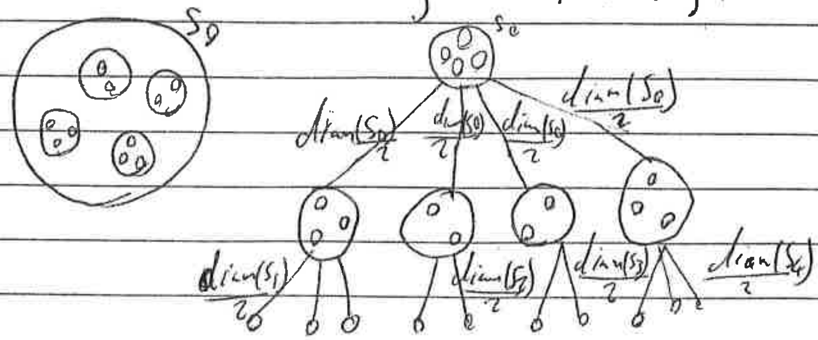
Date

Here we outline the probabilistically embedding an n-point metric into a tree with expected distortion $O(\log n)$ based on Bartal & FRT results.

- First decompose the graph hierarchically
- Then convert the resulting laminar family to a tree.



- let the input metric be (V,d)
- The minimum distance is strictly more than 1 (by scaling)
- Δ be the diameter of (V,d)
- $\Delta = 2^\delta$

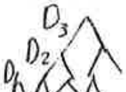
- an r-cut decomposition of (V,d) is a partition of V into clusters, each centered around a vertex and having radius at most r (thus the diameter of each cluster is at most $2r$). A hierarchical cut decomposition of (V,d) is a sequence of $\delta+1$ nested cut decompositions $D_0, D_1, \ldots, D_\delta$ such that [note that the diameter of a cluster in D_j is at most 2^{j+1}]
- $D_\delta = \{V\}$, 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}.

Since all the distances are at least one, and since each cluster in D_0 has radius at most 1, each cluster in D_0 must be a singleton vertex.

A hierarchical cut decomposition defines a laminar family, and naturally corresponds to a rooted tree as follows. Each set in the laminar family is a node in the tree and the children of a node corresponding to a set S are the nodes corresponding to maximal subsets of S in the family.

(Recall a laminar family $F \subseteq 2^V$ is a family of subsets of V such that for any $A, B \in F$, it is the case that $A \subseteq B$, or $B \subseteq A$ or $A \cap B = \emptyset$).

The distance $d^T(u,v)$ is equal to the length of the shortest path distance in T from node $\{u\}$ to node $\{v\}$.



Lm 1: For any tree T as generated above dominates d, i.e., $d_T(x,y) \geq d(x,y)$

Pf: let x,y be such that $\frac{\Delta}{2^j} < d_G(x,y) \leq \frac{\Delta}{2^{j-1}}$. Since x and y are at distance greater than $\frac{\Delta}{2^j}$, they cannot lie in the ~~same~~ level j_{-1} component. Hence they were separated at some level $j' \leq j$ and (j from the bottom of the tree)
the edges of length $\frac{\Delta}{2^{j'}}$ added at this level to connect their subtrees ensure that x and y are at distance at least $2\frac{\Delta}{2^{j'}} \geq \frac{\Delta}{2^{j-1}} \geq d_G(x,y)$ ✓

— $\boxed{\text{Now, we want to put an upper bound on } d^T(u,v)}$.

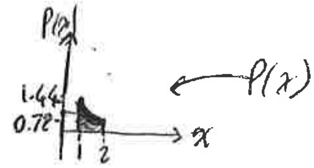
We say an edge (u,v) is at level i if u and v are first separated (cut) in the decomposition D_i. Note that if (u,v) is at level i, then $d^T(u,v) = 2\sum_{j=0}^{i} 2^j \leq 2^{i+2}$

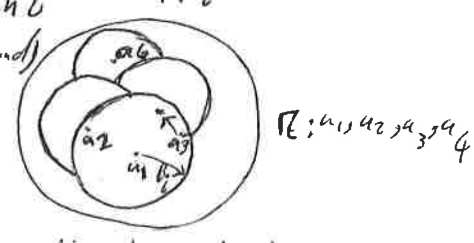
— Also, note the difference of diameter (the maximum distance in the graph) and weak diameter (the maximum distance in the original graph).

e.g. the weak diameter (and not diameter) of each cluster in D_j is at most 2^{j+1}

The 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 $p(x) = \frac{1}{x \ln 2}$ (instead we could choose β uniformly at random from $[1,2]$ with slightly worse bound)
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. $i \leftarrow i - 1$





Illustrating the construction

Note that our randomized algorithm above obtain the laminar decomposition, which will implicitly define the distribution of trees that are used in the embedding.

- Note that by the definition of our distribution

Observation 1: For any $x \geq 1$, $\Pr[\text{some } \beta_i \text{ lies in } [x, x+dx]] \leq \frac{1}{x \ln 2} dx = \frac{1}{(\frac{x}{2^i}) \ln 2} \cdot \frac{dx}{2^i}$ (scale; β_i chooses the desired value)

Fix an arbitrary edge (u,v), we show the expected value of $d_T(v,u)$ is in $O(\log n)\, d(u,v)$.

- Consider level i of clustering. In each iteration, all unassigned vertices v such that $d(v, \pi(l)) \le \beta_i$ assigns themselves to $\pi(l)$.
- First u and v are both assigned. Then at some step l u is assigned to w (v may or may not) for the first time (we say w settles (u,v)).

We say w cuts (u,v) if w settles (u,v) and exactly one is assigned to w.

- Whenever a vertex w cuts edge (u,v) at level i, the tree length of the edge (u,v) is about 2^{i+2}, as we discussed. We blame w for this length.
- Let $d_w^T(u,v) = \sum_{i=1} \mathbb{1}(w \text{ cuts } (u,v) \text{ at level } i) \cdot 2^{i+2}$ ($\mathbb{1}(\cdot)$ is the indicator function)

Thus $d^T(u,v) \le \sum_w d_w^T(u,v)$.

- Sort the vertices of V in order of increasing distance from edge (u,v) (from either end and breaking ties arbitrarily). Let w_s be the sth vertex and say $d(w_s,u) \le d(w_s,v)$.

→ We compute $d_{w_s}^T(u,v)$.

w_s cuts (u,v) if
$$\begin{cases} d(w_s,u) \le \beta_i < d(w_s,v) \text{ for some } i \\ w \text{ settles } (u,v) \text{ at level } i. \end{cases}$$

and if it happens the contribution to $d_{w_s}^T(u,v)$ is at most $2^{i+2} \le 8\beta_i$

- The probability that some β_i falls in $[x, x+dx)$ for a particular $x \in [d(w_s,u), d(w_s,v)]$ is at most $\frac{1}{x \ln 2} dx$ according to observation 1.
- Conditioned on β_i taking this value x, any of $w_1, w_2, \ldots w_s$ can settle (u,v) at level i (trivial) then the conditional probability is at most $\frac{1}{s}$. Thus the expected cost of $d_{w_s}^T(u,v)$ is at most
$$\int_{d(w_s,u)}^{d(w_s,v)} \frac{1}{x \ln 2} \cdot \underbrace{8x}_{8\beta_i} \cdot \frac{1}{s}\, dx = \frac{8}{s \ln 2}\left(d(w_s,v) - d(w_s,u)\right) \le \frac{8}{s \ln 2} d(u,v)$$
(if β_i was uniform in $[1,2)$ then we had $\frac{1}{s} \cdot \frac{d(w_s,v)-d(w_s,u)}{2^2} \cdot 8\beta_i \le \frac{8\, d(u,v)}{s}$) by triangle ineq.

- Thus by linearity of expectation
$$E[d^T(u,v)] \le \sum_s \frac{8\, d(u,v)}{s \ln 2} = \frac{8\, d(u,v)}{\ln 2} \cdot H_n \le 8 \frac{\ln n}{\ln 2} d(u,v) = 8 \log n\, d(u,v).$$

☆ Can be derandomized by the method of conditional expectation and give $O(n \log n)$ trees.

☆ The steiner points of trees can be removed by an extra factor 8 in distortion (by Gupta's lemma) from Awerbuch-Azar'97

Uniform buy-at-bulk network design: A Graph $G(X,E)$ where $|X|=n$

- we are given pairs $(s_1,t_1),(s_2,t_2),\ldots,(s_k,t_k)$ where the demand $s_i t_i$ is dem_i
- Each edge e has length l_e. We want to send all the traffic from u to v on a single path P_{uv} (i.e. it is indivisible) - w.l.o.g. assume l_e is a metric.
- The edge of the network must be installed by purchasing zero or more copies from the set of cables where each cable type i has specified capacity u_i and cost c_i.

No assumption about c_i & u_i, but usually if $u_i < u_j$ then $c_i < c_j$ (otherwise get rid of type i) and $\frac{c_j}{u_j} < \frac{c_i}{u_i}$ (economy of scale). These are not necessary for our algorithms.

Algorithm: first we decide about the route $P_{s_i t_i}$. Then we buy enough capacity. Let $C(dem)$ be the minimum cost to cover a total demand of dem for unit distance. (has PTAS by Knapsack)

Lm 1: $\min_i c_i \lceil \frac{dem}{u_i} \rceil$ is a 2 approximation for $C(dem)$.

Pf: By an averaging argument we need at least $\min_{i|u_i<dem} c_i \frac{dem}{u_i}$. We have two approximation since $\lceil \frac{dem}{u_i} \rceil < \frac{dem}{u_i}+1 \le \frac{2dem}{u_i}$

Lm 2: $C(dem)$ is subadditive, i.e. $C(x+y) \le C(x)+C(y)$.

Pf: by definition

Our Algorithm: Use FRT, route in a tree with min total cost, map (smear) the solution back into the graph and install enough capacity (use 2-approximation). (note that during this maping the total cost can only decreases with respect to the tree cost C_T, due to shorter distances or subadditivity.

So we only need to prove

Thm: Embedded OPT on set of trees has expected cost at most OPT_G.

Suppose opt routes u,v via path Q_{uv} in G. then for each edge e in G let f_e be the flow that passes through e in G. Thus

$$f_e = \sum_{(u,v)|e\in Q_{uv}} dem_{uv}$$

and thus $OPT_G = \sum_{e \in E(G)} l(e)\, C(f_e)$.

Consider an edge $e=(x,y)$ in G. We associate with this edge in each tree $T \in Support(D)$ a path T_e between x and y of length $d_T(x,y) = l(e)\,\alpha_T$ where $E(\alpha_T) = O(\log n)$ by FRT. Note that want to send a flow f_e through this path T_e.

By linearity of expectation we are focusing only on one tree T. The cost of designing a path T_e with a flow of f_e on each edge $e' \in T_e$ satisfies $\sum_{e' \in T_e} l(e')\, C(f_e) = d_T(x,y)\, C(f) = l(e)\,\alpha_T\, C(f)$.

Now, if we design a network in T for all paths T_e for all $e \in E(G)$ each with demand f_e the the flow $f'_{e'}$ on each edge $e' \in E(T)$ is $f'_{e'} = \sum_{e \in E(G) | e' \in T_e} f_e$ and its cost due to subadditivity of C is

$$\sum_{e' \in E(T)} l(e')\, C(f'_{e'}) = \sum_{e' \in E(T)} l(e')\, C\Big(\sum_{e \in E(G) | e' \in T_e} f_e\Big) \leq \sum_{e' \in E(T)} l(e') \sum_{e \in E(G) | e' \in T_e} C(f_e)$$

$$= \sum_{e' \in E(T)} \sum_{e \in E(G) | e' \in T_e} l(e')\, C(f_e) = \sum_{e \in E(G)} \sum_{e' \in T_e} l(e')\, C(f_e) = \sum_{e \in E(G)} l(e)\,\alpha_T\, C(f_e) = \alpha_T \sum_{e \in E(G)} l(e)\, C(f_e) = \alpha_T\, OPT_G$$

Note that a path Q_{uv} in G is associated with a (maybe non-simple) path $Q_{T,(u,v)}$ in T which consists of concatenation of paths in T associated with the edges of Q_{uv} and

$$f'_{e'} = \sum_{e \in E(G) | e' \in T_e} f_e = \sum_{e \in E(G) | e' \in T_e} \sum_{(u,v) | e \in Q_{uv}} dem_{u,v}$$

$$= \sum_{(u,v) | e' \in T_e,\, e \in Q_{uv}} dem_{u,v} = \sum_{(u,v) | e' \in Q_{T,(u,v)}} dem_{u,v}$$

which implies that the network that we designed for each T is a feasible solution for demand.

In next sessions, we show how we can obtain polylog approximation for non-uniform buy-at-bulk. I.e. the c_e and u_e of different edges are different.