Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Computer and System Sciences 69 (2004) 485–497
JOURNAL OF COMPUTER AND SYSTEM SCIENCES
http://www.elsevier.com/locate/jcss

A tight bound on approximating arbitrary metrics by tree metrics

Jittat Fakcharoenphol,[a,1] Satish Rao,[b,2] and Kunal Talwar[c,*,3]

[a] *Kasetsart University, Bangkok, Thailand*
[b] *Computer Science Division, University of California, Berkeley, CA 94720, USA*
[c] *Computer Science Division, University of California, Berkeley, CA 94720, USA*

Received 3 September 2003; revised 12 April 2004

Available online 24 June 2004

Abstract

In this paper, we show that any n point metric space can be embedded into a distribution over dominating tree metrics such that the expected stretch of any edge is $O(\log n)$. This improves upon the result of Bartal who gave a bound of $O(\log n \log \log n)$. Moreover, our result is existentially tight; there exist metric spaces where any tree embedding must have distortion $\Omega(\log n)$-distortion. This problem lies at the heart of numerous approximation and online algorithms including ones for group Steiner tree, metric labeling, buy-at-bulk network design and metrical task system. Our result improves the performance guarantees for all of these problems.

Keywords: Metrics; Embeddings; Tree Metrics

1. Introduction

1.1. Metric approximations

The problem of approximating a given graph metric by a "simpler" metric has been a subject of extensive research, motivated from several different perspectives. A particularly simple metric of

*Corresponding author.
E-mail address: kunal@cs.berkeley.edu (K. Talwar).
[1] Supported in part by a DPST scholarship and NSF Grant CCR-0105533. Work done while the author was at University of California, Berkeley.
[2] Supported by NSF Grant CCR-0105533.
[3] Supported by NSF Grants CCR-0105533 and CCR-9820897.

0022-0000/$ - see front matter
doi:10.1016/j.jcss.2004.04.011

choice, also favored from the algorithmic point of view, is a tree metric, i.e. a metric arising from shortest path distance on a tree containing the given points. Ideally we would like that distances in the tree metric are no smaller than those in the original metric and we would like to bound the *distortion* or the maximum increase. However, there are simple graphs (e.g. the n-cycle) for which the distortion must be $\Omega(n)$ [45,7,26].

To circumvent this, Karp [33] considered approximating the cycle by a probability distribution over paths, and showed a simple distribution such that the expected length of each edge is no more than twice its original length. This gave a competitive ratio of 2 for the k-server problem (on a cycle) that had motivated this approach. Alon et al. [1] looked at approximating arbitrary graph metrics by (a distribution over) spanning trees, and showed an upper bound of $2^{O(\sqrt{\log n \log \log n})}$ on the distortion.

Bartal [7] formally defined probabilistic embeddings and improved on the previous result by showing how to probabilistically approximate metrics by tree metrics with distortion $O(\log^2 n)$. Unlike the result of Alon et al. [1], Bartal's trees were not spanning trees of the original graph. He showed that this probabilistic approximation leads to approximation algorithms for several problems, as well as the first polylogarithmic competitive ratios for a number of on-line problems. We should note that the trees that Bartal used have a special structure which he termed hierarchically well separated. This meant that weights on successive levels of the tree differed by a constant factor. This was important for several of his applications.

Konjevod et al. [37] showed how Bartal's result improves to $O(\log n)$ for planar graphs, and Charikar et al. [17] showed similar bounds for low-dimensional normed spaces. Inspired by ideas from Seymour's work on feedback arc set [49], Bartal [8] improved his earlier result to $O(\log n \log \log n)$. This of course led to improved bounds on the performance ratios of several applications. Bartal also observed that any probabilistic embedding of an expander graph into a tree has distortion at least $\Omega(\log n)$.

In this paper, we show that an arbitrary metic space can be approximated by a distribution over dominating tree metrics with distortion $O(\log n)$, thus closing the gap between the lower and the upper bounds. Our result is constructive and we give a simple algorithm to sample a tree from this distribution. Our trees are also heirarchically well separated, like Bartal's. This gives improved approximation algorithms for various problems including group Steiner tree [24], metric labeling [19,35], buy-at-bulk network design [4], and vehicle routing [16]. We give a more comprehensive list in Section 4.

Moreover, our techniques improve the spreading metrics-based divide-and-conquer algorithms of Even et al. [20] from $O(\log n \log \log n)$ to $O(\log n)$.

1.2. Related work

Divide-and-conquer methods have been used to provide polylogarithmic-factor approximation algorithms for numerous graph problems since the discovery of an $O(\log n)$ approximation algorithm for finding a graph separator [40]. The algorithms proceeded by recursively dividing a problem using the above-approximation algorithm, and then using the decomposition to find a solution. Typically, the approximation factor was $O(\log^2 n)$: a logarithmic factor came from the $O(\log n)$ separator approximation, another $O(\log n)$ factor came from the recursion. Using this framework, polynomial-time approximation algorithms for many problems are given in [40], for example: crossing number, VLSI layout, minimum feedback arc set, and search number.

Independently, Seymour [49] gave an $O(\log n \log\log n)$ bound on the integrality gap for a linear programming relaxation of the feedback arc set problem (for which the above techniques had given an $O(\log^2 n)$ bound). In doing so, he developed a technique that balanced the approximation factor of his separator-based procedure against the cost of the recursion to significantly improve the bounds.

Even et al. [20] introduced linear programming relaxations for a number of other problems and combined them with the afore-mentioned techniques of Seymour to give $O(\log n \log\log n)$-approximation algorithms for many of the problems that previously had $O(\log^2 n)$ approximation algorithms, e.g., linear arrangement, embedding a graph in d-dimensional mesh, interval graph completion, minimizing storage-time product, and (subset) feedback sets in directed graphs.

Bartal's results [8] implied $O(\log n \log\log n)$-approximations for still more problems. Moreover, he used probabilistic techniques to bound the expected stretch of each edge, not just the average. This led to polylogarithmic competitive ratio algorithms for a number of online problems (against oblivious adversaries) such as metrical task system [10]. Charikar et al. [16,17] showed how to derandomize the approximation algorithms that follow from Bartal's embeddings. Moreover, they explicitly showed a correspondence between probabilistic embedding and hierarchical decomposition.

This work also follows the line of research on embeddings, with low distortion, graphs into other "nice" metric spaces which have good structural properties such as Euclidean and ℓ_1 spaces [41,27,18,47,23].

The work of Bourgain [14] showed that any finite metric on n nodes can be embedded into ℓ_2 with logarithmic distortion with the number of dimensions exponential in n. Linial et al. [41] modified Bourgain's result to apply for ℓ_1 metrics and to use $O(\log^2 n)$ dimensions. Aumann and Rabani [3] and Linial et al. [41] gave several applications, including a proof of a logarithmic bound on max-flow min-cut gap for multicommodity flow problems. They also gave a lower bound on the distortion of any embeddings of general graphs into ℓ_1. For more details, we point the reader to the survey by Matousek [42] and the more recent survey by Indyk and Matousek [31].

Embeddings of special graphs have also been considered by many researchers. Gupta et al. [27] considered embeddings or series-parallel graphs and outerplanar graphs into ℓ_1 with constant distortion; Chekuri et al. [18] show a constant-distortion embedding for k-outerplanar graphs. For planar graphs, Rao [47] gave an $O(\sqrt{\log n})$-distortion embedding into ℓ_2, which matched the lower bound given by Newman and Rabinovich [43]. Related questions have been addressed by Krauthgamer et al. [39].

Graph decomposition techniques for many interesting classes of graphs have also been extensively studied. For example, Klein et al. [34] result provided a constant factor approximation for graphs that exclude fixed sized minors (which includes planar graphs). Similar results were given by Charikar et al. [17] for geometric graphs.

1.3. Our techniques

The algorithm relies on techniques from the algorithm for 0-extension given by Calinescu et al. [15], and improved by Fakcharoenphol et al. [21]. The CKR procedure implies a randomized algorithm that outputs clusters of diameter about Δ such that the probability of an edge e being cut is $\frac{d_e}{\Delta}\log n$, where d_e is the length of the edge e. The analysis can in fact be improved to replace the $\log n$ by the logarithm of the ratio of number of vertices within distance Δ of e to the number of vertices within distance $\Delta/2$; i.e. the number of times the size of a neighborhood of e doubles between $\Delta/2$ and Δ. Our algorithm runs a CKR-like procedure for diameters 2^i, $i = 0, 1, 2, \ldots$ to get a hierarchical decomposition of the graph (which can then be converted to a tree). Since, the total number of doublings over all these levels is bounded by $\log n$, we get an upper bound of $O(\log n)$ on the distortion.

2. The algorithm

In this section, we outline the algorithm for probabilistically embedding an n point metric into a tree, and show that the expected distortion of any distance is $O(\log n)$. Like previous algorithms, we first decompose the graph hierarchically and then convert the resulting laminar family to a tree.

2.1. Preliminaries

We define some notation first. Let the input metric be (V, d). We shall refer to the elements of V as vertices or points. We shall refer to a pair of vertices (u, v) as an edge. Without loss of generality, the smallest distance is strictly more than 1. Let Δ denote the diameter of the metric (V, d). Without loss of generality, $\Delta = 2^\delta$.

A metric (V', d') is said to *dominate* (V, d) if for all $u, v \in V$, $d'(u, v) \geqslant d(u, v)$. We shall be looking for tree metrics that dominate the given metric. Let $\mathcal{S}$ be a family of metrics over V, and let $\mathcal{D}$ be a distribution over $\mathcal{S}$. We say that $(\mathcal{S}, \mathcal{D})$ *α-probabilistically approximates* a metric (V, d) if every metric in $\mathcal{S}$ dominates d and for every pair of vertices $(u, v) \in V$, $E_{d' \in (\mathcal{S}, \mathcal{D})}[d'(u, v)] \leqslant \alpha \cdot d(u, v)$. We shall be interested in α-probabilistically approximating an arbitrary metric (V, d) by a distribution over tree metrics.

For a parameter r, an *r-cut decomposition* of (V, d) is a partitioning of V into clusters, each centered around a vertex and having radius at most r. Thus, each cluster has diameter at most $2r$. A *hierarchical cut decomposition* of (V, d) is a sequence of $\delta + 1$ nested cut decompositions $D_0, D_1, \ldots, D_\delta$ such that

- $D_\delta = \{V\}$, i.e. the trivial partition (that puts all vertices in a single cluster).
- D_i is a 2^i-cut decomposition, and a refinement of D_{i+1} (i.e., each cluster in D_i is contained within some cluster in D_{i+1}).

Note that each cluster in D_0 has radius at most 1 and hence must be a singleton vertex.

2.2. Decompositions to trees

A hierarchical cut decomposition defines a laminar family,[4] and naturally corresponds to a rooted tree as follows. Each set in the laminar family corresponds to a node in the tree and the

[4] Recall that a *laminar family* $\mathcal{F} \subseteq 2^V$ is a family of subsets of V, such that for any $A, B \in \mathcal{F}$, it is the case that $A \subseteq B$ or $B \subseteq A$ or $A \cap B = \phi$.

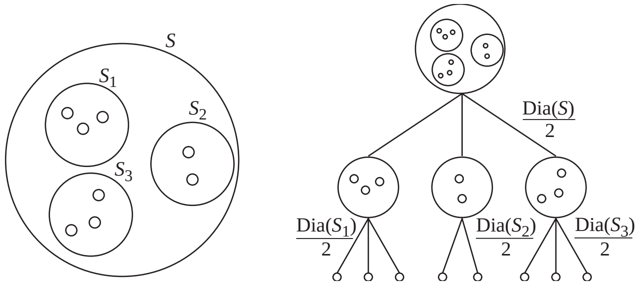

Fig. 1. Converting a laminar family into a tree. The lengths we put on the links ensure that the embedding is an expansion.

children of a node corresponding to a set S are the nodes corresponding to maximal subsets of S in the family. Thus the node corresponding to V is the root and the singletons are the leaves. Also note that the children of a set in D_{i+1} are sets in D_i (see Fig. 1).

We define a distance function on this tree as follows. The links from a node S in D_i to each of its children in the tree have length equal to the 2^i (which is an upper bound on the radius of S). This induces a distance function $d^T(\cdot,\cdot)$ on V where $d^T(u,v)$ is equal to the length of the shortest path distance in T from node $\{u\}$ to node $\{v\}$. Since each cluster in D_i has diameter at most 2^{i+1}, any pair of vertices (u,v) must be separated in the partition D_i, when $i = (\lfloor \log_2 d(u,v) \rfloor - 1)$. Thus, $d^T(u,v)$ is at least $2 \times 2^i \geqslant d(u,v)$ and so d^T dominates d.

We shall also like to place upper bounds on $d^T(u,v)$. We say an edge (u,v) is at level i if u and v are first separated in the decomposition D_i. Note that if (u,v) is at level i, then $d^T(u,v) = 2\sum_{j=0}^{i} 2^j \leqslant 2^{i+2}$.

2.3. Decomposing the graph

We shall describe a random process to define a hierarchical cut decomposition of (V,d), such that the probability that an edge (u,v) is at level i decreases geometrically with i.

We first pick a random permutation π of $\{v_1, v_2, \ldots, v_n\}$, which will be used throughout the process. We also pick a β randomly in the interval $[1,2]$ from the distribution given by the probability density function $p(x) = \frac{1}{x \ln 2}$. We start with the trivial partition $D_\delta = \{V\}$. For each i, we compute D_i from D_{i+1} as follows. First, set β_i to be $2^{i-1}\beta$. Let S be a cluster in D_{i+1}. We assign a vertex $u \in S$ to the first (according to π) vertex $v \in V$ within distance β_i of u. Each child cluster of S in D_i then consists of the set of vertices in S assigned to a single center v. Note that the center v itself need not be in S. Thus one center v may correspond to more than one cluster, each inside a different level $(i+1)$ cluster (see for example, the center $\pi(8)$ in Fig. 2). Note that since $\beta_i \leqslant 2^i$, the radius of each cluster is at most 2^i and thus we indeed get a 2^i-cut decomposition. More formally,

Algorithm.*Partition(V,d)*
1. Choose a random permutation π of $v_1, v_2, \ldots, v_n$.
2. Choose β in $[1,2]$ randomly from the distribution $p(x) = \frac{1}{x \ln 2}$.

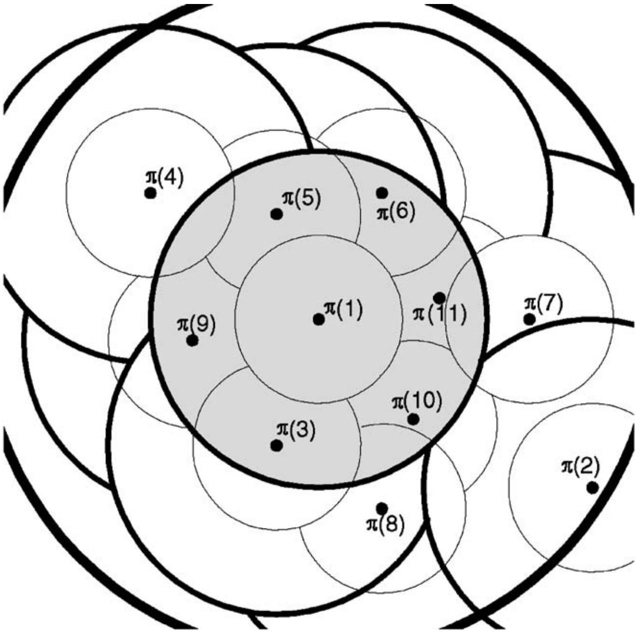

Fig. 2. A possible hierarchical cut decomposition output by the algorithm. The varying thicknesses indicate cuts at different levels.

3. $D_\delta \leftarrow V; i \leftarrow \delta - 1$.
4. while D_{i+1} has non-singleton clusters do
4.1 $\beta_i \leftarrow 2^{i-1}\beta$.
4.2 For $l = 1, 2, \ldots, n$ do
4.2.1 For every cluster S in D_{i+1}.
4.2.1.1 Create a new cluster consisting of all unassigned vertices in S closer than β_i to $\pi(l)$.
4.3 $i \leftarrow i - 1$.

It is easy to see that the algorithm can be implemented in time $O(n^3)$. A more careful implementation can actually be made to run in time $O(n^2)$ (i.e. linear in the size of the input).

First, note that our choice of the distribution implies that

Observation 1. *For any* $x \geqslant 1$, Pr[*some* β_i *lies in* $[x, x + \mathrm{d}x)] = \frac{1}{x \ln 2} \mathrm{d}x$.

We now fix an arbitrary edge (u, v), and show that the expected value of $d^T(u, v)$ is bounded by $O(\log n) \cdot d(u, v)$. We make no attempts to optimize the constants in this analysis.

Consider the clustering step at some level i. In each iteration, all unassigned vertices v such that $d(v, \pi(l)) \leqslant \beta_i$ assign themselves to $\pi(l)$. For some initial iterations of this procedure, both u and v remain unassigned. Then at some step l, at least one of u and v gets assigned to the center $\pi(l)$. We say that center w *settles* the edge (u, v) at level i if it is the first center to which at least one of u and

v get assigned. Note that exactly one center settles any edge (u, v) at any particular level. Further, we say that center w *cuts* the edge $e = (u, v)$ at level i if it settles e at this level, but exactly one of u and v is assigned to w at level i. Whenever w cuts edge (u, v) at level i, the tree length of the edge (u, v) is about 2^{i+2}. We blame this length to vertex w and define $d_w^T(u, v)$ to be $\sum_i \mathbf{1}(w \text{ cuts } (u, v) \text{ at level } i) \cdot 2^{i+2}$, where $\mathbf{1}(\cdot)$ denotes an indicator function. Clearly, $d^T(u, v) \leqslant \sum_w d_w^T(u, v)$.

We now arrange the vertices in V in order of increasing distance from the edge (u, v) (breaking ties arbitrarily). Consider the sth vertex w_s in this sequence. We now upper bound the expected value of $d_{w_s}^T(u, v)$ for an arbitrary w_s.

Without loss of generality, assume $d(w_s, u) \leqslant d(w_s, v)$. For a center w_s to cut (u, v), it must be the case that (see Fig. 3)

(a) $d(w_s, u) \leqslant \beta_i < \mathrm{d}(w_s, v)$ for some i.
(b) w_s settles e at level i.

Moreover, the contribution to $d_{w_s}^T(u, v)$ when this happens is at most $2^{i+2} \leqslant 8\beta_i$. Now consider a particular $x \in [d(w_s, u), d(w_s, v))$. The probability that some β_i falls in $[x, x + \mathrm{d}x)$, from observation 1, is at most $\frac{1}{x \ln 2} \cdot \mathrm{d}x$. Conditioned on β_i taking this value x, any of $w_1, w_2, \ldots, w_s$ can settle (u, v) at level i. The first one amongst these in the permutation π will then settle (u, v), and thus the probability of the event (b), conditioned on (a), is at most $\frac{1}{s}$. Thus, the expected cost of $d_{w_s}^T(u, v)$ is

$$\begin{aligned}\mathbf{E}[d_{w_s}^T(u, v)] &\leqslant \int_{d(w_s,u)}^{d(w_s,v)} \frac{1}{x \ln 2} \cdot 8x \cdot \frac{1}{s} \mathrm{d}x \\ &= \frac{8}{s \ln 2}(d(w_s, v) - d(w_s, u)) \\ &\leqslant \frac{8d(u, v)}{s \ln 2},\end{aligned}$$

where the last step follows by triangle inequality.

Using linearity of expectation, we get $\mathbf{E}[d^T(u, v)] \leqslant \sum_s \frac{8d(u,v)}{s \ln 2} = \frac{8d(u,v)}{\ln 2} \cdot H_n \leqslant \frac{8 \ln n}{\ln 2} \cdot d(u, v) = 8 \log n \cdot d(u, v)$.

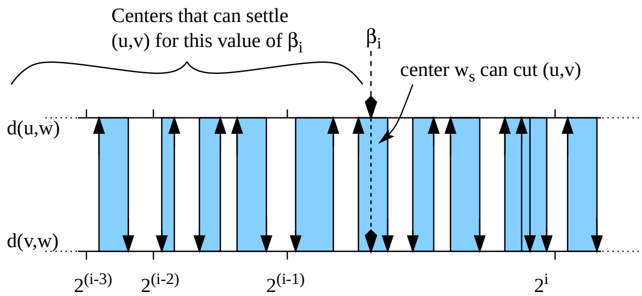

Fig. 3. Bounding the probability of an edge being cut. Each shaded rectangle represents a center; arrow marks indicate distances from u and v. Width of each shaded rectangle is at most $d(u, v)$.

Thus, we have shown that for any edge (u, v), the expected value of $d^T(u, v)$ is $O(\log n) \cdot d(u, v)$. Hence

Theorem 2. *The distribution over tree metrics resulting from our algorithm $O(\log n)$-probabilistically approximates the metric d.*

2.4. HSTs

A tree T is said to be *k-hierarchically well separated* if on any root to leaf path the edge lengths decrease by a factor of k in each step. Bartal [7,8] constructed distributions over trees which were hierarchically well separated, and such trees are more conducive to design of divide-and-conquer-type algorithms. The fact that the trees are well separated has been used in applications such as metrical task system [10] and metric labeling [35]. We note that the trees we construct are 2-HSTs. Bartal [8] also observed that a 2-HST can be converted to a k-HST with distortion $O(k)$, later improved to $O(k/\log k)$ [11]. This combined with our result implies a probabilistic embedding into k-HSTs with distortion $O(k \log n/\log k)$. In fact, a slight modification of our technique (details omitted) can be used to directly get k-HSTs for any k, with distortion $O(\frac{k \log n}{\log k})$.

3. Derandomization

The problem of probabilistic approximation by tree metrics asks for a distribution over tree metrics such that the expected stretch of each edge is small. A dual problem is to find a single tree such that the (weighted) average stretch of the edges is small. More precisely, given weights c_{uv} on edges, find a tree metric d^T such that,

- $\forall u, v \in V, d^T(u, v) \geqslant d(u, v)$.
- $\sum_{u,v \in V} c_{uv} \cdot d^T(u, v) \leqslant \alpha \sum_{u,v \in V} c_{uv} \cdot d(u, v)$.

Charikar et al. [17] showed that solving this problem is enough for most applications, and moreover leads to a probabilistic embedding into a distribution over at most $O(n \log n)$ tree metrics. This also implies deterministic algorithms for applications. The algorithm of the previous section can easily be derandomized by the method of pessimistic estimators [46] along the lines of [15,21], leading to an algorithm for the above problem with $\alpha = O(\log n)$.

Here, we give another deterministic algorithm that uses a modified region growing technique along the lines of [40,25]. We first give the modified region growing algorithm, and then outline how to use it to solve the above problem.

3.1. Region growing lemma

Without loss of generality, we assume that the smallest edge length is at least 1 and the smallest non-zero edge weight c_e is at least 1. Given an edge $e = (u, v)$ of length d_e and weight c_e, we define the *volume* of the edge as $c_e \cdot d_e$. Let $W = \sum_e c_e d_e$. We assume that W is polynomial in n (so that

$\log W = O(\log n)$); we shall show how to relax this at the end of this section. We imagine placing vertices arbitrary close to each other along the edges, and call them volume elements. We consider neighborhoods $B(t,r)$ around volume elements, and define the volume $W(t,r)$ of this neighborhood as follows. An edge $e=(u,v)$ with both end points in $B(t,r)$ contributes $c_e \cdot d_e$ to $W(t,r)$. An edge $e=(u,v)$ with exactly one end point, say u, in $B(t,r)$ contributes $c_e \cdot (r - d(t,u))$ to $W(t,r)$. Let $c(t,r)$ be the total weight of edges cut by $B(t,r)$, i.e. $c(t,r) = \sum_{u \in B(t,r), v \notin B(t,r)} c_{uv}$. From the definitions, it follows that for all r,

$$\frac{\mathrm{d}W(t,r)}{\mathrm{d}r} = c(t,r). \tag{1}$$

The region growing argument of Garg et al. [25] (also implicit in Leighton and Rao [40]) shows that given any $\delta > 0$ and t, there is an $r \leqslant \delta$, such that $\frac{c(t,r)}{W(t,r)} \cdot \delta \leqslant O(\log n)$.

We now show how to find several cuts of geometrically increasing radii, with total overhead at most $O(\log n)$. More precisely, we show that there are radii $r_i : 2^{i-1} \leqslant r_i < 2^i$ such that $\sum_i \frac{c(t,r_i)}{W(t,r_i)} \cdot 2^i \leqslant O(\log n)$.

Let $W_i = W(t, 2^i)$. We grow regions around t. We claim that there exists an $r_i : 2^{i-1} \leqslant r_i < 2^i$ such that $\frac{c(t,r_i)}{W(t,r_i)} \leqslant 2^{-(i-1)} \ln \frac{W_i}{W_{i-1}}$. Suppose not. Then $c(t,r) > \gamma_i \cdot 2^{-(i-1)} W(t,r)$ for all $2^{i-1} \leqslant r < 2^i$, where $\gamma_i = \ln \frac{W_i}{W_{i-1}}$. Then, using (1), we get

$$\begin{aligned}
\ln \frac{W(t,2^i)}{W(t,2^{i-1})} &= \int_{2^{i-1}}^{2^i} \frac{1}{W(t,r)} \frac{\mathrm{d}W(t,r)}{\mathrm{d}r} \mathrm{d}r \\
&= \int_{2^{i-1}}^{2^i} \frac{c(t,r)}{W(t,r)} \mathrm{d}r \\
&> \int_{2^{i-1}}^{2^i} \gamma_i \cdot 2^{-(i-1)} \, \mathrm{d}r \\
&= \gamma_i,
\end{aligned}$$

which contradicts the definition of γ_i. Hence, we can find an $r_i : 2^{i-1} \leqslant r_i < 2^i$ such that $\frac{c(t,r_i)}{W(t,r_i)} \cdot 2^{i-1} \leqslant \gamma_i$.

Adding such equations over all i, we get $\sum_i \frac{c(t,r_i)}{W(t,r_i)} \cdot 2^i \leqslant 2 \ln W_\infty / W_0$. Noting that W_0 is at least 1, and that W_∞ is at most W, we get the desired bound.

We now sketch how to relax the assumption on W. Using standard techniques, we can assume that the largest edge length is at most $O(n)$. We smear an additional volume of W uniformly over each unit of length (irrespective of the edge weight), i.e. we set $w'_e = w_e + \frac{W}{\sum_e l_e}$ for each edge e. This increases the volume of the system by a factor of two. Moreover, this ensures that for every volume element t (assuming without loss of generality that the graph is connected), $W_0 = W(t,1)$ is at least W/n^2. Thus the ratio W_∞ / W_0 is at most $O(n^2)$, and result follows.

3.2. Algorithm

In this section, we shall use the region growing technique of the previous section to construct a cut decomposition of G. Let $\Delta = 2^{i+2}$ be an upper bound on the diameter of G.

The algorithm is as follows. Let t be the volume element that maximizes $W(t, 2^i)$. We cut out $B(t, r_i)$ (where r_i is defined with respect to t), and recurse on the two subpieces. We shall get the tree from the resulting laminar family as before; thus each edge in this cut has a tree length roughly 2^{i+2}. We charge the cost of this cut, i.e. $c(t, r_i) \cdot 2^{i+2}$ to the volume in $B(t, r_i)$. Thus a unit of volume t' in $B(t, r_i)$ gets charged $\frac{c(t,r_i)}{W(t,r_i)} \cdot 2^{i+2} \leqslant 4 \ln \frac{W}{W(t,2^i)} \leqslant 4 \ln \frac{W(t',2^{i+2})}{W(t',2^i)}$ (by choice of t). Moreover, note that t' now lies in a cluster of diameter at most 2^{i+1}; thus we have made progress. The total charge to t' can thus be bounded by $\sum_i 4 \ln \frac{W(t',2^{i+2})}{W(t',2^i)} \leqslant 8 \ln \frac{W}{W(t',1)} = O(\log n)$. Thus

Theorem 3. *The algorithm described above returns a* 2*-hierarchically well separated tree metric* d_T *such that*

- $\forall u, v \in V, d_T(u, v) \geqslant d(u, v)$
- $\sum_{u,v} c_{uv} d_T(u, v) \leqslant O(\log n) \sum_{u,v} c_{uv} d(u, v)$

4. Applications

Many problems are easy on trees. The partitioning algorithm we give produces a tree such that the expected stretch of each edge is at most $O(\log n)$. By using our result, the approximation ratios of various problems can be improved. The following is a list of some of our favorite applications.

The metric labeling problem: The previous result of Kleinberg and Tardos [35] gives an $O(\log k \log \log k)$-approximation algorithm based on a constant factor approximation for the case that the terminal metric is an HST. Our result improves this to $O(\log k)$.

We also note that Archer et al. [2] show that the earthmover linear program of Chekuri et al. [19] is integral when the input graph is a tree. Using this result, the approximation ratio can be improved to $O(\min(\log k, \log n))$.

Buy-at-bulk network design: Awerbuch and Azar [4] give a $O(1)$-approximation algorithm on trees. Thus, we can get an $O(\log n)$-approximation algorithm.

Minimum cost communication network problem: This problem [30,44,50] is essentially the dual problem defined in Section 3 and hence we get an $O(\log n)$ approximation.

The group Steiner tree problem: Garg et al. [24] give an $O(\log k \log n)$-approximation algorithm for trees; thus we get an $O(\log^2 n \log k)$-approximation algorithm, improving on the $O(\lambda \log n \log k)$ result by Bartal and Mendel [12], where $\lambda = O(\min\{\log n \log \log \log n, \log \Delta \log \log \Delta\})$.

Metrical Task system: Improving on the result of Bartal et al. [10], Fiat and Mendel [22] gave an $O(\log n \log \log n)$-competitive algorithms on HSTs. Bartal and Mendel's [12] multiembedding result thus gives an $O(\lambda \log n \log \log n)$-competitive ratio, where λ is as defined above. Our result improves this to an $O(\log^2 n \log \log n)$-competitive ratio against oblivious adversaries.

The result also improves the performance guarantees of several other problems such as vehicle routing [16], min sum clustering [11,9], covering steiner tree [36,28], hierarchical placement [38], topology aggregation [6,48], mirror placement [32], distributed K-server [13], distributed queueing [29] and mobile user [5]. We refer the reader to [8] and [17] for more detailed descriptions of these problems.

Acknowledgments

We thank Yair Bartal for helpful comments on a previous draft of the paper, and for pointing us to several of the aforementioned applications.

References

[1] N. Alon, R.M. Karp, D. Peleg, D. West, A graph–theoretic game and its application to the k-server problem, SIAM J. Comput. 24 (1) (1995) 78–100.
[2] A. Archer, J. Fakcharoenphol, C. Harrelson, R. Krauthgamer, K. Talwar, E. Tardos, Approximate classification via earthmover metrics, in: Proceedings of the 15th Annual ACM–SIAM Symposium on Discrete Algorithm, 2004.
[3] Y. Aumann, Y. Rabani, An $O(\log k)$ approximate min-cut max-flow theorem and approximation algorithm, SIAM J. Comput. 27 (1) (1998) 291–301.
[4] B. Awerbuch, Y. Azar, Buy-at-bulk network design, in: 38th Annual Symposium on Foundations of Computer Science, Miami Beach, FL, 20–22 October 1997, pp. 542–547.
[5] B. Awerbuch, D. Peleg, Concurrent online tracking of mobile users, in: SIGCOMM, 1991, pp. 221–233.
[6] B. Awerbuch, Y. Shavitt, Topology aggregation for directed graphs, in: Proceedings of IEEE ISCC, Athens, Greece, 1998, pp.47–52.
[7] Y. Bartal, Probabilistic approximations of metric spaces and its algorithmic applications, in: IEEE Symposium on Foundations of Computer Science, 1996, pp. 184–193.
[8] Y. Bartal, On approximating arbitrary metrics by tree metrics, in: STOC, 1998.
[9] Y. Bartal, A constant factor approximation for min sum clustering on HSTs, Personal Communication, 2003.
[10] Y. Bartal, A. Blum, C. Burch, A. Tomkins, A polylog(n)-competitive algorithm for metrical task systems, in: Proceedings of the 29th Annual ACM Symposium on Theory of Computing, El Paso, TX, 4–6 May 1997, pp. 711–719.
[11] Y. Bartal, M. Charikar, R. Raz, Approximating min-sum k-clustering in metric spaces, in: Proceedings of the 33rd Annual ACM Symposium on Theory of Computing, Hersonissos, Crete, Greece 6–8 July 2001, pp. 11–20.
[12] Y. Bartal, M. Mendel, Multi-embedding and path approximation of metric spaces, in: Symposium on Discrete Algorithms, 2003.
[13] Y. Bartal, A. Rosen, The distributed k-server problem—a competitive distributed translator for k-server algorithms, in: IEEE Symposium on Foundations of Computer Science, 1992, pp. 344–353.
[14] J. Bourgain, On lipschitz embeddings of finite metric spaces in hilbert space, Israel J. Math. 52 (1–2) (1985) 46–52.
[15] G. Calinescu, H. Karloff, Y. Rabani, Approximation algorithms for the 0-extension problem, in: Proceedings of the 12th Annual ACM–SIAM Symposium on Discrete Algorithms, 2001, pp. 8–16.
[16] M. Charikar, C. Chekuri, A. Goel, S. Guha, Rounding via trees: deterministic approximation algorithms for group Steiner trees and k-median, in: STOC, 1998, 114–123.
[17] M. Charikar, C. Chekuri, A. Goel, S. Guha, S.A. Plotkin, Approximating a finite metric by a small number of tree metrics, in: IEEE Symposium on Foundations of Computer Science, 1998, pp.379–388.
[18] C. Chekuri, A. Gupta, I. Newman, Y. Rabinovich, A. Sinclair, Embedding k-outerplanar graphs into ℓ_1, in: SODA, 2003.

[19] C. Chekuri, S. Khanna, J. Naor, L. Zosin, Approximation algorithms for the metric labeling problem via a new linear programming formulation, in: Symposium on Discrete Algorithms, 2001, pp. 109–118.
[20] G. Even, J. Naor, S. Rao, B. Schieber, Divide-and-conquer approximation algorithms via spreading metrics (extended abstract), in: IEEE Symposium on Foundations of Computer Science, 1995, pp. 62–71.
[21] J. Fakcharoenphol, C. Harrelson, S. Rao, K. Talwar, An improved approximation for the 0-extension problem, in: Symposium on Discrete Algorithms, 2003.
[22] A. Fiat, M. Mendel, Better algorithms for unfair metrical task systems and applications, in: Proceedings of the 32nd STOC, 2000, pp. 725-734.
[23] U. Feige, Approximating the bandwidth via volume respecting embeddings (extended abstract), in: Proceedings of the 30th Annual ACM STOC, 1998, pp. 90–99.
[24] N. Garg, G. Konjevod, R. Ravi, A polylogarithmic approximation algorithm for the group steiner tree problem, J. Algorithms 37 (2000) 253–259.
[25] N. Garg, V. Vazirani, M. Yannakakis, Approximate max-flow min-(multi)cut theorems and their applications, SIAM J. Comput. 125 (2) (1996) 235–251.
[26] A. Gupta, Steiner points in tree metrics don't (really) help, in: Proceedings of the 12th Annual ACM–SIAM Symposium on Discrete Algorithms, ACM Press, New York, 2001, pp. 220–227.
[27] A. Gupta, I. Newman, Y. Rabinovich, A. Sinclair, Cuts, trees and l_1-embeddings of graphs, in: 40th FOCS, 1999, pp. 399–408.
[28] A. Gupta, A. Srinivisan, On the covering Steiner problem, in: Proceedings of the Annual Conference on Foundations of Software Technology and Theoretical Computer Science, 2003, pp. 244–251.
[29] M. Herlihy, S. Tirthapura, R. Wattenhofer, Competitive concurrent distributed queueing, in: Proceedings of the 20th ACM Symposium on Principles of Distributed Computing, 2001.
[30] T.C. Hu, Optimum communication spanning trees, SIAM J. Comput. 3 (1974) 188–195.
[31] P. Indyk, J. Matousek, Low distortion embeddings of finite metric spaces, in: J.E. Goodman, J. O'Rourke (Eds.), Handbook of Discrete and Computational Geometry, 2nd Edition, 2004.
[32] S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, L. Zhang, On the placement of internet instrumentation, in: Proceedings of IEEE INFOCOM, Tel-Aviv, Israel, 2000.
[33] R. Karp, A 2k-competitive algorithm for the circle, Manuscript, August 1989.
[34] P.N. Klein, S.A. Plotkin, S. Rao, Excluded minors, network decomposition, and multicommodity flow, in: ACM Symposium on Theory of Computing, 1993, pp. 682–690.
[35] J.M. Kleinberg, E. Tardos, Approximation algorithms for classification problems with pairwise relationships: metric labeling and markov random fields, in: IEEE Symposium on Foundations of Computer Science, 1999, 14–23.
[36] G. Konjevod, R. Ravi, An approximation algorithm for the covering steiner problem, Symposium on Discrete Algorithms, 2000, pp. 338–344.
[37] G. Konjevod, R. Ravi, F. Salman, On approximating planar metrics by tree metrics, Inform. Process. Lett. 80 (4) (2001) 213–219.
[38] M.R. Korupolu, C.G. Plaxton, R. Rajaraman, Placement algorithms for hierarchical cooperative caching, in: Proceedings of the 10th Annual ACM–SIAM Symposium on Discrete Algorithms, 1999, pp. 586–595.
[39] R. Krauthgamer, J.R. Lee, M. Mendel, A. Naor, Measured descent: a new embedding method for finite metrics, Manuscript, 2003.
[40] T. Leighton, S. Rao, An approximate max-flow min-cut theorem for uniform multicommodity flow problems with application to approximation algorithms, in: IEEE Symposium on Foundations of Computer Science, 1988, pp. 422–431.
[41] N. Linial, E. London, Y. Rabinovich, The geometry of graphs and some of its algorithmic applications, Combinatorica 15 (1995) 215–245.
[42] J. Matousek, Lectures on Discrete Geometry, Springer, Berlin, 2002.
[43] I. Newman, Y. Rabinovich, A lower bound on the distortion of embedding planar metrics into euclidean space, in: Proceedings of the 18th Annual Symposium on Computational Geometry, ACM Press, New York, 2002, pp. 94–96.
[44] D. Peleg, E. Reshef, Deterministic polylog approximation for minimum communication spanning trees, ICALP, Lecture Notes in Computer Science, Springer, Berlin, Vol. 1443 1998, pp. 670–681.

[45] Y. Rabinovich, R. Raz, Lower bounds on the distortion of embedding finite metric spaces in graphs, Discrete Comput. Geom. 19 (1998) 79–94.
[46] P. Raghavan, Probabilistic construction of deterministic algorithms: approximating packing integer programs, J. Comput. System Sci. 37 (2) (1988) 130–143.
[47] S. Rao, Small distortion and volume preserving embeddings for planar and euclidean metrics, in: Proceedings of the 15th Annual Symposium on Computational Geometry, ACM Press, New York, 1999, pp. 300–306.
[48] Y. Shavitt, Topology aggregation for networks with hierarchical structure: a practical approach, in: Proceedings of 36th Annual Allerton Conference on Communication, Control, and Computing, Allerton Park, IL, 1998.
[49] P.D. Seymour, Packing directed circuits fractionally, Combinatorica 15 (2) (1995) 281–288.
[50] B.Y. Wu, G. Lancia, V. Bafna, K. Chao, R. Ravi, C.Y. Tang, A polynomial time approximation scheme for mimimum routing cost spanning trees, in: ACM–SIAM Symposium on Discrete Algorithms, 1998.