (1). It remains to prove property (2) for $\sqcup_s$.

Let $G' = (V'_G, E'_G)$ be a subgraph of diameter D. Let $v' \in V'_G$, and consider the ball $B = B(v', D + \delta^s/\beta)$. Any scale s bin that intersects V'_G is completely contained in B. We bound the number of scale s bins in B. Let U be the set of centers of such bins. As $U \subseteq V_{\geq s}$, the mutual distance between points of U is at least $\delta^s/(2\beta)$. Therefore, Lemma 2.2 implies:

$$|U| \leq \left(\frac{2D + 2\delta^s/\beta}{\delta^s/(2\beta)}\right)^{\lambda} = \left(O(\beta D \delta^{-s})\right)^{\lambda}.$$

The Gonzalez permutation and the set of bins in every scale can be computed in $m^{O(1)}$. ☐

This lemma implies that a family of bins of density λ and accuracy 3 exists in G. For the rest of this paper, let $\{\sqcup_0, \sqcup_1, \ldots, \sqcup_{S+1}\}$ be such a family. That is the diameter of a scale s bin is smaller than $\delta^s/3$.

5.2 Net views vs. focused views

A collection of partial maps $N = \{\eta_0, \eta_1, \ldots, \eta_{S-1}\}$, where for each $0 \leq s < S$, $\eta_s : X_{\geq s-1} \nrightarrow \sqcup_s$, is called a ***net view***. For any $0 \leq s < S$, η_s is called the scale s map of the net view. Note the domain of η_s is $X_{\geq s-1}$, which is a $\delta^{s-1}/3$ net.

Let f be an embedding of X into G, and for each $x \in X$, let $b_s(x) \in \sqcup_s$ denote the unique bin of scale s that contains $f(x)$. For any $r > 1$, and $U \subseteq V_G$ the ***span*** r ***net restriction*** of f into U, is a net view $N = \{\eta_0, \eta_1, \ldots, \eta_{S-1}\}$ defined as follows. For each $0 \leq s < S$, $x \in X_{\geq s-1}$,

(1) if $b_s(x)$ intersects $B(U, r\delta^{s+1})$ then $\eta_s(x) = b_s(x)$,

(2) otherwise, η_s does not act on x.

Note that a net restriction looks like a focused view, except η_s is only defined on the net $X_{\geq s-1}$, and its images has slightly smaller diameters. We now define how to extend it to a focused view by mapping each $x \in X$ roughly to the same set mapped onto by $z = nn_{s-1}(x)$. Specifically, if z is "visible" we map x to the same bin as z, enlarged by an additive factor of $\delta d_X(x, z)$ to account for the error in approximating the location of the image of x by the image of z.

For a span r net restriction, $N = \{\eta_0, \eta_1, \ldots, \eta_{S-1}\}$, of an embedding into U, we define its ***induced*** focused view $F = \{f_0, f_1, \ldots, f_{S-1}\}$ as follows. For each $x \in X$, and $0 \leq s < S$,

(1) if η_s acts on $nn_{s-1}(x)$, and $B = B(\eta_s(nn_{s-1}(x)), \delta \cdot d_X(x, nn_{s-1}(x)))$ intersects $B(U, (r-1)\delta^{s+1})$ then $f_s(x) = B$,

(2) otherwise, $f_s(x) = V_G \backslash B(U, (r-1)\delta^{s+1})$.

Note that $nn_{-1}(x) = x$, therefore, $f_0(x) = \eta_0(x)$, which is a vertex, if $\eta_0(x)$ is defined, otherwise $f_0(x) = V_G \backslash B(U, (r-1)\delta)$.

Lemma 5.4. *Let f be a feasible embedding, $U \subseteq V_G$, N be the span r net restriction of f into U, and F be the focused view induced by N. We have that F is a span $r-1$ focused view of f. Moreover, F can be computed from N in $n(\delta\Delta)^{O(1)}$ time.*

Proof: It is immediate from the definition of induced focused views that $f_s(x)$ is either a subset of diameter at most δ^s that intersects $B(U,(r-1)\delta^{s+1})$ or it is $V_G \backslash B(U,(r-1)\delta^{s+1})$, for every $x \in X$, and $0 \leq s < S$. We show that $f(x) \in f_s(x)$. To this end, we consider two cases. Let $z = nn_{s-1}(x)$ in the rest of the proof.

First, suppose η_s acts on z, and let $B_x = B(\eta_s(z), \delta \cdot d_X(x,z))$. We have $f(x) \in B_x$ because $f(z) \in \eta_s(z)$, and f has expansion at most δ. If B_x intersects $B(U,(r-1)\delta^{s+1})$ then $f_s(x) = B_x \ni f(x)$, otherwise, $f_s(x) = V_G \backslash B(U,(r-1)\delta^{s+1}) \supseteq B_x \ni f(x)$.

Second, suppose that η_s does not act on z, so, $f_s(x) = V_G \backslash B(U,(r-1)\delta^{s+1})$. Additionally, $b_s(z)$, the scale s bin that contains $f(z)$, does not intersect $B(U, r\delta^{s+1})$, that is $d_G(f(z),U) \geq r\delta^{s+1}$. Consequently,

$$d_G(f(x),U) \geq d_G(f(z),U) - d_G(f(x),f(z)) \geq r\delta^{s+1} - \delta \cdot \delta^{s-1}/3 > (r-1)\delta^{s+1},$$

thus, $f(x) \in f_s(x)$.

For any $x \in X$, $z = nn_{s-1}(x)$ was precomputed, and hence the bin $\eta_s(z)$ can be found in $O(1)$ time. Using a BFS from the bin $\eta_{s-1}(z)$ we can compute $f_s(x)$ in $O(|f_s(x)|) = O(\delta^s)$ time. Therefore, the total running time to compute $f_s(x)$ for all $0 \leq s < S$ is $O(\delta^S) = (\delta\Delta)^{O(1)}$. The running time of the lemma follows because we there are n choices for x. $\square$

5.3 Enumerating net views

For a $U \in V_G$, we wish to list a set of net views that contain the span r net restriction of every feasible embedding into U. The following lemma shows how to find this list and bound its size simultaneously.

Lemma 5.5. *For any $U \subseteq V_G$, and $r \geq 1$, there is an algorithm to compute a list of $n^{|U|} \cdot (\delta\Delta)^{|U|\cdot(1+\log_\delta r)\cdot\lambda\cdot\left(O(r\delta^2)\right)^\lambda}$ net views that contains the span r net restriction of every feasible embedding into U, in the same asymptotic running time.*

Proof: We describe an algorithm to find a list of net views that include the span r net restriction of every feasible map f into U. Our algorithm, first guesses the preimage X_U of U under f, and the restriction of f to X_U, $g : X_U \to U$. There are $n^{|U|}$ choices g.

Next, for each $0 \leq s < S$, our algorithm lists a set of possibilities for $\eta_s : X_{\geq s-1} \to \sqcup_s$. Note that, the partial map η_s only maps into scale s bins that intersect a $B(u, r\delta^{s+1})$ for a $u \in U$. For each $u \in U$, let $\eta_s[u]$ be the restriction of η_s in its range to the subset of $\sqcup_s$ that intersect $B(u, r\delta^{s+1})$. This subset has size $(O(r\delta))^\lambda$ because $\sqcup_s$ belongs to a family of bins of density λ with a constant accuracy. On the other hand, $\eta_s[u]$ can only act on the points of $X_{\geq s-1}$ that are at distance at most $r\delta^{s+1} + \delta^s/3 \leq (r+1)\delta^{s+1}$ of $g^{-1}(u)$, as otherwise any extension of N must be contracting. There are $\left(O(r\delta^3)\right)^\lambda$ such points by Lemma 5.2. Since f is non-contracting, Lemma 2.2 implies that at most $(O(\delta))^\lambda$ points of $X_{\geq s-1}$, have their images in a bin of scale s. Therefore, provided $g^{-1}(u)$, there are

$$\left[\left(O(r\delta^3)\right)^\lambda\right]^{(O(\delta))^\lambda} = (r\delta)^{\lambda\cdot(O(\delta))^\lambda}$$

choices for the preimage of each bin that intersects $B(u, r\delta^{s+1})$. Since, there are $(O(r\delta))^\lambda$ number of such bins, the total number of choices for η_s is

$$\left[(r\delta)^{\lambda\cdot(O(\delta))^\lambda}\right]^{(O(r\delta))^\lambda} = (r\delta)^{\lambda\cdot\left(O(r\delta^2)\right)^\lambda}$$

Our algorithm lists a set of possibilities for η_s by combining its guesses for $\eta_s[u]$ for all $u \in U$. Then, it makes the list for N by combining the lists for η_s's for all $0 \leq s < S$. Thus, provided g, there are

$$(r\delta)^{\lambda \cdot \left(O(r\delta^2)\right)^{\lambda} \cdot |U| \cdot S}$$

number of choices for N. Since there are $n^{|U|}$ choices for g, our algorithm outputs a list of size

$$n^{|U|} \cdot (r\delta)^{\lambda \cdot \left(O(r\delta^2)\right)^{\lambda} \cdot |U| \cdot S} = n^{|U|} \cdot (\delta\Delta)^{|U| \cdot (1+\log_\delta r) \cdot \lambda \cdot \left(O(r\delta^2)\right)^{\lambda}}$$

that contains the net restriction of all feasible embeddings. The last equality holds because $S = \lceil \log_\delta \Delta \rceil + 1$. $\square$

Now, we are ready to prove the main lemma of this section.

Proof (of Lemma 5.1): By Lemma 5.4, to obtain focused views of span ρ, we need to consider span $\rho+1$ net restrictions of feasible embeddings. We use Lemma 5.5 to compute a list of net views into each U_i that contains a span $\rho + 1$ net restriction of every feasible embedding. According to the same lemma, the size of this list is

$$n^{|U_i|} \cdot (\delta\Delta)^{|U_i| \cdot (1+\log_\delta(\rho+1)) \cdot \lambda \cdot \left(O((\rho+1)\delta^2)\right)^{\lambda}} = n^{|U_i|} \cdot (\delta\Delta)^{|U_i| \cdot (1+\log_\delta \rho) \cdot \lambda \cdot \left(O(\rho\delta^2)\right)^{\lambda}}$$

and it can be computed in the same asymptotic running time. Each of these net views N induces a unique focused view F of f by Lemma 5.4. Since for any f, U, and span there is a unique net restriction, which in turn induces a unique focused view, any two focused views of the same f computed by our algorithm must be consistent. The total running time to compute the list for each U_i is bounded by

$$n^{\alpha+1} \cdot (\delta\Delta)^{\alpha \cdot (1+\log_\delta \rho) \cdot \lambda \cdot \left(O(\rho\delta^2)\right)^{\lambda}},$$

as $\alpha = \max_{0 \leq i \leq k} |U_i|$, and we need $n(\delta\Delta)^{O(1)}$ time to extend each of the net views in the list to a focused view. The final running time is as stated in the lemma because we are computing k lists. $\square$

6 Merging focused views on a tree decomposition

Let's recap. In Section 4, we outlined the goal of finding a complete set of focused views. As previously discussed, to achieve this we choose a collection of centers, define lists of possible focused views over these centers, and then extract a complete set from these lists. For each chosen center, in Section 5 it was shown how to construct such a list whose size is significantly smaller than what would result from brute force enumeration of focused views.

What remains is to extract a complete set from these lists. As completeness is a property of an entire collection of focused views, naively we must try all possibilities of selecting one focused view from each center's list, and check whether each resulting set is complete. Fortunately, it just so happens that in Lemma 5.1 we proved our lists have the stronger property of containing a pairwise consistent set, i.e. a set such that satisfying a local condition between each pair, implies completeness for the set. While pairwise consistency is a strong property, it is still unclear how to extract consistent focused views from the lists. This is where tree decomposition comes into play.

Let $\mathcal{T} = (V_{\mathcal{T}}, E_{\mathcal{T}})$ be a tree decomposition of G. We choose the centers for our focused views to be the sets in $V_{\mathcal{T}}$. The crucial property of the tree decomposition is that to check for consistency between all pairs of a collection of focused views, it suffices to check for consistency between the pairs determined by $E_\mathcal{T}$, and this property allows us to find a consistent set from the lists using dynamic programming over $\mathcal{T}$. Note that this is the only place where we use the tree decomposition, and moreover Lemma 5.1 relates the running time to the size of the bags in $V_\mathcal{T}$. Lemma 6.3 proves the above described crucial property, and to this end we have the following helper lemmas.

Lemma 6.1. *Let $\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T})$ be a tree decomposition of G, and let $\mathcal{B} \in V_\mathcal{T}$. Let $\mathcal{T}_1, \ldots, \mathcal{T}_t$ be the connected components of $\mathcal{T}\backslash\mathcal{B}$, and let C be a connected component of $G\backslash\mathcal{B}$. We have that C intersects bags of exactly one of the $\mathcal{T}_i$'s.*

Proof: Suppose, to derive a contradiction, there are $u, u' \in C$ such that $u \in \mathcal{B} \in \mathcal{T}_i$, and $u' \in \mathcal{B}' \in \mathcal{T}_j$, for $1 \leq i < j \leq t$. Let γ be a u-to-u' path in C, and note that $\gamma \cap \mathcal{B} = \emptyset$. Therefore, for each $v \in \gamma$ there is a unique $1 \leq i \leq t$ such that $v \in \mathcal{T}_i$. Since, u and u' belong to bags of different connected component of $\mathcal{T}\backslash\mathcal{B}$, there is an edge $(v, v') \in \gamma$, such that $v \in \mathcal{T}_l$ and $v' \in \mathcal{T}_k$ for some $1 \leq l < k \leq t$. Since $\mathcal{T}$ is a tree decomposition it must have a bag $\mathcal{B}'$ that contains both v and v'. Moreover, since v and v' belong to different connected components of $\mathcal{T}\backslash\mathcal{B}$, and since $v, v' \notin \mathcal{B}$, it follows that $\mathcal{B}'$ cannot belong to any $\mathcal{T}_i$. □

Lemma 6.2. *Let $(\mathcal{B}, \mathcal{B}') \in E_\mathcal{T}$, and let F and F' be consistent focused views at $\mathcal{B}$ and $\mathcal{B}'$, respectively. Let $\mathcal{T}_{\mathcal{B}'}$ be the connected component of $\mathcal{T}\backslash\mathcal{B}$ that contains $\mathcal{B}'$, and let U' be the set of all vertices in the bags of $\mathcal{T}_{\mathcal{B}'}$. For any $x \in X$, and $0 \leq s < S$, if x is visible under F' and not under F at scale s then $f'_s(x) \subseteq U'$, and $f'_s(x) \cap \mathcal{B} = \emptyset$.*

Proof: The consistency of F and F' implies that $f'_s(x)$ does not intersect $B(\mathcal{B}, \rho\delta^{s+1})$, and, in particular, $f'_s(x) \cap \mathcal{B} = \emptyset$. Suppose, to derive a contradiction, that there is $u \in f'_s(x)$, $u \notin U'$.

If there exists $v \in f'_s(x) \cap U'$, by Lemma 6.1 any u-to-v path must intersect $\mathcal{B}$. As $f'_s(x)$ is connected, there is a u-to-v path contained in $f'_s(x)$, but we know that $f'_s(x) \cap \mathcal{B} = \emptyset$, thus, $f'_s(x) \cap U' = \emptyset$. Therefore, any path from $f'_s(x)$ to $\mathcal{B}'$ in G intersects $\mathcal{B}$, in particular, $d_G(f'_s(x), \mathcal{B}) \leq d_G(f'_s(x), \mathcal{B}') \leq \rho\delta^{s+1}$. Thus, the consistency of F and F' implies that $f'_s(x)$ intersects $B(\mathcal{B}, \rho\delta^{s+1})$, which is contradictory. □

Lemma 6.3. *Let $(\mathcal{B}, \mathcal{B}'), (\mathcal{B}', \mathcal{B}'') \in E_\mathcal{T}$. Let F, F', and F'' be plausible focused views at $\mathcal{B}$, $\mathcal{B}'$ and $\mathcal{B}''$, respectively. If F and F' are consistent, and F' and F'' are consistent, then F and F'' are consistent.*

Proof: For any $x \in X$ and any $0 \leq s < S$, we show that the conditions of consistency for F and F'' hold. Let $\mathcal{T}_\mathcal{B}$ and $\mathcal{T}_{\mathcal{B}''}$ be the connected components of $\mathcal{T}\backslash\mathcal{B}'$ that contain $\mathcal{B}$ and $\mathcal{B}''$, respectively. Let U and U'' be the set of all vertices in the bags of $\mathcal{T}_\mathcal{B}$ and $\mathcal{T}_{\mathcal{B}''}$, respectively.

If x is invisible under both F and F'' at scale s then no condition is imposed by consistency. So, assume x is visible under at least one of them, and without loss of generality, suppose that is F. If x is visible at scale s under F' then the consistency of F with F' implies that $f_s(x) = f'_s(x)$, thus, the consistency of F' and F'', implies our desired condition for x and s.

So, suppose that x is visible under F but not under F' at scale s. There are two cases to consider, either x is visible at scale s under F'' or not.

First, suppose that x is visible under F''. By Lemma 6.2, since F and F' are consistent $f_s(x) \subseteq U$, and since F' and F'' are consistent $f''_s(x) \subseteq U''$. Let $a > s$ be the smallest scale at which x is visible under F'. Note that $f'_a(x)$ does not intersect $\mathcal{B}'$, as x is not visible at scale $a-1$ under F'. Also, by Lemma 4.1, x is visible at scale a, under both F and F''. Therefore, let $A = f_a(x) = f'_a(x) = f''_a(x)$. Since F and F'' are plausible, A intersects both $f_s(x)$ and $f''_s(x)$. But, this is not possible as $f_s(x)$ and $f''_s(x)$ are in different connected components of $G\backslash\mathcal{B}'$, and A does not intersect $\mathcal{B}'$, yet A is connected.

Second, suppose that x is not visible under F'' at scale s. Since x is visible under F and not under F' at scale s, Lemma 6.2 implies that $f_s(x) \subseteq U$. Also, as F and F' are consistent, $f_s(x)$ does not intersect $B(\mathcal{B}', \rho\delta^{s+1})$, that is $d_G(f_s(x), \mathcal{B}') > \rho\delta^{s+1}$. Additionally, $d_G(f_s(x), \mathcal{B}') \leq d_G(f_s(x), \mathcal{B}'')$ because $f_s(x)$ is in U, so any path from $f_s(x)$ to $\mathcal{B}''$ intersects $\mathcal{B}'$ by Lemma 6.1. Therefore, $d_G(f_s(x), \mathcal{B}'') > \rho\delta^{s+1}$, so $f_s(x)$ does not intersect $B(\mathcal{B}'', \rho\delta^{s+1})$ as required for consistency. $\square$

The following corollary that immediately follows from Lemma 6.3 and Lemma 4.6 facilitates the computation of a complete set of focused views from the lists of Lemma 5.1.

Corollary 6.4. *Let $\mathcal{T} = (V_{\mathcal{T}}, E_{\mathcal{T}})$ be a tree decomposition of G, and let $\mathcal{F}$ be a collection of focused views at bags of $\mathcal{T}$. If the focused views of $\mathcal{T}$ into any adjacent pair of bags are consistent then $\mathcal{F}$ is a complete set of focused views.*

6.1 The Algorithm, Proof of Theorem 1.1

We describe a dynamic programming procedure to extract a complete set of focused views at bags of $\mathcal{T}$ from the lists computed by Lemma 5.1. The key requirement of our dynamic programming is Corollary 6.4, that the consistency of focused views at adjacent bags implies completeness.

Let $\mathcal{R}$ be an arbitrary vertex of $\mathcal{T}$ and direct all edges of $\mathcal{T}$ towards $\mathcal{R}$. Let $\mathcal{B} \in \mathcal{T}$, and let $\mathcal{T}_{\mathcal{B}}$ denote the subtree of $\mathcal{T}$ with root $\mathcal{B}$. Let $\mathcal{F}_{\mathcal{B}}$ be a set of focused views at the bags of $\mathcal{T}_{\mathcal{B}}$, which contains exactly one view at each bag of $\mathcal{T}_{\mathcal{B}}$. We say that $\mathcal{F}_{\mathcal{B}}$ is consistent if the focused views at every pair of adjacent bags in $\mathcal{T}_{\mathcal{B}}$ are consistent. We say that a focused view F at $\mathcal{B}$ is an ***end view*** at $\mathcal{B}$ if there is a consistent set of focused views at $\mathcal{T}_{\mathcal{B}}$ that contains F. The goal of our dynamic programming procedure is to find an end view at $\mathcal{R}$.

Lemma 6.5. *Suppose, a tree decomposition $\mathcal{T}$ of G with treewidth ω' is provided. For any $\epsilon > 0$, and $\delta > 1$ (note both strict inequalities), there is a*

$$n^{O(\omega')} \cdot (\delta\Delta)^{\omega' \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda}$$

time algorithm to compute a $(1+\epsilon)\delta$ embedding of X into V_G if $\delta \geq \delta_{opt}$, where $\rho = 1/(\delta-1)+2/\epsilon+5$ (provided $|V_G| = (\delta\Delta)^{O(\lambda)}$). If $\delta < \delta_{opt}$, this algorithm either computes an embedding of distortion $(1+\epsilon)\delta$, or (correctly) decides that $\delta < \delta_{opt}$.

Proof: Our algorithm computes a list of focused views $L_{\mathcal{B}}$ at every $\mathcal{B} \in V_{\mathcal{T}}$ by Lemma 5.1, and trims implausible views in $L_{\mathcal{B}}$ using the algorithm of Lemma 4.2, in the following total running time.

$$\left(n^{O(\omega')} \cdot (\delta\Delta)^{\omega' \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda}\right) \cdot \left(n^{O(1)} \cdot (\delta\Delta)^{O(\lambda)}\right) = n^{O(\omega')} \cdot (\delta\Delta)^{\omega' \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda}$$

Starting from the leaves, our algorithm computes all end views at every bag of $\mathcal{T}$ iteratively. Any plausible focused view at a leaf is an end view, so there is nothing to do at leaves.

Now, let $\mathcal{B}$ be a non-leaf bag and suppose that all end views in the lists of all of its children are computed. Our algorithm considers all plausible maps in $L_{\mathcal{B}}$ one by one, and keeps each $F \in L_{\mathcal{B}}$ only if for every child $\mathcal{B}'$ of $\mathcal{B}$ there is an $F' \in L_{\mathcal{B}'}$ that is consistent with F. Otherwise, it removes

F from $L_{\mathcal{B}}$. To this end, our algorithm tests consistency between every $F \in L_{\mathcal{B}}$ and every $F' \in L_{\mathcal{B}'}$ using Lemma 4.5. The total running time for trimming $L_{\mathcal{B}}$ to keep only the end views is thus,

$$\left(n^{O(\omega')} \cdot (\delta\Delta)^{\omega' \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda}\right)^2 \cdot \left(n^{O(1)} \cdot (\delta\Delta)^{O(\lambda)}\right) = n^{O(\omega')} \cdot (\delta\Delta)^{\omega' \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda}$$

The time to compute all lists of end views is asymptotically the same. In the end, we pick any end view at $\mathcal{R}$ and trace back $\mathcal{T}$ to output a set of focused views $\mathcal{F}$ that are pairwise consistent over the edges of $\mathcal{T}$. Thus, $\mathcal{F}$ is a complete set of focused views by Corollary 6.4, so $h_{\mathcal{F}}$, defined in Equation (1), is an embedding of distortion $(\rho+1)\delta/(\rho-1)$ by Lemma 4.4. To obtain our approximation requirement we set $\rho = 1/(\delta-1) + 2/\epsilon + 5 \geq \max(1/(\delta-1), 4, 2/\epsilon+1)$ (recall that $\rho \geq \max(1/(\delta-1), 4)$ is already required from the definition of focused views). Therefore, the final running time is as in the statement of the lemma. □

Proof (of Theorem 1.1): Our algorithm first computes a tree decomposition $\mathcal{T}$ of G of width $O(\omega)$ via the algorithm of Lemma 2.1 in $m \cdot 2^{O(\omega)}$ time. By Lemma 2.3, there is a list L of $(nm)^{O(1)}$ numbers that contain δ_{opt} and that can be computed in $(nm)^{O(1)}$ time. For each $\beta \in L$, in increasing order, and for each $v \in V_G$, we call the algorithm of Lemma 6.5, with parameters $\mathcal{T}$, $\delta = \beta + \varepsilon/2$ and $\epsilon = \varepsilon/3$, to check if an embedding of distortion close to β from X into $B(v, 4\beta\Delta)$ exists. By Lemma 2.4, if X can embed into G with distortion at most β then it can embed into $B(v, 4\beta\Delta)$ for some $v \in V_G$ with the same distortion.

As $\beta \geq 1$ and $\varepsilon > 0$, we have $\delta > 1$ and $\epsilon > 0$. The algorithm of Lemma 6.5 outputs an embedding with distortion

$$(1+\varepsilon/3)(\beta+\varepsilon/2) \leq (1+\varepsilon)\beta,$$

if $\beta \geq \delta_{opt}$, otherwise, it either outputs an embedding of distortion $(1+\varepsilon)\beta$, or it decides that $\beta < \delta_{opt}$. We stop, and return the embedding, as soon as we obtain one.

The running time of each call to this algorithm is bounded by

$$n^{O(\omega)} \cdot (\delta\Delta)^{\omega \cdot (1+\log_\delta \rho) \cdot \lambda \cdot O\left((\rho\delta^2)\right)^\lambda} \tag{7}$$

Note that,

$$\rho = 1/(\delta-1) + 2/\epsilon + 5 = 1/(\beta+\varepsilon/2-1) + 2/(\varepsilon/3) + 5 \leq 8/\varepsilon + 5 = O(1/\varepsilon).$$

Substituting δ, ϵ and ρ in (7), we find the new bound

$$n^{O(\omega)} \cdot (\delta\Delta)^{\omega \cdot (1+\log_{\beta+\varepsilon/2} 1/\varepsilon) \cdot \lambda \cdot O\left((\beta^2/\varepsilon)\right)^\lambda} \tag{8}$$

We bound $\log_{\beta+\varepsilon/2}(1/\varepsilon)$. As $\beta \geq 1$ and $\varepsilon > 0$, we have

$$\log_{\beta+\varepsilon/2}(1/\varepsilon) = \frac{\log(1/\varepsilon)}{\log(\beta+\varepsilon/2)} \leq \frac{1/\varepsilon}{\log(1+\varepsilon/2)}.$$

We know (via Taylor expansion, and because $\varepsilon > 0$) that $\log(1+\varepsilon/2) \geq (\varepsilon/2)/(1+\varepsilon/2)$. Therefore,

$$\log_{\beta+\varepsilon/2}(1/\varepsilon) \leq \frac{1/\varepsilon}{\varepsilon/(2+\varepsilon)} \leq \frac{3}{\varepsilon^2}.$$

Replacing in (8), we conclude that the running time of one call to the algorithm of Lemma 6.5 is bounded by

$$n^{O(\omega)} \cdot (\delta\Delta)^{\omega \cdot (1/\varepsilon^2) \cdot \lambda \cdot O\left((\beta^2/\varepsilon)\right)^\lambda}.$$

So, the total running time for computing L, and for calls of the algorithm of Lemma 6.5 for all $\beta \leq \delta_{opt}$ in L is bounded by

$$(nm)^{O(1)} + m \cdot n^{O(\omega)} \cdot (\delta_{opt}\Delta)^{\omega \cdot (1/\varepsilon)^{\lambda+2} \cdot \lambda \cdot (O(\delta_{opt}))^{2\lambda}}$$

□

References

[Ass83] Patrice Assouad. Plongements lipschitziens dans $\mathbb{R}^n$. *Bull. Soc. Math. France*, 11(4):429–448, 1983.

[Bar96] Yair Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, FOCS '96, pages 184–, Washington, DC, USA, 1996. IEEE Computer Society.

[BCIS05] Mihai Badoiu, Julia Chuzhoy, Piotr Indyk, and Anastasios Sidiropoulos. Low-distortion embeddings of general metrics into the line. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, STOC '05, pages 225–233. ACM, 2005.

[BDD+13] Hans L. Bodlaender, Pål Grønås Drange, Markus S. Dregi, Fedor V. Fomin, Daniel Lokshtanov, and Michal Pilipczuk. An $O(c^k n)$ 5-approximation algorithm for treewidth. In *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, pages 499–508, Washington, DC, USA, 2013. IEEE Computer Society.

[BDG+05] Mihai Badoiu, Kedar Dhamdhere, Anupam Gupta, Yuri Rabinovich, Harald Räcke, R. Ravi, and Anastasios Sidiropoulos. Approximation algorithms for low-distortion embeddings into low-dimensional spaces. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '05, pages 119–128, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[BDH+08] Mihai Bădoiu, Erik D. Demaine, Mohammadtaghi Hajiaghayi, Anastasios Sidiropoulos, and Morteza Zadimoghaddam. Ordinal embedding: Approximation algorithms and dimensionality reduction. In *Proceedings of the 11th International Workshop, APPROX 2008, and 12th International Workshop, RANDOM 2008 on Approximation, Randomization and Combinatorial Optimization: Algorithms and Techniques*, APPROX '08 / RANDOM '08, pages 21–34, Berlin, Heidelberg, 2008. Springer-Verlag.

[BIS07] Mihai Badoiu, Piotr Indyk, and Anastasios Sidiropoulos. Approximation algorithms for embedding general metrics into trees. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 512–521, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[Bod93] Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *Proceedings of the Twenty-fifth Annual ACM Symposium on Theory of Computing*, STOC '93, pages 226–234, New York, NY, USA, 1993. ACM.

[Bou85] Jean Bourgain. On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52(1-2):46–52, 1985.

[CDN+10] Victor Chepoi, Feodor F. Dragan, Ilan Newman, Yuri Rabinovich, and Yann Vaxes. Constant approximation algorithms for embedding graph metrics into trees and outerplanar graphs. In Maria J. Serna, Ronen Shaltiel, Klaus Jansen, and Jose D. P. Rolim, editors, *APPROX-RANDOM*, volume 6302 of *Lecture Notes in Computer Science*, pages 95–109. Springer, 2010.

[FFL+13] Michael Fellows, Fedor V. Fomin, Daniel Lokshtanov, Elena Losievskaja, Frances Rosamond, and Saket Saurabh. Distortion is fixed parameter tractable. *ACM Trans. Comput. Theory*, 5(4):16:1–16:20, November 2013.

[FRT03] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 448–455, New York, NY, USA, 2003. ACM.

[GKL03] Anupam Gupta, Robert Krauthgamer, and James R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 534–543, 2003.

[IM04] Piotr Indyk and Jiri Matousek. Low-distortion embeddings of finite metric spaces. In *in Handbook of Discrete and Computational Geometry*, pages 177–196. CRC Press, 2004.

[Ind01] Piotr Indyk. Algorithmic applications of low-distortion geometric embeddings. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 10–, Washington, DC, USA, 2001. IEEE Computer Society.

[JL84] William Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in modern analysis and probability (New Haven, Conn., 1982)*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society, 1984.

[KRS04] Claire Kenyon, Yuval Rabani, and Alistair Sinclair. Low distortion maps between point sets. In *Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pages 272–280, New York, NY, USA, 2004. ACM.

[Mat13] Jirí Matousek. *Lecture notes on metric embeddings*, 2013.
Available at: http://kam.mff.cuni.cz/~matousek/ba-a4.pdf.

[NR15] Amir Nayyeri and Benjamin Raichel. Reality distortion: Exact and approximate algorithms for embedding into the line. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 729–747, 2015.