Session (4)

In this class, we are considering n point metric space X
We have a distance function $d: X \times X \rightarrow R_+$ that

$$\begin{cases} d(i,i)=0 \text{ for all } i \in X \\ d(i,j)=d(j,i) \text{ for all } i,j \in X \text{, i.e. it is symmetric} \\ d(i,j) \leq d(i,k)+d(k,j) \quad \forall i,j,k \in X \text{, i.e., satifies the triangle inequality.} \end{cases}$$

We represent finite metric spaces with (X,d) and e.g. can be represented by a symmetric Matrix M. Usually to visualize metrics, we represent them by a graph $G(d)$ where set X is the set of vertices of the graph and we have an edge between nodes $i,j \in X$ with length $d(i,j)$. (note that the shortest-path of G is identical to the original metric d)

Conversely, given a graph G on n vertices endowed with lengths on the edges, we can get a natural metric d_G by setting, for every $i,j \in V(G)$, the distance $d_G(i,j)$ to be the length of the shortest path between i and j in G (note that G is always undirected.

note that other than graphs, X can be a set of points in l_2 (i.e. Euclidean space) or in l_1 (i.e., Manhattan metric) or L_∞ (max diff. in any coordinate), but we generally focus on Graph metrices in this course (though we use graphs and metrics interchangably in this course)

Given metric spaces (X,d) and (X',d'), a map $f: X \rightarrow X'$ will be called an embedding (a distance-preserving embedding will be called isometric) embedding is a general term for any map from a metric into another

Why embedding? since we can embed a hard problem into line, trees, or even l_1 for which we can solve the problems easily.

we focus on embedding of graph metrices into trees in this course

Embeddings with distortion: It is hard to get isometric embedding thus we consider "close" distorted embedding instead.

Given two metrics (X,d) and (X',d') and a map $f: X \to X'$, the contraction of f is the maximum factor by which distances are shrunk, i.e.,

$$\max_{x,y \in X} \frac{d(x,y)}{d'(f(x),f(y))},$$

the expansion or stretch of f is the maximum factor by which distances are stretched: $\max_{x,y \in X} \frac{d'(f(x),f(y))}{d(x,y)}$

the distortion of f, $\alpha(f)$, is the product of the contraction and the expansion. A very useful property of distortion is that it is invariant under scaling. Hence in many arguments we assume that the embedding is non-contractive and thus $\alpha(f)$ is just its expansion. i.e.

$$d(x,y) \leqslant d'(f(x),f(y)) \leqslant \alpha\, d(x,y),$$ where α is the distortion.

It would be ideal if we could always embed into a tree with low distortion but it is not possible always. e.g. see C_n a cycle of length n needs distortion $\Omega(n)$ into a tree.

However much better results can be obtained if one wants to embed the given metric (X,d) into a distribution of trees, instead of one tree, so that the expected distances are preserved.

Formally, we say a metric (X,d) embeds probablistically into a distribution D of trees with distortion α:

- Each tree $T=(V_T, E_T)$ in the support of the distribution D contains the points of the metric; i.e. $X \subseteq V_T$. Furthermore, the distances in T dominate those in d; i.e., $d_T(x,y) \geqslant d(x,y)$ for all $x,y \in X$. support of $D = \{T_1, T_2, \dots, T_k\} \to k$ in $O(n \log n)$
  probability: $P_1, P_2, \dots, P_k$

- Given a pair of vertices $x,y \in X$, the expected distance in not too much larger than $d(x,y)$ i.e., $E[d_T(x,y)] = P_1 d_{T_1}(x,y) + P_2 d_{T_2}(x,y) + P_3 d_{T_3}(x,y) + \dots + P_k d_{T_k}(x,y) \leq \alpha\, d(x,y)$

For example: consider a cycle C_n of unit length edges: D can be obtained by taking each tree obtained from deleting an edge. Then α is $1 \frac{n-1}{n} + n-1 \frac{1}{n} = 2(1-\frac{1}{n})$. seems very good

Bartal first showed α is in $O(\log^2 n)$, then he showed $O(\log n \lg\lg n)$
Fakcharoenphol, Rao and Talwar (FRT) improved the result above to $O(\lg n)$: This tight even for very simple (planar) graphs.

Note that at the beginning X can be a strict subset of V_T. (by a result of Anupam Gupta'01, we can remove steiner nodes $(V_T - X)$ by paying an extra constant distortion 8) but still the edge sets and their lengths can be quite different from G. However Elkin, Emek, Spielman, Teng05 showed that α is in $O(\lg^2 n \cdot \lg\lg n)$ even when the support of D is from the spanning subtrees of G (note that the domination is trivial then). Again the lower bound is $\Omega(\lg n)$ but improving the gap is a very important open problem (the conjecture is $\Theta(\lg n)$).
Usually working with EEST'05 is easier, but we usually can improve the bounds by using FRT'03. The technique of proving FRT is important and quite simple and useful. Thus, we present it in the next session. Now let's see some applications.

A New Result by Abraham, Bartal and Neiman improve the result of EEST to $\tilde{O}(\log n)$ $= O(\log n \cdot \log\log n (\log\log\log n)^3)$ in FOCS'08

[Find other two Applications as an exercise and a homework]

First example: Steiner tree: given a set of vertices (terminals) $t_1, t_2, \dots, t_n$ connect them to a root S in a graph G in which each edge e has a buying cost b_e.
- First metric (X, b) should be a metric. Otherwise make it a metric!!!
- Use EEST, You have a set of trees in each tree there is a unique path from each t_i to S, buy all the edges on these path. output the solution with the minimum total cost C (among k solutions of k trees in D) in tree T_C

We prove $C \le \lg^2 n \lg\lg n \, OPT_G$.
[Note that since T_C is spanning subgraph of G, we have already a solution to G correct in EEST'05 but not FRT'03]

let OPT_G be the solution (and its cost) in G. Now for each edge $e=(u,v)$ of OPT_G we map that edge to the unique path between u,v in T_C and any tree T_i where $i \neq C$ (and buy such edge.)
Note that ~~if~~ we obtain a solution OPT_i in each tree T_i where $OPT_i \geq C_i \geq C$

C_i = opt for tree$_i$, C = global optimum

Now: we want to prove $C \leq \log_n^2 \log\log n \; OPT_G$.

$$\log_n^2 \log\log n \; opt(G) \geq \sum_{i=1}^{k} P_i OPT_i \geq \sum_{i=1}^{k} P_i C_i \geq \sum_{i=1}^{k} P_i C \geq C \sum_{i=1}^{k} P_i = C \quad \checkmark \quad (A)$$

Now if we use FRT instead C (this is optimum) might be not a solution in G since edges of T_C are not present in G, but then again smear edges of C via shortest paths in G. The solution cost can only goes down since $d_{T_C}(x,y) \geq d_G(x,y)$ i.e. expansiveness of the embedding. But then we are done (using the argument above). Indeed usually we can use EESTF FRT interchangably as long as we can compare OPT_G with C_T.

The above gives $O(\log n)$ approximation: indeed it works for a more general case when we want to connect (t_i, s_i) pairs. Called the generalized steiner tree or steiner forest.

Indeed ~~[illegible]~~ we can obtain constant approximation for both of these 1.38 and $2-\frac{1}{n}$ in order (we see in next sessions).

For directed graphs we can't use this technique, since it is not correct. Indeed the best approximation is n^{ε} for directed steiner tree and there is a hardness of $\Omega(2^{\log^{1-\varepsilon} n})$ for directed generalized steiner tree. (open to obtain polylog)

Group steiner tree: Gives sets $S_1, S_2, \ldots S_p$ of vertices of G, find a tree that connects at least one vertex of S_1, one vertex S_2, ..., one vertex of S_p.

Again use the probabilistic embedding: use this nontrival result;

Thm: Group steiner tree on trees can be approximated within a factor $O(\log^2 n)$.

Again consider OPT, and you can see that it give a solution which is $O(\log n)$ larger in expection on each trees (note that the optimum solution on trees can be embeded in to the original graph with at most the same cost. Thus again we have equation (A) above