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

Amir Nayyeri∗ Benjamin Raichel†

Abstract

We describe a (1+ε)-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

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

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 bounded degree unweighted tree, and a bounded degree unweighted k-outer planar graph (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 \(f : X \rightarrow Y\), with the expansion \(e_f\) and the contraction \(c_f\) defined as follows.

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

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. In other areas of computer science, such as machine learning, low distortion embeddings are often used as a means to “simplify” the input metric. 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 \(\ell_2^{O(\log^2 n)}\) with \(O(\log n)\) distortion [Bou85], any \(n\)-point Euclidean metric embeds into \(\ell_2^{O(\log n)}\) with constant 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, where \(\delta\) is the minimum distortion. Fellows et al. [FFL^+13] improved their exact algorithm to an \(O(n\delta^3(2\delta + 1)^2)\) 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. 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 \(\delta^{O(\log^2 D)}n^{O(1)}\) 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 \(\Delta^{O(\delta^3)(mn)^{O(1)}}\) time \(O(1)\)-approximation algorithm, where \(m\) is the number of vertices of the path, and \(\Delta\) 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.\footnote{See references \cite{BIS07,BDH08,CDN10} 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.} Kenyon et al. \cite{KRS04} developed an exact $O(n^2 \cdot 2^{\lambda\delta^3})$ time algorithm to compute an optimal bijective embedding from an unweighted tree of maximum degree $\lambda$. Fellows et al. \cite{FFL13} furthered this result by developing a $n^2 \cdot |V(T)| \cdot 2^{O((5\delta)^{\lambda^2+1} \cdot \delta)}$ time algorithm for embedding an unweighted graph into an unweighted tree with maximum degree $\lambda$.

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. Note that as $G$ is a weighted graph in our case, doubling dimension is considered, which is an equivalent but more generally applicable form of the bounded degree assumption from previous results.

Our Result

**Theorem 1.1.** Let $(X,d_X)$ be an $n$-point metric space with spread $\Delta$. Let $G = (V_G,E_G)$ be a weighted graph of treewidth $\omega$ and doubling dimension $\lambda$, with $m$ vertices. Let $\delta_{opt}$ be the minimum distortion of any embedding of $X$ into $V_G$. There is a $\Delta^{\omega \cdot \lambda^2(1/\varepsilon)^{\lambda+2} \cdot (O(\delta_{opt}))^{3\lambda} \cdot n^{O(\omega)} \cdot m^{O(1)}$ 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 $\delta_{opt}$, are all constant. For example, if $X$ has a polynomial spread and $\delta_{opt}$ is a constant, we obtain PTAS’s for embedding $X$ into the following spaces: the line, a cycle, a bounded degree unweighted tree, and a bounded degree unweighted $k$-outer planar graph (for constant $k$).

2 Overview

We provide a high-level overview of our approach, a generalization of the approach used by Nayyeri and Raichel \cite{NR15} for embedding a metric space into the line, which in turn was inspired by techniques introduced by Kenyon et al. \cite{KRS04}, Badiou et al. \cite{BDG05} and Fellows et al. \cite{FFL13}.

A simpler example. To build intuition we consider a very restricted subcase. Let $X$ be a $n$ point metric space. We consider embedding $X$ into the vertex set $V$ of an $n$ point tree $T$, with maximum degree $\lambda$.

Let $f$ denote the minimum distortion embedding. To find $f$ we could guess all possible maps, but this is clearly too expensive. So instead, for each vertex $v \in V$, we guess what $f$ looks like in a small window around $v$. More precisely, consider any function $f : X \to B(v,R)$, where $B(v,R)$ is a ball with center $v$ and radius $R$, and let $f^{-1}[v,R]$ denote the preimage of $B(v,R)$ under $f$. We refer to $f^{-1}[v,R]$ as the view through the window $B(v,R)$. In order to see the “full picture” of $f$ we guess (all possibilities for) its local view, and then stitch together these local views, by sliding this window around $T$.

In the following, assume the smallest and largest inter-point distances in $X$ are 1 and $\Delta$, respectively, and that $f$ is non-contracting, i.e. the distortion $\delta$ is also the expansion. These assumptions are justified in the next section.
Choosing $R$. The difficulty lies in the choice of the radius $R$. From a running time perspective, we want $R$ to be as small as possible as this limits the number of possible views through a window. On the other hand if $R$ is too small then we can get a mismatch between local versus global properties. Ultimately our goal is to perform dynamic programming over the set of possibilities for $f^{-1}[v, R]$, for all $v \in V_G$, hence $R$ must be large enough to give “separability”. Specifically, to ensure that our composition of the views over the different window positions gives a well defined function globally, for each $x \in X$, we need either (i) $x \in \Dom(f)$ or (ii) that the connected component of $T \setminus \{v\}$ that contains $f(x)$ can be deduced by inspecting $f^{-1}[v, R]$.

Perhaps the more stringent requirement on $R$ concerns how the local distortion through a window relates to the global distortion. Note that distortion is a measure concerning pairs of vertices from $X$, so intuitively $R$ should be large enough such that every pair is seen in the view of a window $f^{-1}[v, R]$ for some $v \in V_G$. To ensure this, naively we should set $R = \Delta \delta$. However, this is still far too large from an efficiency stand point, and so we attack this from several perspectives.

Scales and custom windows. Our final running time will be exponential in the number of points which can map into a window. Since the maximum degree in $T$ is $\lambda$, and since $f$ is non-contracting, at most $(O(\Delta \delta))^\lambda$ points can fit in a $\Delta \delta$ window. If all inter-points distances in $X$ are roughly the same, then $\Delta$ is constant (since we already scaled such that the smallest distance is 1). However, assuming uniform inter-point distances is far too restrictive an assumption as it fails to capture many natural metric spaces. So instead we find the next best thing.

Specifically, by using the greedy permutation ordering of $X$ (used in the standard Gonzalez algorithm for $k$-center clustering), we can define a coarse to fine sequence $\{x_i\} = X_{>s} \subseteq X_{>s-1} \subseteq \ldots X_{\geq 1} \subseteq X_{=0} = X$, such that each $X_{>s}$ is a maximal subset of $X$ whose points have mutual distance at least $\delta^s$. Correspondingly, on the $T$ side, around each vertex $v \in V_T$, we consider a collection of nesting windows of radii $r\delta^{s+2}, \ldots, r\delta^3, r\delta^2$. For any given value $s$, the windows of radius $r\delta^{s+2}$ have been custom built for packing the set $X_{>s}$. As the inter-point distances in $X_{>s}$ are at least $\delta^s$, we get a packing condition on the number of points in a window, thus removing the exponential dependence on $\Delta$ (and instead replacing it with $S = \log_\delta \Delta$).

While this multi-scale window approach provides a packing condition, limiting the number of views, it introduces a problem. It is no longer true that for any pair of points in $y, z \in X$, there is some $v \in V_G$, such that $y, z$ are both visible in $f^{-1}[v, R]$. Fortunately (or rather by design), the greedy permutation ensures that for any pair $y, z$, there is always some pair $y', z'$ which are visible in some window view, such that $d(y, z) \approx d(y', z')$, and $y$ and $z$ are close to $y'$ and $z'$, respectively (where “close” means small relative to $d(y, z)$). In particular, this means our embedding will be approximate rather than exactly optimal. However, the extent to which $d(y', z')$ approximates $d(y, z)$, and hence the quality of our approximation, can be controlled by the parameter $r$.

Bins and embedding. Now that we have transitioned to an approximation algorithm, this frees us to make a number of improvements. So far we observed that a window $B(v, r\delta^{s+2})$, can be mapped onto by at most $\delta^2$ points from $X_{>s}$. However, the window $B(v, r\delta^{s+2})$ may still contain many points from $V$ and a priori we don’t know which points get mapped onto by points from $X_{>s}$. Therefore, instead we break this window into $\delta^s$ diameter bins (each of which can contain the image of at most one point from $X_{>s}$), and map to these approximate bins rather than directly to points, intuitively slightly pixelating our view. In addition to improving our running time, this binning allows us to transition from bijections to embeddings (i.e. $|V|$ may be larger than $n$), for the same reason that we no longer need to know exactly which points to map onto.
Bounded treewidth. When one transitions from embedding into trees to embedding into graphs with cycles, immediately the partitioning property required for dynamic programming is lost. By assuming bounded treewidth, we can recover this partitioning, intuitively by applying the above techniques to the bags of the tree decomposition rather than directly to the vertices of the graph. This makes the technique more challenging as now each window becomes a collection of windows over the vertices in a single bag, but at a high level it is the same approach.

3 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 $\text{diam}_G(U)$ to denote the diameter of $G[U]$, the maximum distance between any pair of vertices in $G[U]$.

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 $X'$. Note however that the word “ball” will only be used in reference to a standard single center ball.

Tree decomposition and treewidth. A tree decomposition of a graph $G = (V_G, E_G)$ is a pair $(V_T, T)$, where $V_T = \{B_1, B_2, \ldots, B_k\}$ is a family of subsets of $V_G$ that are called bags, and $T$ is a tree whose vertex set is $V_T$, with the following properties:

(1) $\bigcup_{B \in V_T} B = V_G$.

(2) For each edge $(u, v) \in E_G$ there is a bag $B \in V_T$ that contains both $u$ and $v$.

(3) For any $v \in V_G$, any pair $B_i, B_j \in V_T$, and any $B_t \in V_T$ that is on the unique $B_i$-to-$B_j$ path in $T$, if $v \in B_i \cap B_j$ then $v \in B_t$.

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 $\omega$ 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 3.1 (Bodlaender et al. [BDD+13], Theorem I). Let $G$ be a graph with $m$ vertices and treewidth $\omega$. 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 $\lambda$ such that, for any $R \in \mathbb{R}^+$, any ball of radius $R$ in $X$ can be covered by at most $2^\lambda$ 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 3.2 (Gupta et al. [GKL03], Proposition 1.1). Let \((Y, d_Y)\) be a metric with doubling dimension \(\lambda\), and let \(Y' \subseteq Y\). If all pairwise distances in \(Y'\) are at least \(\ell\), then any ball of radius \(R\) in \(Y\) contains at most \((2R)^\lambda\) points of \(Y'\). \(^2\)

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

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

\[
e_f = \max_{x,x' \in X \text{ } x \ne x'} \frac{d_Y(f(x), f(x'))}{d_X(x, x')} \quad \text{and} \quad c_f = \max_{x,x' \in X \text{ } x \ne x'} \frac{d_X(x, x')}{d_Y(f(x), f(x'))}.
\]

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 \(\leq 1\). 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 \(\delta\) is a known value.

**Lemma 3.3 (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 \(\delta\), or correctly report that no such embedding exists.

### 4 Embedding into graphs with bounded treewidth and bounded doubling dimension

**How to read this section.** The high level overview of the approach in this section was already given in Section 2. To put this plan into action a fair amount of low level definitions need to be setup. These are covered in subsections 4.1 through 4.4, and on a first read we suggest skipping the proofs in these subsections. Subsection 4.5 shows that any function satisfying all the definitions provided will lead to a proper embedding with bounded distortion, and as such, other than the initial definition of the function \(h\), can also be skipped on a first read. Subsection 4.6 then provides the algorithm and proof for our approach.

Let \((X, d_X)\) be a discrete metric space over \(X = \{x_1, x_2, \ldots, x_n\}\). Assume that for any pair \(x, y \in X, 1 \leq d_X(x, y) \leq \Delta\), and \(\min_{x,y \in X} d_X(x, y) = 1\). \(\Delta\) will be called the spread of \(X\). Also, let \(G = (V_G, E_G)\) be a weighted graph, where \(|V_G| = m \geq n\). Let \((V_G, d_G)\) be the induced shortest path metric space of \(G\), and let \(\lambda\) denote its doubling dimension. Finally, let \(T = (V_T, E_T)\) be a tree decomposition of \(G\) of width \(\omega\) \((V_T\) is composed of bags, each containing at most \(\omega + 1\) vertices of \(G\)). Fix \(\delta > 0\) to be a value so that \(X\) embeds into \(G\) with distortion at most \(\delta\).

\(^2\)Note that \(\lambda\) in their paper is the doubling constant, whereas in this paper it denotes the doubling dimension.
Lemma 4.1. 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 \( \delta \).

**Proof:** Let \( f : X \to V_G \) be a non-contracting embedding of expansion \( \delta \), and let \( v \in V_G \) be any vertex that \( f \) maps onto. As the expansion is at most \( \delta \) the image of \( f \) is within \( B(v, \delta\Delta) \). Let \( u, u' \in V_G \) be any two vertices that \( f \) maps onto, so \( u, u' \in B(v, \delta\Delta) \). Consider the shortest \( u \)-to-\( u' \) path \( \gamma \) in \( G \). Any vertex on this path has distance at most \( 2\delta\Delta \) from \( v \), thus, \( B(v, 2\delta\Delta) \) includes the image of \( f \) as well as the shortest path between any pair of vertices in this image. \( \square \)

**Remark 4.2.** In light of Lemma 4.1, we assume, in the rest of this section, 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.

### 4.1 Scales and Bins

**Scales 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 \( 2 \leq i \leq n \), the point \( x_i \in X \setminus \{x_1, \ldots, x_{i-1}\} \) is the farthest point from the set \( \{x_1, \ldots, x_{i-1}\} \). For each \( s \in \{0, 1, \ldots, S = \lfloor \log_\delta \Delta \rfloor + 1 \} \), the set \( X_{\geq s} \) 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 \( \delta^s \). Note that \( X_{\geq S} = \{x_1\} \), as \( \Delta < \delta^S \). The scale of a point \( x \in X \), is the largest \( s \) such that \( x \in X_{\geq s} \).

For the remainder of the paper we assume that we have precomputed the sets \( X_s = X_{\geq s} \setminus X_{\geq s+1} \), for all \( s \in \{0, 1, \ldots, S-1\} \). Additionally, we also precompute for each \( x \in X \) and for each \( 0 \leq s < S \), the nearest neighbor of \( x \) in \( X_{\geq s} \). This can all be done in \((Sn)^O(1)\) time. We find the following result of Lemma 3.2 helpful in the paper.

**Lemma 4.3.** Any ball \( B \) of radius \( R \) in \( X \) contains at most \( \left( \frac{2R}{\delta^s - 1} \right)^\lambda \) points from \( X_{\geq s} \).

**Proof:** Let \( f : X \to V \) be a non-contracting distortion \( \delta \) embedding. The image \( f(B) \) will be contained in a ball \( B' \) of radius \( \delta R \) of \( G \). Additionally, since \( f \) is non-contracting the distance between any pair of points in \( f(X_{\geq s}) \) is at least \( \delta^s \). Also, \( |f(X_{\geq s}) \cap B'| \geq |X_{\geq s} \cap B| \). Lemma 3.2 then implies that

\[
|f(X_{\geq s}) \cap B'| \leq \left( \frac{2\delta R}{\delta^s} \right)^\lambda = \left( \frac{2R}{\delta^s - 1} \right)^\lambda
\]

\( \square \)

**Bins in \( G \).** In order to obtain a faster algorithm, we consider approximate maps into bins rather than into the vertices of \( G \). A family of bins \( \{\sqcup_0, \sqcup_1, \ldots, \sqcup_{S+1}\} \) of local density \( \alpha \) in \( G \) is a family of partitions of \( V_G \) with the following properties:

1. For each \( 0 \leq s < S \) and \( b \in \sqcup_s \), \( G[b] \) is a subgraph with diameter smaller than \( \delta^s \).

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

Each set in \( \sqcup_s \) is called a bin of scale \( s \). In particular, a bin of scale zero contains exactly one vertex.

**Lemma 4.4.** There exists a family of bins of local density \( \lambda \) in \( G \) that can be computed in \( m^{O(1)} \) time.
The Gonzalez permutation and the set of bins in every scale can be computed in $m$.

Let $\mathcal{B} = \{V_G, E_G\}$ be a subgraph of diameter $D$. Let $v' \in V_G$, and consider the ball $B = B(v', D + \delta^s)$. 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 $U \subseteq V_{s}$, the mutual distance between points of $U$ is at least $\delta^s/2$. Therefore, Lemma 3.2 implies:

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

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

For any $U \subseteq V_G$, we define $\mathcal{B}[U, r]$ to be composed of the scale $s$ bins that intersect $B(U, r)$. Note that there may be sets in $\mathcal{B}[U, r]$ that are not contained in $B(U, r)$.

4.2 Approximate Maps, and their Induced Partitions

Approximate maps. Let $r \in \mathbb{R}$ and $r \geq 20$. A collection of partial maps $F = \{f_0, f_1, \ldots, f_{s-1}\}$, where for each $0 \leq s < S$, $f_s : X_{s} \rightarrow \mathcal{B}, r\delta^{s+2}$, is called an approximate map into $B$. Note that the partial maps in $F$ do not need to be consistent, for example, two partial maps in $F$ may map the same point of $x$ into disjoint bins. Also, any $f_s$ can be the empty partial map. We focus on more useful approximate maps in the next subsection where we introduce the notion of plausibility for an approximate map.

Restriction and Extension. Let $f$ be an embedding of $X$ into $G$, and let $f_s : X_{s} \rightarrow \mathcal{B}, r\delta^{s+2}$ be a scale $s$ map (from vertices in $X_{s}$ into bins in $\mathcal{B}[B, r\delta^{s+2}]$). Also, let $Q = \bigcup_{U \subseteq \mathcal{B}[B, r\delta^{s+2}]} U$ that is $Q$ is composed of all vertices of the bins in $\mathcal{B}[B, r\delta^{s+2}]$. In particular, $Q \supseteq B(B, r\delta^{s+2})$ (note that $Q$ might be a proper superset of $B(B, r\delta^{s+2})$). The partial function $f_s$ is an approximate restriction of $f$ into $B$ at scale $s$ if the following properties hold:

1. For any $x \in X_{s}$ such that $f(x) \in Q$, $f_s(x)$ is defined and $f(x) \in f_s(x)$.
2. For all other $x \in X$, $x \notin \text{Dom}(f_s)$.

Under these conditions, we also say that $f$ is an extension of $f_s$.

The approximate map $F = \{f_0, f_1, \ldots, f_{s-1}\} \rightarrow \mathcal{B} \in V_{T}$ is the approximate restriction of the embedding $f$ (into $B$) if for each $s \in \{0, \ldots, S-1\}$, $f_s$ is the approximate restriction of $f$ into $B$ at scale $s$. Under these conditions, we also say that $f$ is an extension of $F$.

An approximate map $F = \{f_0, f_1, \ldots, f_{s-1}\}$ is feasible if it has a non-contracting extension, $f$, of expansion at most $\delta$. In this case $f$ is a feasible extension of $F$. Conversely, it is easy to verify that the restriction of a non-contracting and expansion at most $\delta$ embedding of $X$ into $G$ satisfies all the requirements defining an approximate map.
4.3 Partitions

Let $\mathcal{B} \in V_T$. Let $\mathcal{T}_1, \ldots, \mathcal{T}_t$ be the connected components of $\mathcal{T} \setminus \{\mathcal{B}\}$. Let $F$ be an approximate map into $\mathcal{B}$, and let $f$ be a feasible extension of $F$. For each $0 \leq s < S$, and for each $x \in X_{\geq s}$,

1. $x \in C(F, s)$ if $x \in \text{Dom}(f_s)$, and
2. $x \in P(f, F, s, \mathcal{T}_i)$ if $x \notin C(F, s)$ and $f(x) \in V(\mathcal{T}_i) \setminus V(\mathcal{B})$.

**Lemma 4.5.** Let $F$ be an approximate map into $\mathcal{B}$. Let $f : X \rightarrow V_G$ and $f' : X \rightarrow V_G$ be two feasible extensions of $F$. We have:

1. The sets $C(F, s), P(f, F, s, \mathcal{T}_1), \ldots, P(f, F, s, \mathcal{T}_t)$ form a partition of $X_{\geq s}$.
2. For each $1 \leq i \leq t$, $P(f, F, s, \mathcal{T}_i) = P(f', F, s, \mathcal{T}_i)$.

**Proof:** Let $x \in X_{\geq s}$. First, we show that $x \in C(F, s) \cup P(f, F, s, \mathcal{T}_1) \cup \ldots \cup P(f, F, s, \mathcal{T}_t)$. Equivalently, for any $x \notin C(F, s)$ we show there is an $1 \leq i \leq t$ such that $x \in P(f, F, s, \mathcal{T}_i)$. Since $F$ is an approximate restriction of $f$, $x \notin C(F, s)$ implies $f(x) \notin \mathcal{B}$, therefore, there is an $1 \leq i \leq t$ such that $f(x) \in V(\mathcal{T}_i)$ (as $\mathcal{T}$ is a tree decomposition of $G$). Second, we show that $x$ is in exactly one of the sets $C(F, s), P(f, F, s, \mathcal{T}_1), \ldots, P(f, F, s, \mathcal{T}_t)$. For each $1 \leq i \leq t$, we have $P(f, F, s, \mathcal{T}_i) \setminus C(F, s) = \emptyset$ by the definition of $P(f, F, s, \mathcal{T}_i)$. For $1 \leq i < j \leq t$, we have $(V(\mathcal{T}_i) \setminus V(\mathcal{B})) \cap (V(\mathcal{T}_j) \setminus V(\mathcal{B})) = \emptyset$ (by the definition of tree decomposition), therefore, $P(f, F, s, \mathcal{T}_i) \cap P(f, F, s, \mathcal{T}_j) = \emptyset$. This completes the proof of Property (1).

To prove Property (2), we show that given $F$ (into $\mathcal{B}$) the condition of any point in any feasible extension is uniquely determined. We present the following algorithmic proof for this statement.

By Remark 4.2, $B(B, r\delta^{s+1})$ contains all $V_G$. Therefore, $f_{S-1}$, by its definition, should act on all $X_{\geq S-1}$, that is $C(F, S-1) = X_{\geq S-1}$. Now, assume that the condition of any point in $X_{\geq S+1}$ is uniquely determined, and let $x \in X_{\geq s} \setminus X_{\geq s+1}$, and $x \notin C(F, s)$. We show there is exactly one $1 \leq i \leq t$ such that $x \in P(f, F, s, \mathcal{T}_i)$. Property (2) will follow by induction.

Let $z$ be $x$'s nearest neighbor in $X_{\geq s+1}$. By the definition of $X_{\geq s+1}$, we have $d_X(x, z) \leq \delta^{s+1}$ (as otherwise $x \in X_{\geq s+1}$), and so for any feasible $f$ we have:

$$d_G(f(x), f(z)) \leq \delta^{s+2}. \quad (1)$$

We consider two cases:

1. $z \notin C(F, s+1)$: By the induction hypothesis $z \in P(f, F, s+1, \mathcal{T}_i)$ for exactly one $1 \leq i \leq t$. We claim that $x \in P(f, F, s, \mathcal{T}_i)$. Suppose, to derive a contradiction, that $x \notin P(f, F, s, \mathcal{T}_i)$, for $j \neq i$. Therefore, any $f(x)$-to-$f(z)$ path in $G$ must pass through $\mathcal{B}$. But, the distance of $f(z)$ from $\mathcal{B}$ is at least $r\delta^{s+3}$ (since $f(z) \notin C(F, s+1)$). Thus, $d_G(f(x), f(z)) \geq r\delta^{s+3}$, which is a contradiction because of (1).

2. $z \in C(F, s+1)$, let $b = f_{s+1}(z)$. We consider two (sub-)cases:

   i. The bin $b$ intersects $\mathcal{B}$, therefore, $d_G(\mathcal{B}, f(z)) \leq \delta^{s+1}$. Since $d_G(f(x), f(z)) \leq \delta^{s+2}$, we have $d_G(f(x), \mathcal{B}) \leq 2\delta^{s+2}$. Thus, $f(x) \in B(\mathcal{B}, r\delta^{s+2})$, therefore, $x \in C(F, s)$, which is a contradiction because of (1).

   ii. The bin $b$ does not intersect $\mathcal{B}$, therefore, there is an $1 \leq i \leq t$ such that $b \in V(\mathcal{T}_i) \setminus V(\mathcal{B})$. We claim $x \in P(f, F, s, \mathcal{T}_i)$ in this case. Otherwise, any $f(x)$-to-$f(z)$ path passes through $\mathcal{B}$. As $x \notin C(F, s)$ we have $d_G(f(x), f(z)) \geq d_G(f(x), \mathcal{B}) \geq r\delta^{s+2}$, which is a contradiction because of (1).
Lemma 4.5 implies that all feasible extensions of $F$ induce the same partition of each $X\geq s$. Therefore, we can define $P(F,s,T_i) = P(f,F,s,T_i)$ for an arbitrary feasible $f$. We conclude that $C(F,s), P(F,s,T_1), \ldots, P(F,s,T_t)$ is a partition of $X\geq s$.

Lemma 4.5 also gives an algorithm which given a feasible $F$ into a bag $B$, outputs the partitions $C(F,s), P(F,s,T_1), \ldots, P(F,s,T_t)$ at all scales $0 \leq s < S$. If $F$ is not feasible, then ideally this could be detected, however being able to do so without knowing the extension $f$ seems unlikely. Therefore, if $F$ is not feasible the algorithm either returns that $F$ is infeasible if a partition is not produced or outputs some bogus partition. The following corollary formalizes this statement.

**Corollary 4.6.** Given an approximate map $F$ into a bag $B$, there is $(S \cdot n \cdot m)^O(1)$ time algorithm such that if $F$ is feasible it outputs the corresponding partition of $X\geq s$ into sets $C(F,s), P(F,s,T_1), \ldots, P(F,s,T_t)$ for all $0 \leq s < S$, as described above. If $F$ is not feasible it either outputs a partition or returns that $F$ is infeasible.

**Remark 4.7.** By Remark 4.2, the diameter of $G$ is at most $4\delta \Delta \leq 4\delta^{S+1}$, therefore, any ball of radius $r\delta^{S+1}$ contains all $G$. Thus, the algorithm of Corollary 4.6 returns the trivial partition $C(F,S-1) = X_{\geq S-1}$ at scale $S-1$ if $F$ is feasible. In light of this observation, we assume, that $f_{S-1}$ acts on all $X_{\geq S-1}$ in the rest of the paper.

### 4.4 Plausibility and Consistency

**Plausibility.** An approximate map is not necessarily feasible. In fact, it is not necessarily extendable to any embedding, for example when it maps a point to disjoint bins in different scales. Here, we define plausibility for approximate maps. Intuitively, an approximate map is plausible if one cannot conclude it is not feasible by locally examining it.

Let $B \in \mathcal{T}$, and let $F = \{f_0, \ldots, f_{S-1}\}$ be an approximate map into $B$. Each $f_s$ assigns a scale $s$ bin to each point in its domain, which intuitively is an estimation of the image of the point in an optimal solution. Lemma 4.5 extends these estimations to the points that are not in the domain of $f_s$ by assigning them to a partition $P(F,s,T_i)$. Therefore, $F$ together with Lemma 4.5 gives an estimation for the image of all points in $X_{\geq s}$ in an optimal solution at each scale $s$. Corollary 4.6 ensures that these estimations are computable in polynomial time. The following definition, formalizes this idea by introducing a function $\tilde{f}_s : X_{\geq s} \to 2^{V_G}$ for each $0 \leq s < S$.

1. If $x \in C(F,s)$ then $\tilde{f}_s(x) = f_s(x)$.
2. If $x \in P(F,s,T_i)$ then $\tilde{f}_s(x) = V(T_i) \setminus B(B,r\delta^{s+2})$.

(Lemma 4.5 implies that $\tilde{f}_s$ acts on all $X_{\geq s}$.) In turn, we define $\tilde{F} : X \to 2^V$ as follows.

$$\tilde{F}(x) = \bigcap_s \tilde{f}_s(x)$$

Intuitively, $F$ is plausible if one cannot conclude that it is not feasible by examining $\tilde{F}$. Formally, $F$ is **plausible** if the following properties hold.
(1) For all \( x \in X, \tilde{F}(x) \neq \emptyset \).

(2) For all \( x, y \in X \), there are \( u \in \tilde{F}(x) \) and \( v \in \tilde{F}(y) \) such that:

\[
1 \leq d_G(u, v)/d_X(x, y) \leq \delta.
\]

In particular, if \( F \) is feasible then it is plausible, but not necessarily the other way around. The following lemma ensures that plausibility can be checked in polynomial time.

**Lemma 4.8.** Let \( F \) be an approximate map into \( B \). There is a \( (S \cdot n \cdot m)^{O(1)} \) time algorithm to decide whether \( F \) is plausible or not.

**Proof:** By Corollary 4.6, \( \tilde{f}_s \), for all \( 1 \leq s < S \), can be computed in \( (S \cdot n \cdot m)^{O(1)} \) time. For each \( x \in X \) the intersection of all \( \tilde{f}_s(x) \)'s can be computed in \( O(S \cdot m^2) \) time as each \( \tilde{f}_s(x) \) contains at most \( m \) vertices. Therefore, if all \( \tilde{f}_s \)'s are provided, \( \tilde{F} \) can be computed in \( (S \cdot n \cdot m)^{O(1)} \) time. Finally, Property (2) of a plausible approximate map for all pairs \( x,y \in X \) can be checked in the same asymptotic running time. \( \square \)

Next, we bound the number of plausible approximate maps into a bag. We start by showing that a partial map of any scale in an approximate map must be one-to-one.

**Lemma 4.9.** If \( F = \{f_0, \ldots, f_{S-1}\} \) is a plausible approximate map, then for each \( 0 \leq s < S \), \( f_s \) is one-to-one.

**Proof:** Let \( x, y \in X_{\geq s} \), and suppose that \( f_s(x) = f_s(y) \). Since \( f_s(x) \) is a bin of diameter smaller than \( \delta^s \), for any \( u, v \in f_s(x) \), we have \( d_G(u, v) < \delta^s \). Since \( x, y \in X_{\geq s} \), \( d_X(x, y) \geq \delta^s \), which implies that \( F \) is not plausible, as any of its extensions must be contracting. \( \square \)

**Lemma 4.10.** Let \( B \) be a bag of the tree decomposition of \( G \). There are \( n^{O(|B|)} \cdot \Delta^{|B|} \cdot \lambda \cdot (O(r \delta^2))^{\lambda \cdot \log_s r} \) plausible approximate maps into \( B \). Moreover, there is an algorithm to list these plausible approximate maps in \( \Delta^{|B|} \cdot \lambda \cdot (O(r \delta^2))^{\lambda \cdot \log_s r} \cdot n^{O(|B|)} \cdot m^{O(1)} \) time.

**Proof:** Let \( F = \{f_0, \ldots, f_{S-1}\} \) be a plausible approximate map into \( B \). For every \( 0 \leq s < S \) and for each \( v \in B \) let \( f_s[v] \) denote the range restriction of \( f_s \) into \( \sqcup_s (v, r \delta^{s+2}) \). Also, suppose \( x = f_0^{-1}(v) \), our algorithm tries all possible \( n \) choices to find \( x \).

As \( f_s[v] \) is non-contracting its preimage is contained in \( B_x = B_X(x, 2r \delta^{s+2}) \) (we need the factor of 2 as some of the bins in \( \sqcup_s (v, r \delta^{s+2}) \) are not completely inside \( B(v, r \delta^{s+2}) \)). Thus, \( f_s[v] \) is a one-to-one partial map from \( B_x \cap X_{\geq s} \) into \( \sqcup_s (v, r \delta^{s+2}) \). The number of such one-to-one partial maps is at most

\[
(|B_x \cap X_{\geq s}| + 1)^{|\sqcup_s (v, r \delta^{s+2})|}.
\]

There are \( |B_x \cap X_{\geq s}| + 1 \) choices for the preimage of each bin under \( f_s \); either it is empty or it is in \( B_x \cap X_{\geq s} \).

By Lemma 4.3,

\[
|B_x \cap X_{\geq s}| \leq \left( \frac{4r \delta^{s+2}}{\delta^{s} - 1} \right)^{\lambda} = (O(r \delta^3))^{\lambda}
\]

On the other hand,

\[
|\sqcup_s (v, r \delta^{s+2})| = \left( O\left( \frac{r \delta^{s+2}}{\delta^{s}} \right) \right)^{\lambda} = (O(r \delta^2))^{\lambda},
\]

From this and the fact that \( n^{O(|B|)} \cdot \Delta^{|B|} \cdot \lambda \cdot (O(r \delta^2))^{\lambda \cdot \log_s r} \), we can conclude that the algorithm runs in \( n^{O(|B|)} \cdot \Delta^{|B|} \cdot \lambda \cdot (O(r \delta^2))^{\lambda \cdot \log_s r} \cdot n^{O(|B|)} \cdot m^{O(1)} \) time.
which follows from Property (2) of the family of bins and Lemma 4.4, which ensures the existence of a family of bins of local density $\lambda$.

So, the number of one-to-one partial maps, given in (2), is bounded by

$$\left(\left(O(r\delta^3)\right)^\lambda\right)^{O(r\delta^2)} = (r\delta)^\lambda (O(r\delta^2))^\lambda.$$ 

Therefore, the number of choices for all $f_0[v], f_1[v], \ldots, f_{S-1}[v]$, is

$$n \cdot (r\delta)^{S\lambda (O(r\delta^2))^\lambda},$$

where we need the factor of $n$ to guess $f_0^{-1}(v)$.

We conclude that the total number of choices for $F = \{f_0, \ldots, f_{S-1}\}$ into $B$ is

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

By Lemma 4.8, we can check if each of the enumerated candidates is plausible in $(S \cdot n \cdot m)^{O(1)}$ time. Also, $S = O(\log \frac{1}{\delta} \Delta)$. Thus, the total time to list all plausible maps into $B$ is bounded by

$$(r\delta)^{\log \frac{1}{\delta} |B| \cdot \lambda \cdot (O(r\delta^2))^\lambda} \cdot n^{O(|B|)} \cdot m^{O(1)} = \Delta^{\frac{1}{\delta} \cdot \lambda \cdot (O(r\delta^2))^\lambda} \cdot n^{O(|B|)} \cdot m^{O(1)}$$

Consistency. A feasible embedding induces plausible approximate maps into the bags of $T$. Conversely, we would like to infer an almost feasible embedding provided plausible approximate maps into the bags of $T$. Intuitively, we need to ensure that these approximate maps are consistent on where they map points of $X$. We formalize this notion of consistency here by inspecting the maps at adjacent bags of $T$.

Let $(B, B') \in E_T$. Let $T_1, \ldots, T_t$ be the connected components of $T \setminus \{B\}$, and let $T'_1, \ldots, T'_t$ be the connected components in $T \setminus \{B'\}$. Also, let $T_1$ be the connected component of $T \setminus \{B\}$ that contains $B'$, and let $T'_1$ be the connected component of $T \setminus \{B'\}$ that contains $B$. See Figure 1.

Figure 1. An edge $(B, B')$ of $T$, and the connected components of $T \setminus \{B\}$ and $T \setminus \{B'\}$ colored red and blue, respectively.

Let $F = \{f_0, \ldots, f_{S-1}\}$ and $F' = \{f'_0, \ldots, f'_{S-1}\}$ be plausible approximate maps into $B$ and $B'$, respectively. We say that $F$ and $F'$ are consistent if for any $0 \leq s < S$, and for any $x \in X_{\geq s}$ we have the following properties.

(1a) If $x \in \bigcup_{2 \leq i \leq t} P(F, s, T_i)$ then $x \in P(F', s, T'_i)$.

(1b) If $x \in \bigcup_{2 \leq i \leq t} P(F', s, T'_i)$ then $x \in P(F, s, T_1)$.  

11
(2a) If $x \in C(F,s)$ then $x \in C(F',s) \cup P(F',s,\mathcal{T}_1')$.

(2b) If $x \in C(F',s)$ then $x \in C(F,s) \cup P(F,s,\mathcal{T}_1)$.

(3a) If $x \in C(F,s)$, and $f_s(x) \cap B(\mathcal{B}',r\delta^{s+2}) \neq \emptyset$ then $x \in \text{Dom}(f_s')$ and $f_s'(x) = f_s(x)$.

(3b) If $x \in C(F',s)$, and $f_s'(x) \cap B(\mathcal{B},r\delta^{s+2}) \neq \emptyset$ then $x \in \text{Dom}(f_s)$ and $f_s(x) = f_s'(x)$.

Lemma 4.11. Let $\mathcal{B}$ and $\mathcal{B}'$ be adjacent bags of $\mathcal{T}$, and let $F$ and $F'$ be plausible approximate maps into $\mathcal{B}$ and $\mathcal{B}'$, respectively. There is a $(S \cdot n \cdot m)^O(1)$ time algorithm to decided whether $F$ and $F'$ are consistent or not.

**Proof:** By Corollary 4.6 the partitions of $X$ with respect to $F$ and $F'$ can be computed in $(S \cdot n \cdot m)^O(1)$ time. The conditions of consistency for each $x \in X$ can be checked in constant time. Therefore, the total running time is $(S \cdot n \cdot m)^O(1)$. □

Lemma 4.12. If there exists a non-contracting expansion $\delta$ embedding of $X$ into $V_G$ then there are plausible approximate maps into each bag in $V_T$ that are mutually consistent over edges of $E_T$.

**Proof:** Let $f : X \rightarrow V_G$ be a non-contracting embedding of expansion $\delta$. For each $\mathcal{B}$, let the approximate map into $\mathcal{B}$ be the approximate restriction of $f$ into $\mathcal{B}$. The plausibility of each of these approximate maps and the consistency of each pair of them are immediate as they are restrictions of a feasible map. □

### 4.5 Bounded Distortion Embedding implied by Consistent Maps

Let $\mathcal{U}$ be a set of plausible approximate maps into the bags of $\mathcal{T}$, which contains exactly one approximate map into each bag of $\mathcal{T}$. We say that $\mathcal{U}$ is a **consistent set of approximate maps** into $\mathcal{T}$ if for each $F, F' \in \mathcal{U}$ we have: if $F$ and $F'$ are into adjacent bags then they are consistent.

Lemma 4.12 shows that the restriction of a feasible $f$ into the bags of $\mathcal{T}$ gives a set of consistent approximate maps. Conversely, we show that a bounded distortion embedding can be inferred from a consistent set of approximate maps $\mathcal{U}$ into $\mathcal{T}$. Specifically, we prove the following lemma in the rest of this subsection.

**Lemma 4.13.** Let $\mathcal{U}$ be a consistent set of approximate maps into $\mathcal{T}$, let $r \geq 20$, and let $h : X \rightarrow V_G$ be defined as follows.

$$h = \bigcup_{\{f_0,\ldots,f_{s-1}\} \in \mathcal{U}} f_0.$$

We have that $h$ is an embedding of distortion at most

$$(1 + \frac{20}{r}) \cdot \delta.$$

#### 4.5.1 An embedding

To prove that $h$ is an embedding we find the following lemma helpful. Intuitively, this lemma ensures that the estimations provided for the image of each $x \in X$ by different maps in $\mathcal{U}$ are not conflicting.

**Lemma 4.14.** Let $\mathcal{B}, \mathcal{B}' \in V_T$, and let $F, F' \in \mathcal{U}$ be approximate maps into $\mathcal{B}$ and $\mathcal{B}'$, respectively, where $F = \{f_0,\ldots,f_{s-1}\}$ and $F' = \{f'_0,\ldots,f'_{s-1}\}$. For any $0 \leq s < S$ and for any $x \in C(F,s)$ we have the following properties.
(1) If \( x \in C(F', s) \) then \( f_s(x) = f'_s(x) \).

(2) If \( x \in P(F', s, T') \) then

(i) \( B \in T' \), and
(ii) \( f_s(x) \cap B(B', r\delta^{s+2}) = \emptyset. \)

Proof: Let \( \mathcal{B} = \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_k = \mathcal{B}' \) be the \( \mathcal{B} \)-to-\( \mathcal{B}' \) path in \( \mathcal{T} \). For each \( 1 \leq i \leq k \), let \( F^i = \{ f'_0, \ldots, f'_S \} \) be the approximate map into \( \mathcal{B}_i \), in particular, \( F^1 = F \) and \( F^k = F' \). We consider the two cases one by one.

(1) \( (x \in C(F', s)) \) Properties (1) and (2) (of consistent maps) imply that for all \( 1 \leq i \leq k \), \( x \in C(F', s). \) Property (3) over the edges \((\mathcal{B}_i, \mathcal{B}_{i+1})\) (for \( 1 \leq i < k \)) implies

\[
 f_s(x) = f'_s(x) = f''_s(x) = \ldots = f'_s(x) = f'_s(x)
\]

(2) \( (x \notin C(F', s)) \) Let \( 1 \leq j \leq k \) be the smallest value such that \( x \in C(F^{j-1}, s) \) and \( x \notin C(F^j, s) \).

Property (2) implies that \( x \in P(F_j, s, T^{j-1}) \) where \( T^{j-1} \) is the subtree in \( \mathcal{T} \setminus \mathcal{B}_j \) that contains \( \mathcal{B}_{j-1} \). Then, property (1) implies for all \( j < i \leq k \), \( x \in P(F_i, s, T^{i-1}) \) where \( T^{i-1} \) is the subtree in \( \mathcal{T} \setminus \mathcal{B}_i \) that contains \( \mathcal{B}_{i-1} \). In particular, \( x \in P(F^k, s, T^{k-1}) \) where \( T^{k-1} \) is the subtree in \( \mathcal{T} \setminus \mathcal{B}_k \) that contains \( \mathcal{B}_{k-1} \). Since \( (\mathcal{B} = \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_k = \mathcal{B}') \) is a path in \( \mathcal{T} \), the subtree \( T^{k-1} \) must contain \( \mathcal{B}_1 = \mathcal{B} \).

Property (3) implies that \( f_s(x) \cap B(\mathcal{B}_j, r\delta^{s+2}) = \emptyset. \) Since any path from \( \mathcal{B}_k \) to \( f_s(x) \) passes through \( \mathcal{B}_j \), we have \( d_G(\mathcal{B}_j, f_s(x)) \leq d_G(\mathcal{B}_k, f_s(x)) \), therefore, \( f_s(x) \cap B(\mathcal{B}_k, r\delta^{s+2}) = \emptyset. \)

Next, we show that \( h \) is a valid embedding.

Lemma 4.15. Let \( \mathcal{U} \) be a consistent set of approximate maps into \( \mathcal{T} \). The map \( h : X \to V_G \), defined in (3), is an embedding of \( X \) into \( V_G \).

Proof: We prove the following three statements, which together imply the lemma.

(1) \( h \) acts on all members of \( X \). We prove two facts for all \( 0 \leq s < S \) and \( x \in X_{S_s} \): (i) For any \( 0 < a < S \), if a scale \( a \) map acts on \( x \) then a scale \( a - 1 \) map acts on \( x \). (ii) A scale \( s \) map acts on \( x \). We conclude from (i) and (ii) that a scale zero map acts on \( x \), therefore, \( h \) acts on it.

(i) Let \( f'_a \in F' \) be a scale \( a \) map that acts on \( x \). Let \( v \in V_G \) be any vertex in \( f'_a(x) \), a bin with diameter at most \( \delta^a \), let \( \mathcal{B} \in V_T \) be any bag that contains \( v \), and let \( F = \{ f_0, \ldots, f_{S-1} \} \) be the plausible approximate map of \( \mathcal{U} \) into \( \mathcal{B} \). Lemma 4.14 implies that \( f_a \) acts on \( x \), and that \( f_a(x) = f'_a(x) \). Since \( F \) is plausible, and \( B(\mathcal{B}, r\delta^{a+1}) \) contains \( f_a(x) \), the map \( f_{a-1} \) must act on \( x \).

(ii) If \( x \in X_{S_{S-1}} \) then Remark 4.7 implies statement (ii). We use (reverse) induction to prove statement (ii) for all \( 0 \leq s < S \). Let \( x \in X_{S_s} \setminus X_{S_{s+1}} \), and let \( z \) be the closest point of \( X_{S_{s+1}} \) to \( x \). By the induction hypothesis, there is a scale \( s + 1 \) map that acts on \( z \), so by property (i) there is a scale zero map \( f'_0 \in F \) that acts on \( z \). Let \( v = f'_0(z) \), let \( \mathcal{B} \) be any bag that contains \( v \), and let \( F = \{ f_0, f_1, \ldots, f_{S-1} \} \) be the approximate map into \( \mathcal{B} \). We have, \( x \in C(F, s) \), as otherwise the distance between the image of \( x \) and \( z \) in any extension of \( F \) is at least \( r\delta^{s+2} \) contradicting the plausibility of \( F \).
(2) \emph{h is a function.} Equivalently, for \( F = \{ f_0, \ldots, f_{S-1} \} \) and \( F' = \{ f'_0, \ldots, f'_{S-1} \} \), if \( x \in C(F, 0) \cap C(F', 0) \) then \( f_0(x) = f'_0(x) \), which is implied by Lemma 4.14.

(3) \emph{h is one-to-one.} Let \( F = \{ f_0, \ldots, f_{S-1} \} \) and \( F' = \{ f'_0, \ldots, f'_{S-1} \} \) be plausible approximate maps into bags \( B \) and \( B' \), respectively. Suppose that \( f_0(x) = v \) and \( f'_0(y) = v \), for \( x, y \in X \) and \( v \in V_G \). Let \( (B = B_1, B_2, \ldots, B_k = B') \) be the \( B \)-to-\( B' \) path in \( T \). For each \( 1 \leq i \leq k \), let \( F_i \) be the approximate map of \( U \) into \( B_i \), in particular, \( F^1 = F \) and \( F^k = F' \). We show that \( v \) is within distance \( r\delta^2 \) of all \( B_i \)'s, thus, all \( F^0_i \)'s map the same vertex of \( X \) into \( v \), in particular, \( x = y \).

Since \( v \in \text{Im}(f_0) \), and \( v \in \text{Im}(f'_0) \), it is within distance \( r\delta^2 \) of \( B \) and \( B' \). Let \( B'' \) be any bag in \( (B = B_1, B_2, \ldots, B_k = B') \). \( B \) and \( B' \) are in different connected components of \( T \setminus B'' \). Therefore, the shortest path from \( B \) to \( v \) or the shortest path from \( B' \) to \( v \) (in \( G \)) must intersect \( B'' \). So, the distance of \( B'' \) from \( v \) is not larger than the maximum of the distances of \( B \) and \( B' \) from \( v \). Hence, \( B'' \) is within distance \( r\delta^2 \) of \( v \).

\[ \square \]

### 4.5.2 A bounded distortion embedding

We bound the distortion of \( h \) by bounding its contraction and expansion. To this end, images of \( x \) under different approximate maps (that are bins of different scales in \( G \)) are considered as estimates for \( h(x) \). The following lemma ensures that these estimates are consistent with \( h(x) \).

**Lemma 4.16.** Let \( U \) be a consistent set of approximate maps into \( T \), and let \( h \) be as defined in (3). Also, let \( x \in X \geq s \), and let \( F = \{ f_0, \ldots, f_{S-1} \} \in U \) be an approximate map into \( B \in V_T \).

1. If \( x \in C(F, s) \) then \( h(x) \in f_s(x) \).
2. If \( x \in P(F, s, T) \) then \( h(x) \in V(T') \) and \( h(x) \notin B(B, r\delta^{s+2}) \).

**Proof:** Let \( B' \) be any bag that contains \( h(x) \), let \( F' = \{ f'_0, \ldots, f'_{S-1} \} \in U \) be the approximate map into \( B' \). Because \( F' \) is plausible, \( f'_a(x) \) is defined for all \( 0 \leq a \leq s \) and \( h(x) \in f'_a(x) \). We consider the two cases one by one.

1. If \( x \in C(F, s) \) then Lemma 4.14 implies that \( f'_s(x) = f_s(x) \), so \( h(x) \in f_s(x) \).
2. If \( x \in P(F, s, T') \) then, by Lemma 4.14, \( T' \) is the subtree of \( T \setminus B \) that contains \( B' \). In particular, we have \( h(x) \in V(T') \). Since \( x \notin C(F, s) \), Lemma 4.14 implies that \( f'_s(x) \) does not intersect \( B(B, r\delta^{s+2}) \). Additionally, \( h(x) = f'_0(x) \in f'_s(x) \) because of the plausibility of \( F' \). Therefore, \( h(x) \notin B(B, r\delta^{s+2}) \).

\[ \square \]

**Relatively close pairs.** First, we bound the distortion between relatively short edges, for \( x, y \in X \) such that \( y \) is visible in the approximate map into a bag that contains \( h(x) \) or vice versa.

**Lemma 4.17.** Let \( x, y \in X \), let \( B \) be any bag that contains \( h(x) \), let \( F \in U \) be the map into \( B \), and let \( \rho = 1/(r-1) \). If \( y \in C(F, s) \), for any \( 0 \leq s < S \), then

\[
(1 - \rho) \leq \frac{d_G(h(x), h(y))}{d_X(x, y)} \leq (1 + \rho) \cdot \delta.
\]
Both last inequalities hold as \( v, h \) bounding the contraction. Next, we bound the contraction between all pairs of points in \( y \in X \). On the other hand, \( B \) be a bag that contains \( h \), and let \( r \) be the closest point of \( X \) such that \( x \leq d_X(x, y) \leq \delta^a \). Consequently, we have

\[
d_X(x, y) \leq \delta d_X(x, y) + \delta^a.
\]

Both last inequalities hold as \( v, h \) be a bag that contains \( h \), a bin of diameter at most \( \delta^a \). On the other hand, \( y \notin \text{Dom}(f_a-1) \), so \( d_G(h(x), h(y)) \geq r\delta^{a+1} \) (by Lemma 4.16 (2)). Therefore, \((r - 1)\delta^{a+1} \leq r \cdot \delta^{a+1} - \delta^a \leq d_G(h(x), h(y)) - d_G(h(y), v) \leq d_G(h(x), v) \leq \delta d_X(x, y)\).

Consequently,

\[
\delta^a \leq \frac{1}{r - 1} \cdot d_X(x, y) = \rho \cdot d_X(x, y).
\]

Substituting in (4) and (5), we obtain

\[
d_G(h(x), h(y)) \geq (1 - \rho) \cdot d_X(x, y),
\]

and

\[
d_G(h(x), h(y)) \leq (\delta + \rho) \cdot d_X(x, y) \leq (1 + \rho) \cdot \delta \cdot d_X(x, y),
\]

respectively.

Bounding the contraction. Next, we bound the contraction between all pairs of points in \( X \).

Lemma 4.18. Let \( x, y \in X \), and \( \rho = 1/(r - 1) \). We have,

\[
\frac{d_G(h(x), h(y))}{d_X(x, y)} \geq 1 - 3\rho
\]

Proof: For each \( 0 \leq s < S \), let \( z_s \) be the closest point of \( X_{>s} \) to \( y \), thus, \( d_X(z_s, y) \leq \delta^s \). Let \( B' \) be a bag that contains \( h(y) \), and let \( F' \in U \) be the map into \( B' \). The plausibility of \( F' \) implies that \( z_s \in C(F', s) \), therefore, by Lemma 4.17 we have

\[
(1 - \rho) \cdot d_X(y, z_s) \leq d_G(h(y), h(z_s)) \leq (1 + \rho) \cdot \delta \cdot d_X(y, z_s) \leq (1 + \rho) \cdot \delta^{s+1}.
\]

Let \( B \) be a bag that contains \( h(x) \), and let \( F = \{f_0, \ldots, f_{S-1}\} \in U \) be the plausible approximate
map into $B$. Let $a$ be the smallest scale such that (i) $z_a \in \text{Dom}(f_a)$, and (ii) the bin $f_a(z_a)$ intersects the ball $B(h(x),(1/\rho)\delta^{a+2})$. So, by Lemma 4.17,

$$(1 - \rho) \cdot d_X(x, z_a) \leq d_G(h(x), h(z_a))$$  

(7)

By the triangle inequality, the definition of $z_a$, (7), and (6) we have,

$$d_G(h(x), h(y)) \geq d_G(h(x), h(z_a)) - d_G(h(z_a), h(y)) \geq d_G(h(x), h(z_a)) - (1 + \rho)\delta^{a+1},$$

(8)

and

$$d_X(x, y) \leq d_X(x, z_a) + d_X(z_a, y) \leq d_X(x, z_a) + \delta^a \leq \frac{d_G(h(x), h(z_a))}{1 - \rho} + \delta^{a+1}.$$  

(9)

We consider two cases: (1) $d_X(h(x), h(z_a)) > (1/\rho)\delta^{a+1}$, and (2) $d_X(h(x), h(z_a)) \leq (1/\rho)\delta^{a+1}$.

**Case (1) $((1/\rho)\delta^{a+1} < d_X(h(z_a), h(x)))** From (8) and (9), we have:

$$\frac{d_G(h(x), h(y))}{d_X(x, y)} \geq (d_G(h(x), h(z_a)) - (1 + \rho)\delta^{a+1}) \left/ \left( \frac{d_G(h(x), h(z_a))}{1 - \rho} + \delta^{a+1} \right) \right..$$

Now let $\alpha = d_G(h(x), h(z_a))$ and consider

$$g(\alpha) = (\alpha - (1 + \rho)\delta^{a+1}) \left/ \left( \frac{\alpha}{1 - \rho} + \delta^{a+1} \right) \right..$$

as a function of $\alpha$ for $\alpha \geq (1/\rho)\delta^{a+1}$. The function $g(\alpha)$ is increasing for $\alpha \in ((1/\rho)\delta^{a+1}, \infty)$ because $\rho \leq 1/10$. So, it obtains its minimum at $\alpha = (1/\rho)\delta^{a+1}$. Therefore, we have:

$$g(\alpha) \geq \left( \frac{\delta^{a+1}}{\rho} - (1 + \rho)\delta^{a+1} \right) \left/ \left( \frac{1}{1 - \rho} \cdot \frac{\delta^{a+1}}{\rho} + \delta^{a+1} \right) \right..$$

$$= \frac{1 - 2\rho + \rho^3}{1 + \rho - \rho^2} \geq 1 - \frac{3\rho}{1 + \rho} \geq 1 - 3\rho$$

The last inequality holds because $\rho \geq 0$.

**Case (2) $d_X(h(z_a), h(x)) \leq (1/\rho)\delta^{a+1}$** Substituting in (9), we obtain,

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

(10)

Since $a$ is the smallest scale with properties (i) and (iii), either $z_{a-1} \notin \text{Dom}(f_{a-1})$ or $d_G(h(x), f_{a-1}(z_{a-1})) > (1/\rho)\delta^{a+1}$. In either case, Lemma 4.16 implies:

$$d_G(h(x), h(z_{a-1})) \geq \frac{1}{\rho} \cdot \delta^{a+1}.$$  

Additionally, from (6), we have

$$d_G(h(z_{a-1}), h(y)) \leq (1 + \rho) \cdot \delta \cdot d_X(z_{a-1}, y) \leq (1 + \rho)\delta \leq (1 + \rho)\delta^{a+1}.$$
Consequently, by the triangle inequality, we have:

\[
d_G(h(x), h(y)) \geq d_G(h(x), h(z_{a-1}) - d_G(h(z_{a-1}, h(y)))
\]

\[
\geq \left(\frac{1}{\rho} - 1 - \rho\right) \delta^{a+1}
\]

\[
= \frac{1 - \rho - \rho^2}{\rho} \cdot \delta^{a+1}.
\]

From (10) and (11) we conclude:

\[
\frac{d_G(h(x), h(y))}{d_X(x, y)} \geq \left(\frac{1 - \rho - \rho^2}{\rho}\right) \cdot \left(\frac{1 + \rho}{\rho(1 - \rho)}\right) = \frac{1 - 2\rho + \rho^3}{1 + \rho} \geq \frac{1 - 2\rho}{1 + \rho} = 1 - \frac{3\rho}{1 + \rho} \geq 1 - 3\rho.
\]

The last inequality holds because \(\rho \geq 0\).

\[
\square
\]

**Bounding the expansion.** Finally, we bound the expansion between all pairs of points in \(X\).

**Lemma 4.19.** Let \(x, y \in X\), and \(\rho = 1/(r - 1)\). We have,

\[
\frac{d_G(h(x), h(y))}{d_X(x, y)} \leq \delta \cdot (1 + 4\rho)
\]

**Proof:** For each \(0 \leq s < S\), let \(z_s\) be the closest point of \(X_{>s}\) to \(y\), thus, \(d_X(z_s, y) \leq \delta^s\). Let \(B'\) be a bag that contains \(h(y)\), and let \(F' \in \mathcal{U}\) be the map into \(B'\). The plausibility of \(F'\) implies that \(z_s \in C(F', s)\), therefore, by Lemma 4.17 we have

\[
(1 - \rho) \cdot d_X(y, z_s) \leq d_G(h(y), h(z_s)) \leq (1 + \rho) \cdot \delta \cdot d_X(y, z_s) \leq (1 + \rho) \cdot \delta^{s+1}. \tag{12}
\]

Let \(B\) be a bag that contains \(h(x)\), and let \(F = \{f_0, \ldots, f_{S-1}\} \in \mathcal{U}\) be the plausible approximate map into \(B\). Let \(a\) be the smallest scale such that (i) \(z_a \in \text{Dom}(f_a)\), and (ii) the bin \(f_a(z_a)\) intersects the ball \(B(h(x), (1/\rho)\delta^{a+2})\). So, by Lemma 4.17,

\[
d_G(h(x), h(z_a)) \leq (1 + \rho) \cdot \delta \cdot d_X(x, z_a). \tag{13}
\]

By the triangle inequality, the definition of \(z_a\), (12), and (13) we have,

\[
d_G(h(x), h(y)) \leq d_G(h(x), h(z_a)) + d_G(h(z_a), h(y)) \leq d_G(h(x), h(z_a)) + (1 + \rho)\delta^{a+1}, \tag{14}
\]

and

\[
d_X(x, y) \geq d_X(x, z_a) - d_X(z_a, y) \geq d_X(x, z_a) - \delta^a \geq \frac{d_G(h(x), h(z_a))}{(1 + \rho)\delta} - \delta^a. \tag{15}
\]

We consider two cases: (1) \(d_X(h(x), h(z_a)) > (1/\rho)\delta^{a+1}\), and (2) \(d_X(h(x), h(z_a)) \leq (1/\rho)\delta^{a+1}\).
Case (1) \(((1/\rho)^{\alpha+1} < d_X(h(z_a), h(x)))\) From (14) and (15), we have:

\[
\frac{d_G(h(x), h(y))}{d_X(x, y)} \leq \left( \frac{d_G(h(x), h(z_a)) + (1 + \rho)\delta^{a+1}}{1 + \rho} \right) \cdot \frac{\delta^{a+1}}{(1 + \rho)\delta - \delta^a}
\]

Now let \(\alpha = d_G(h(x), h(z_a))\) and consider

\[
g(\alpha) = \left( \frac{\alpha}{1 + \rho} \right) \cdot \frac{\alpha}{\delta} - \delta^a
\]

as a function of \(\alpha\) for \(\alpha \geq (1/\rho)\delta^{a+1}\). The function \(g(\alpha)\) is decreasing for \(\alpha \in ((1/\rho)\delta^{a+1}, \infty)\) because \(\rho \leq 1/10\). So, it obtains its maximum at \(\alpha = (1/\rho)\delta^{a+1}\). Therefore, we have:

\[
g(\alpha) \leq \left( \frac{\delta^{a+1}}{\rho} + (1 + \rho)\delta^{a+1} \right) \cdot \frac{\delta^{a+1}}{(1 + \rho)\delta - \delta^a} = \delta \cdot \left( 1 + \rho \cdot \frac{3 + 3\rho + \rho^2}{1 - \rho - \rho^2} \right) \leq \delta \cdot (1 + 4\rho)
\]

The last inequality holds because \(\rho \leq 1/10\).

Case (2) \(d_X(h(x), h(z_a)) \leq (1/\rho)\delta^{a+1}\) Substituting in (14), we obtain,

\[
d_G(h(x), h(y)) \leq \frac{1}{\rho} \cdot \delta^{a+1} + (1 + \rho)\delta^{a+1} = \delta^{a+1} \cdot \frac{1 + \rho + \rho^2}{\rho}.
\]

Since \(a\) is the smallest scale with properties (i) and (ii), either \(z_{a-1} \notin \text{Dom}(f_{a-1})\) or \(d_G(h(x), f_{a-1}(z_{a-1})) > (1/\rho)\delta^{a+1}\). In either case, the plausibility of \(F\) implies:

\[
d_X(x, z_{a-1}) \geq \frac{1}{\rho} \delta^{a}.
\]

On the other hand, by the triangle inequality, we have:

\[
d_X(x, y) \geq d_X(x, z_{a-1}) - d_X(z_{a-1}, y) \geq \frac{1}{\rho} \delta^{a} - \delta^{a-1} \geq \delta^{a} \cdot \frac{1 - \rho}{\rho}.
\]

From (16) and (17) we conclude:

\[
\frac{d_G(h(x), h(y))}{d_X(x, y)} \leq \delta \cdot \frac{1 + \rho + \rho^2}{1 - \rho} \leq \delta \cdot \left( 1 + \rho \cdot \frac{2 + \rho}{1 - \rho} \right) \leq \delta \cdot (1 + 3\rho).
\]

The last inequality holds because \(\rho \leq 1/10\).

Bounding the distortion. Now, we are ready to prove the main lemma of this subsection.

Proof (of Lemma 4.13): Lemma 4.15 shows that \(h\) is a valid embedding. Let \(\rho = 1/(r - 1)\). Lemma 4.18 and Lemma 4.19 show that the contraction and the expansion of \(h\) are at most \((1-3\rho)^{-1}\) and \((1+4\rho)\delta\), respectively. Therefore, as \(r \geq 20\ (\rho \leq 1/10)\), the distortion of \(h\) is at most

\[
\frac{1 + 4\rho}{1 - 3\rho} \cdot \delta \leq (1 + 10\rho) \cdot \delta \leq \frac{10}{r - 1} \cdot \delta \leq \frac{20}{r} \cdot \delta.
\]
4.6 The Algorithm, Proof of Theorem 1.1

We describe a dynamic programming procedure to compute a consistent set of approximate maps into \( \mathcal{T} \). To that end, we need some definitions.

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 \( F \) be a plausible approximate map into \( \mathcal{B} \). Let \( \mathcal{T}_F \) denote the subtree of \( \mathcal{T} \) with root \( \mathcal{B} \). Let \( \mathcal{U}_F \) be a set of plausible approximate maps into the bags of \( \mathcal{T}_F \), which contains exactly one approximate map into each bag of \( \mathcal{T}_F \). We say that \( \mathcal{U}_F \) is a consistent set of approximate maps into \( \mathcal{T}_F \) if for each \( F, F' \in \mathcal{U}_F \) we have: if \( F \) and \( F' \) are into adjacent bags then they are consistent. Finally, we say that an approximate map \( F \) into \( \mathcal{B} \) is compatible (with \( \mathcal{T}_F \)) if there is a consistent set of approximate maps into \( \mathcal{T}_F \) that contains \( F \).

The goal of our dynamic programming procedure is to find a compatible map into \( \mathcal{R} \). To achieve that our algorithm will end up computing a consistent set of approximate maps into \( \mathcal{T} \).

Lemma 4.20. Let \( \epsilon > 0 \) and \( \delta > 1 \). There is a \( \Delta^{\omega \cdot \lambda \cdot (O(\delta^2/\epsilon))^{\lambda \cdot \log_3 n} \cdot n^O(\omega) \cdot m^{O(1)} \) time algorithm that computes a \((1 + \epsilon)\delta\) embedding of \( X \) into \( V_G \) if \( \delta \geq \delta_{opt} \). 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 first computes a tree decomposition \( \mathcal{T} \) of \( G \) of width \( O(\omega) \) via the algorithm of Lemma 3.1 in \( m \cdot 2^{O(\omega)} \) time.

Next, it computes a consistent set of approximate maps into \( \mathcal{T} \) using a dynamic programming procedure. Starting from leaves, our algorithm computes compatible maps into the bags of \( \mathcal{T} \) iteratively. A compatible map into \( \mathcal{R} \) implies a set of consistent maps into \( \mathcal{T} \).

For a leaf, each plausible approximate map is also compatible. Therefore, all compatible maps into a leaf can be listed in \( \Delta^{\omega \cdot \lambda \cdot (O(\delta^2))^{\lambda \cdot \log_3 n} \cdot n^O(\omega) \cdot m^{O(1)} \) time by Lemma 4.10.

Now, let \( \mathcal{B} \) be a non-leaf bag and suppose that the lists of compatible maps into all children of \( \mathcal{B} \) are computed. To list all compatible maps into \( \mathcal{B} \), our algorithm first computes a list \( L \) of all plausible approximate maps into \( \mathcal{B} \). A plausible approximate map \( F \) into \( \mathcal{B} \) is compatible if and only if for every child \( \mathcal{B}' \) of \( \mathcal{B} \) there is a compatible map into \( \mathcal{B}' \) that is consistent with \( F \). Using this characterization and Lemma 4.10 and Lemma 4.11, our algorithm extracts the set of compatible maps into \( \mathcal{B} \) from \( L \). The algorithm computes compatible maps into all bags of \( \mathcal{T} \), iteratively. Overall, a consistent set \( \mathcal{U} \) of approximate maps into \( \mathcal{T} \) can be computed in \( \Delta^{\omega \cdot \lambda \cdot (O(\delta^2))^{\lambda \cdot \log_3 n} \cdot n^O(\omega) \cdot m^{O(1)} \) time (the running time is dominated by the cost of computing plausible maps).

From \( \mathcal{U} \), our algorithm computes the embedding \( h \), as described in (3). Lemma 4.13, ensures that the distortion of \( h \) is \((1 + 20/r)\delta \). Therefore, by setting \( r = 20/\epsilon \), we obtain an embedding of distortion at most \((1 + \epsilon)\delta \) in the following running time.

\[
\Delta^{\omega \cdot \lambda \cdot (O(\delta^2/\epsilon))^{\lambda \cdot \log_3 n} \cdot n^O(\omega) \cdot m^{O(1)} .
\]

Proof (of Theorem 1.1): By Lemma 3.3, there is a list \( L \) of \((nm)^{O(1)} \) numbers that contain \( \delta_{opt} \) and that can be computed in \((nm)^{O(1)} \) time. For each \( \beta \in L \), in increasing order, we call the algorithm of Lemma 4.20, with parameters \( \delta = \beta + \epsilon/2 \) and \( \epsilon = \epsilon/3 \). As \( \beta \geq 1 \) and \( \epsilon > 0 \), we have \( \delta > 1 \) and \( \epsilon > 0 \). This algorithm outputs an embedding with distortion

\[
(1 + \epsilon/3)(\beta + \epsilon/2) \leq (1 + \epsilon)\beta,
\]
if $\beta \geq \delta_{\text{opt}}$, otherwise, it either outputs an embedding of distortion $(1 + \varepsilon)\beta$, or it decides that $\beta < \delta_{\text{opt}}$. We stop, and return the embedding, as soon as we obtain one. The running time of each call to the algorithm of Lemma 4.20 is bounded by

$$\Delta \omega \cdot \lambda \cdot (O(\beta^2/\varepsilon))^{1/2} \cdot n^{O(\omega)} \cdot m^{O(1)} = \Delta \omega \cdot \lambda \cdot (O(\beta^2/\varepsilon))^{1/2} \cdot n^{O(\omega)} \cdot m^{O(1)}. \quad (18)$$

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 (18), we conclude that the running time of one call to the algorithm of Lemma 4.20 is bounded by

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

As we call this algorithm for different values of $\beta \in L$, and $\beta \leq \delta_{\text{opt}}$, the total running time is bounded by

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

□

References


