Advanced Algorithms	10/31, 2017
Lecture 7	
Lecturer: Mohsen Ghaffari	*Scribe: Martin Nägele*

This draft has not been checked by the lecturer yet.

1 Probabilistic Tree Embedding

The technique of probabilistic tree embeddings was introduced by Bartal in 1996 [Bar96]. The motivation is that many metric problems on graphs are easy to solve if the graph is a tree (and thus, the metric is a tree metric), so if we can approximate a general graph by a tree, we can solve the problem on the tree and thus obtain an approximation for the original problem.

1.1 Setting

An undirected graph $G = (V, E)$ induces a metric space (V, d_G), where $d_G : E \to \mathbb{R}_{\geq 0}$ given by

$$d_G(u, v) = \text{distance between } u \text{ and } v \text{ in } G$$

is the graph metric. Ideally, we would like to find a tree T on a vertex set V', $V \subseteq V'$, such that the induced tree metric d_T satisfies

$$\forall u, v \in V : \quad d_G(u, v) \leq d_T(u, v) \leq c \cdot d_G(u, v) \ ,$$

for some reasonably small factor c. However, by considering G to be a cycle, we immediately see that we need $c = \Omega(n)$, as the one edge of the cycle not contained in T is stretched by a factor of $n - 1$. Thus, Bartal introduced the notion of *probabilistic tree embeddings*. Instead of a single tree, we consider a collection $\mathcal{T} = \{T_1, T_2, \ldots, T_k\}$ of trees on a vertex set $V' \supseteq V$, together with a probability distribution on $\mathcal{T}$ such that $\Pr(T_i) = p_i$, for some $p_1, \ldots, p_k$ with $\sum_{i=1}^{k} p_i = 1$. We will see that for any graph G, there exists such a probabilistic tree embedding with the following two properties:

$$\forall u, v \in V,\ \forall i \in [k] : \quad d_G(u, v) \leq d_{T_i}(u, v) \ , \tag{1}$$

$$\text{and} \quad \forall u, v \in V : \quad \mathbb{E}_{T \sim \mathcal{T}}[d_T(u, v)] \leq O(\log n) \cdot d_G(u, v) \ . \tag{2}$$

As shown by Bartal in [Bar96], the factor of $O(\log n)$ is asymptotically best possible.

1.2 FRT tree embeddings

We present an algorithm by Fakcharoenphol, Rao, and Talwar [FRT03] constructing a probabilistic tree embedding of a graph $G = (V, E)$ with the above properties.

An iterative construction. Suppose D is the diameter of G. The idea is to decompose G into disjoint subgraphs $G_1, G_2, \ldots, G_\ell$ (in the sense of partitioning the vertices of G) using a randomised procedure such that

(a) for all $i \in [\ell]$, we have $\operatorname{diam}(G_i) \leq \frac{D}{2}$, and

(b) for all $u, v \in V$, the probability that u and v are assigned to different parts G_i is at most $\frac{d_G(u,v)}{D} \cdot \alpha$, for some α.

We will see a method to achieve the above decomposition with $\alpha = O(\log n)$. For now, we assume that we can obtain the stated guarantees for any graph. We construct a first layer of our tree T by invoking the decomposition result for G, introducing a root r and connecting it the components G_i with edges of weight D. We iteratively apply this construction to all components, with the weights on the edges reducing by a factor of 2 in each step, see Figure 1.

The iterative construction ends once all remaining subgraphs are single nodes. Thus, the resulting tree T has precisely the vertices of G as leaves, and all internal vertices were added in the process of

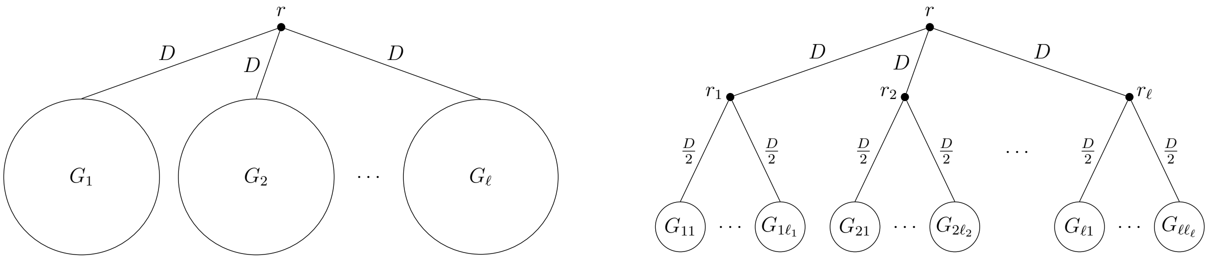

Figure 1: Two steps of constructing a tree T by iteratively splitting graphs into subgraphs.

constructing the tree. Note that in each step, the partition of the current graph into disjoint subgraphs will be done using a randomised procedure, thus the resulting tree T is actually a random object from a finite collection $\mathcal{T}$ of trees.

Moreover, note that the weights on the edges of any path from the root r to a leaf have weights $D, \frac{D}{2}, \frac{D}{4}, \ldots$, hence its length is, if the path has k edges,

$$D + \frac{D}{2} + \frac{D}{4} + \ldots + \frac{D}{2^{k-1}} < \frac{D}{1-\frac{1}{2}} = 2D \ ,$$

and thus, T satisfies $\operatorname{diam}(T) \leq 4D$. Moreover, by induction and using property (a) of the graph partitions, it is easy to see that for any $u, v \in V$, we have $d_G(u,v) \leq d_T(u,v)$ on every level of the above iterative construction, and hence also at the very end. Consequently, the family of trees T that can be obtained through our construction satisfies property (1).

To see that property (2) is satisfied as well, fix $u, v \in V$ and let A_i be the event that the nodes u and v are separated at the i^{th} level of the recursion. Then, we have

$$\mathbb{E}[d_T(u,v)] \leq \Pr[A_1] \cdot 4D + \Pr\left[A_2 | \overline{A_1}\right] \cdot \frac{4D}{2} + \Pr\left[A_3 | \overline{A_1} \cap \overline{A_2}\right] \cdot \frac{4D}{4} + \ldots \ ,$$

because the distance of two vertices that are separated at the i^{th} level can be bounded by $\frac{4D}{2^{i-1}}$. Calculating the probabilities in the above bound is easy: By property (a) of the graph decomposition, a subgraph at the $(i-1)^{th}$ level of the recursion has diameter at most $\frac{D}{2^{i-1}}$, and hence by property (b), the probability of separating two vertices at level i is bounded by $\frac{d_G(u,v)}{D/2^{i-1}} \cdot \alpha$ from above. Using this, we get the bound

$$\mathbb{E}[d_T(u,v)] \leq \frac{d_G(u,v)\alpha}{D} \cdot 4D + \frac{d_G(u,v)\alpha}{D/2} \cdot \frac{4D}{2} + \frac{d_G(u,v)\alpha}{D/4} \cdot \frac{4D}{4} + \ldots = 4\alpha \cdot \log D \cdot d_G(u,v) \ , \qquad (3)$$

where the factor $\log D$ comes from the fact that the recursion has at most $\log D$ many levels, as the initial graph G has diameter D.

Using the decomposition method with $\alpha = O(\log n)$, we thus get property (2) with a factor of $O(\log D \cdot \log n)$ instead of $O(\log n)$. A better analysis will later improve this to the desired factor.

Randomised graph decomposition. We now put the focus on obtaining the decomposition result used above. There are various such results in literature, including results by Linial and Saks (1993) [LS91], Bartal (1996) [Bar96], and Carlinescu, Karloff, and Rabani (2001) [CKR01]. We study the last one here, where the algorithm is as follows:

1. Pick a radius $\theta \in \left[\frac{D}{8}, \frac{D}{4}\right]$ at random.
2. Permute, via a random permutation π, all vertices of G.
3. The i^{th} subgraph G_i is the graph induced by the vertices in $B(\pi(i),\theta) \setminus \bigcup_{j<i} B(\pi(j),\theta)$ (balls use distances with respect to G, and we still build the i^{th} ball if $\pi(i)$ is already included).

With this construction, each of the balls obviously has radius at most $\frac{D}{4}$ and thus diameter at most $\frac{D}{2}$, so property (a) is guaranteed by construction, and it remains to show property (b). We need to obtain an upper bound on the probability that the two vertices u and v are not assigned to the same component G_i. As an upper bound, it is enough to bound the probability that the whole ball $B(u,r)$ is not assigned to the same component. If $B(u,r)$ is not assigned to the same component, we say that $B(u,r)$ is *cut*. Note that in this case, there is a vertex $w \in V$ and $i \in [n]$ with $\pi(i) = w$ such that

(a) for all $j < i$, $G_j \cap B(u,r) = \emptyset$, and

(b) $G_j \cap B(u,r) \neq \emptyset$ and $B(u,r) \not\subseteq G_j$.

We then say that $B(u,r)$ *is cut by* w. In other words, if $B(u,r)$ is cut by $w = \pi(i)$, then the component G_i cuts $B(u,r)$ into two pieces (condition (b)), and this did not happen when constructing previous components G_j with $j < i$. For the analysis, let $v_1, v_2, \ldots$ be the vertices of G sorted in a non-decreasing order based on the distance from u. Note that

$$\Pr[B(u,r) \text{ is cut}] = \sum_{i=1}^{n} \Pr[B(u,r) \text{ is cut by } v_i] \tag{4}$$

There are two things that need to happen such that $B(u,r)$ is cut by v_i: First, the radius θ has to be in the right range, which is the interval $(d(v_i,u) - r, d(v_i,u) + r)$ of length $2r$. Second, all vertices $v_1, \ldots, v_{i-1}$ (which have shorter distance to u than v_i) have to appear after v_i in the permutation π, as else, v_i would not be the first vertex at which the ball $B(u,r)$ is touched. The corresponding probabilities for these events are at most $\frac{2r}{D/8}$ and $\frac{1}{i}$, respectively. Plugging this into the above sum, we obtain

$$\Pr[B(u,r) \text{ is cut}] \leq \sum_{i=1}^{n} \frac{2r}{D/8} \frac{1}{i} \leq \frac{16r \cdot \ln n}{D} \quad , \tag{5}$$

which is the desired result with $\alpha = c \cdot \log n$, for some constant c.

Improving the analysis. We can assume without loss of generality that $r \leq \frac{D}{16}$. Note that in (4) and (5), we do not have to sum over all vertices: It is enough to sum over vertices v_i such that $\theta - r \leq d(v_i,u) \leq \theta + r$. Indeed, if $\theta - r > d(v_i,u)$, then all of $B(u,r)$ is covered by $B(v_i,\theta)$; if $d(v_i,u) > \theta + r$, then $B(u,r)$ and $B(v_i,\theta)$ are disjoint. By the assumption on the size of r, we immediately see that the interesting vertices v_i are a subset of those v_i with $d(v_i,u) \in \left[\frac{D}{16}, \frac{D}{2}\right]$. Restricting the harmonic sum in (5) to only the vertices with distances in this range, we obtain the better bound

$$\Pr[B(u,r) \text{ is cut}] \leq \frac{2r}{D/8} \sum_{i=|B(u,D/16)|}^{|B(u,D/2)|} \frac{1}{i} \leq \frac{16r}{D} \cdot \ln\left(\frac{|B(u,D/2)|}{|B(u,D/16)|}\right) \quad ,$$

and thus, we obtain the decomposition property (b) with a factor $\alpha_D = \ln\left(\frac{|B(u,D/2)|}{|B(u,D/16)|}\right)$, depending on the diameter of the graph that is decomposed. Using this in (3), we get the bound

$$\mathbb{E}[d_T(u,v)] \leq 4 \cdot d_G(u,v) \cdot (\alpha_D + \alpha_{D/2} + \alpha_{D/4} + \ldots) \leq 12 \cdot \ln n \cdot d_G(u,v) \quad ,$$

which is of the desired form $O(\log n) \cdot d_G(u,v)$. Indeed, note that in the sum $\alpha_D + \alpha_{D/2} + \ldots$, after writing $\log(x/y) = \log(x) - \log(y)$, all but three terms cancel, and each of these can be bounded by $\ln n$.

2 Application: Buy-at-Bulk Network Design

Given: A weighted undirected graph $G = (V,E)$ with edge lengths ℓ_e for all $e \in E$, pairs (s_i,t_i) of vertices, and demands d_i.

Goal: For each i, route d_i units of commodity i from s_i to t_i through some s_i-t_i-path P_{s_i,t_i} in G such that the total costs are minimized, where the cost of a connection of capacity c_e on edge e is $f(c_e) \cdot \ell_e$, for some subadditive function f, i.e., a function with the property $f(x+y) \leq f(x) + f(y)$. For each edge $e \in E$, c_e must be $\sum_{e \in P_{s_i,t_i}} D(i)$.

Special case: Trees. In this case, finding a solution is trivial: There is only one unique path between any two vertices, so there is no alternative to choosing that path. We can thus hope that obtaining a probabilitstic tree embedding for the graph metric of G, and solving the problem on the tree, gives an approximation.

Algorithm. Let d be the distance metric in $G = (V, E)$, and embed d into a probabilistic tree metric (V', T). We first want to transform this tree metric to another tree metric (V, T') on the original vertex set. For each $v \in V$ such that the parent of v in T is not in V, contract the edge connecting V to its parent. This potentially decreases distances, but if we multiply everything by a factor of 4 in the end, we are safe (by the geometric decay of the distances in the tree that we constructed). Now solve the network design problem on the tree T' (i.e., find the right paths in T'), and project it back into the original graph: For each edge (x, y) in T', find a shortest x-y-path in in the original graph, and then combine all those paths to obtain a solution in G.

Analysis. Let OPT be the optimal solution in the original graph, let SOL denote our solution. We have

$$\text{SOL projected back on } G \leq \text{SOL on } T'$$

$$\text{SOL on } T' \leq \text{OPT projected to } T'$$

$$E[\text{OPT projected to } T'] \leq O(\log n) \cdot \text{OPT}$$

The first inequality follows from subadditivity of f and the fact that distances in G are shorter than in G'. The second one is true as SOL is the best solution on T'. Finally, the third one follows from the fact that distances are streched by a $\log n$-factor, so costs increase by a $\log n$-factor. Together, the three inequalities prove that we indeed found a $O(\log n)$-approximation (in expectation).

References

[Bar96] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. In *Proc. of the Symp. on Found. of Comp. Sci. (FOCS)*, pages 184–, 1996.

[CKR01] Gruia Calinescu, Howard Karloff, and Yuval Rabani. Approximation algorithms for the 0-extension problem. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '01, pages 8–16, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[FRT03] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proc. of the Symp. on Theory of Comp. (STOC)*, pages 448–455, 2003.

[LS91] Nathan Linial and Michael Saks. Decomposing graphs into regions of small diameter. In *Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '91, pages 320–330, 1991.