**Advanced Approximation Algorithms** **(CMU 15-854B, Spring 2008)**

# Lecture 20: Embeddings into Trees and L1 Embeddings

March 27 2008

*Lecturer: Anupam Gupta* *Scribe: Aaron Roth*

## 1 Recap

Recall from last time, that we are studying the *sparsest cut* problem. We are given:

- A graph $G = (V, E)$ with positive edge weights $c_e$ for each $e \in E$.
- A set of pairs of vertices $(s_1, t_1), \ldots, (s_k, t_k)$ with associated demands $D_i$ between them.

We wish to output a cut $S$ that minimizes *sparsity*:

$$\Phi(S) = \frac{c(E(S, \bar{S}))}{D(S, \bar{S})}$$

where $c(E(S, \bar{S}))$ is the sum of the weights of the edges that cross the cut, and $D(S, \bar{S})$ is the sum of the demands of the pairs $(s_i, t_i)$ that are separated by the cut.

We recall that optimizing over the set of cuts is equivalent to optimizing over $\ell_1$ metrics, and is NP-hard. Instead, we may optimize over the set of all metrics. In this lecture, we bound the gap introduced by this relaxation by showing how these metrics embed into $\ell_1$ with low distortion.

From last time, we have a theorem of Bourgain:

**Theorem 1.1** (Bourgain 85 [1])**.** *Any $n$ point metric $d$ admits an $\alpha$-distortion embedding into $\ell_p$ for any $1 \leq p \leq \infty$ with $\alpha = O(\log n)$.*

This embedding is into $2^n$ dimensions, however! Fortunately, Linial, London, and Rabinovich [3] proved that it is possible to get such an embedding into online $O(\log^2 n)$ dimensions.

In this lecture, we will show a somewhat weaker result: We will achieve an $\alpha = O(\log n)$ embedding into $\ell_1$ using $\text{poly}(n, d_{\max}/d_{\min})$ dimensions, where $d_{\max}$ is the maximum distance between any two points according to metric $d$, and $d_{\min}$ is the minimum distance. We note that $d_{\max}/d_{\min}$ could potentially be exponentially large – but we won't worry about this detail.

## 2 What is an Embedding?

**Definition 2.1.** An exact embedding of a metric space $(V, d)$ into $\ell_1$ is a mapping $f : V \to \mathbb{R}^k$ such that for all $x, y \in V$, $d(x, y) = ||f(x) - f(y)||_1$,

In the following examples, we will consider graphs, and the corresponding shortest-path metric on vertices, and consider the embedding of these metrics into $\ell_1$.

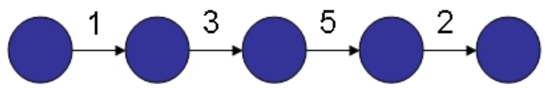


Figure 1: In this example, the line graph would be embedded as: $\{0, 1, 4, 9, 11\}$

## 2.1 Examples

## 2.2 A Line

Consider a line graph $G = (V, E)$ with vertices $V = \{1, \ldots, n\}$ and edges $E = \{(1, 2), (2, 3), \ldots, (n-1, n)\}$. This is easily embedded exactly into $\ell_1$ over $\mathbb{R}^1$. The embedding may be defined recursively: $f(1) = 0$, $f(i) = f(i-1) + d_G(i-1, 1)$.

## 2.3 A Tree

Consider a tree $T = (V, E)$ on $n$ vertices. This can be embedded exactly into $\ell_1$ over $\mathbb{R}^{n-1}$, which can be seen by induction. As the base case, when $|V| = 2$, $T$ is a line graph, which we have seen can be embedded into $\ell_1$ over $\mathbb{R}^1$. Given a tree $T_k = (V_k, E_k)$ with $k$ vertices, we may remove a leaf $v^*$ and the corresponding edge $(u^*, v^*)$ to obtain the tree $T_{k-1} = (V_{k-1}, E_{k-1})$, and inductively embed this into $\ell_1$, using the embedding $f : V_{k-1} \rightarrow \mathbb{R}^{k-2}$. Let $g : V_k \rightarrow \mathbb{R}^{k-1}$ be defined as follows: For every $v \in V_{k-1}$, let $g(v) = (f(v), 0)$. Let $f(v^*) = (f(u^*), d_{T_k}(v^*, u^*))$. Clearly, the $\ell_1$ distances between any two vertices in $V_{k-1}$ have not changed in this new embedding (since we have simply added a 0 in the new coordinate of all such vertices). Finally, the $\ell_1$ distance between any vertex $v \in V_{k-1}$ and $v^*$ is $d(v, u^*) = d(u^*, v^*) = d(v, v^*)$, since there is a unique path between any pair of vertices in a tree.

Given two trees $T = (V, E)$, $T' = (V, E')$ over the same vertex set, we may define a new metric $d(u, v) = d_T(u, v) + d_{T'}(u, v)$ (Recall that metrics form a vector space). The new metric $d$ also has an exact embedding into $\ell_1$. To see this, let $f : V \rightarrow \mathbb{R}^d$ be an exact embedding of $d_T$, and $f' : V \rightarrow \mathbb{R}^d$ be an exact embedding of $d_{T'}$. The metric $d()$ has the embedding $g : V \rightarrow \mathbb{R}^{2d}$ defined by $g(u) = f(u).f'(u)$, the concatenation of $f$ and $f'$. Since $\ell_1$ distance is additive, we therefore have $||g(u) - g(v)||_1 = ||f(u) - f(v)||_1 + ||f'(u) - f'(v)||_1 = d_T(u, v) + d_{T'}(u, v)$ as desired.

Similarly, it is easily seen that the metric $\alpha d_T u, v$ for all $\alpha > 0$ exactly imbeds into $\ell_1$: given an embedding of $d_T$, we may simply scale all coordinates by $\alpha$.

Combining these two facts, we have the following observation:

**Proposition 2.2.** *For any set of trees $T_1, \ldots, T_n$ over the same vertex set, and for all constants $c_1, \ldots, c_n \geq 0$, the metric*

$$d(u, v) = \sum_{i=1}^{n} c_i d_{t_i}(u, v)$$

*embeds exactly into $\ell_1$.*

# 3 The Embedding Result

We will write $\mathcal{D}$ to denote a probability distribution on the set of spanning trees on a fixed vertex set $V$. Going forward, we will concern ourselves with the metric $d$ defined by this distribution:

$$\hat{d}(u,v) = \mathbf{E}_{T\sim\mathcal{D}}[d_T(u,v)]$$

**Proposition 3.1.** *$\hat{d}$ embeds exactly into $\ell_1$.*

*Proof.* This follows directly from proposition 2.2 and the linearity of expectation. Simply set $c_i = \mathbf{Pr}_{\mathcal{D}}[T_i]$. □

Because we embed sums of two metrics as the concatenations of the embedding of each metric, $\hat{d}$ will embed into a space of dimension proportional to the size of the support of $\mathcal{D}$. Therefore, in designing such embeddings, we would like $\mathcal{D}$ to have small support. We will prove the following theorem using tree embeddings, due to Fakcheroenphol, Rao, and Talwar: [2]

**Theorem 3.2.** *Given any metric $d$ on $V$, there exists a distribution $\mathcal{D}$ over spanning trees of $V$ such that for all $u, v \in V$:*

$$d(u,v) \leq \mathbf{E}_{T\sim\mathcal{D}}[d_T(u,v)] \leq O(\log n) \cdot d(u,v)$$

Actually, we will prove something slightly stronger. For all $u, v \in V$:

1. $\mathbf{E}_{T\sim\mathcal{D}}[d_T(u,v)] \leq O(\log n) d(u,v)$, and
2. For *all* $T \in$ Support($\mathcal{D}$), $d_T(u,v) \geq d(u,v)$.

To prove this theorem, we will use the low diameter decomposition covered in a previous lecture. As a reminder:

**Theorem 3.3** (Low diameter decomposition)**.** *Given a metric $(V, d)$, and a parameter $r$, we can construct a random partition of $V = C_1 \uplus \ldots \uplus C_t$ such that the following two properties hold:*

1. *(Low diameter) For all $u, v \in C_i$, $d(u,v) \leq r$.*
2. *(Low cut probability):*

$$\mathbf{Pr}[x, y \textit{ separated by the partition}] \leq \frac{4d(x,y)}{r} \log\left(\frac{|B(x,2r)|}{|B(x,r/4)|}\right)$$

*where $B(x,r)$ denotes the ball around point $x$ of radius $r$.*

We construct the embedding $\hat{d}$ with the following recursive algorithm, which takes as input a pair $(U, i)$ where $U \subseteq V$ is a set of vertices of diameter at most $i$, and returns a rooted tree $(T, r)$.

TreeEmbed$(U, i)$:

1. Apply the low-diameter decomposition to $(U, d)$ with the parameter $r \rightarrow 2^{i-1}$ to get the partition $C_1, \ldots, C_t$.

2. Recurse: Let $(T_j, r_j) \leftarrow$ TreeEmbed$(C_j, i-1)$. As a base case, when $C_i$ is a single point, simply return that point.

3. For every tree $T_j$ with $j > 1$, add the edge $(r_1, r_j)$ with weight $2^i$. This is a new tree which we denote $T$.

4. Return the tree/root pair $(T, r_1)$.

Recall that since the low diameter decomposition is randomized, this algorithm defines a distribution over trees, as desired.

We rescale so that for all $u, v \in V$, $d(u, v) \geq 1$ and $d(u, v) \leq \delta = 2^\delta$. We may therefore draw from the distribution defining $\hat{d}$ by calling TreeEmbed$(V, \delta)$.

**Claim 3.4** (Claim 1)**.** *For all $u, v \in V$, $\hat{d}(u, v) \geq d$.*

*Proof.* Fix $x$ and $y$, and let $i$ be such that $d(x, y) \in (2^{i-1}, 2^i)$. Consider the invocation of TreeEmbed$(U, i)$ such that $x \in U$. First, we examine the case in which $y \in U$. By the definition of the low diameter decomposition, since $d(x, y) > 2^i$, $x$ and $y$ will fall into separate parts of the partition, and so we will have $\hat{d}(x, y) \geq 2^i$, the length of the edge placed between partition subtrees. In the case in which $y \notin U$, then it must be that $x$ and $y$ have been separated at a higher level of the recursion, $i'$, and so are separated by a higher subtree edge. Therefore, $\hat{d}(x, y) \geq 2^{i'} > 2^i$. □

**Claim 3.5.** *For all $x, y \in V$:*

$$\hat{d}(x, y) \leq d(x, y) \cdot O(\log n)$$

*Proof.* We begin the proof with two easy subclaims Suppose $(T, r) \leftarrow$ TreeEmbed$(U, i)$:

1. Claim 1: $d_T(r, x) \leq 2^{i+1}$ for all $x \in U$. This holds because $x$ is in some partition with diameter at most $2^i$ by definition of the low diameter decomposition, and is possibly separated from the root $r$ by an intertree edge of weight $2^i$.

2. Claim 2: If $x, y \in U$, then $d_T(x, y) \leq 2 \cdot 2^{i+1}$. This is immediate from the previous claim, since each $x$ and $y$ is at distance at most $2^{i+1}$ from $r$, and distances are symmetric.

We now have from the definition:

$$\begin{aligned}
\hat{d}(x,y) &\leq \sum_{i=\delta}^{0} \mathbf{Pr}[(x,y) \text{ separated at level i}] \cdot 4 \cdot 2^i \\
&\leq \sum_{i=\delta}^{0} \frac{d(x,y)}{2^{i-1}} \cdot \log\left(\frac{|B(x,2^i)|}{|B(x,2^{i-3})|}\right) \cdot 2^i \cdot 4 \\
&= 8d(x,y)\sum_{i=0}^{\delta}(\log(|B(x,2^i)|) - \log(|B(x,2^{i-3})|)) \\
&= 8d(x,y)(\log(|B(x,2^\delta)|) + \log(|B(x,2^{\delta-1})|) + \log(|B(x,2^{\delta-2})|) \\
&\leq 24d(x,y)\log n
\end{aligned}$$

where the first inequality follows from our subclaims, the second follows from the definition of the low diameter decomposition, and the last equality follows from observing that we have a telescoping sum. □

We note that we have defined a distribution $\mathcal{D}$ with a huge support, but with a bit more work, it is possible to show that a distribution over $O(n \log n)$ trees suffices.

# 4 Concluding Remarks

Recall that in order to solve sparsest cut, we really wished to solve for:

$$\Phi^* = \min_{\mu \in \ell_1} \frac{C \cdot \mu}{D \cdot \mu}$$

the minimization over $\ell_1$ metrics (we *really* want the minimal cut metric, but we showed this was equivalent to the minimum $\ell_1$ metric). However, solving this problem is NP hard, and so we solved instead for:

$$\lambda^* = \min_{d \text{ a metric}} \frac{C \cdot d}{D \cdot d}$$

the minimization over all metrics, which can be expressed as a linear program. In this lecture, we showed that this creates at worst a $\log n$ gap (and in fact, this is tight).

Can we do better by minimizing over something else? How about $\ell_2$ metrics? Unfortunately, the set of $\ell_2$ metrics is not convex, and so we can't minimize over it efficiently. How about $\ell_2^2$, the space of squared $\ell_2$ metrics? This set is convex, but unfortunately gives an even worse gap ($\approx n$). In fact, these aren't even metrics – they don't satisfy the triangle inequality.

How about, then, $\mathcal{C} = \ell_2^2 \cap$ metrics?. It is possible to optimize over $\mathcal{C}$ with an SDP, and indeed, this gives the best known factor approximation, of $\sqrt{\log n} \cdot (\log \log n)$.

# References

[1] J. Bourgain. On lipschitz embedding of finite metric spaces in Hilbert space. *Israel Journal of Mathematics*, 52(1):46–52, 1985.

[2] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485–497, 2004.

[3] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.