Network Design Foundation
Fall 2011
Lecture 8

Instructor: Mohammad T. Hajiaghayi
Scribe: Manish Purohit

October 19, 2011

1 Overview

We have studied distance preserving probabilistic mapping of graph metrics into tree metrics by Bartal and FRT and also studied a similar probabilistic mapping by Racke which preserves the cuts / bottlenecks in the graph. In this lecture we show the equivalence between the FRT and Racke decompositions.

We also look at some other classical problems for which oblivious or incremental algorithms are desirable.

2 Racke $\Leftrightarrow$ FRT decompositions

2.1 Preconditions

Throughout the following discussion, without loss of generality, we assume that the FRT / Racke trees do not contain any steiner nodes. This is true because as shown by Anupam Gupta, any steiner nodes can be removed by increasing the distance between any nodes by at most a factor of 8. Further, once we have such a tree without any steiner nodes, we only need that $d_T(u, v) \geq d_G(u, v)$ to maintain the dominance.

Similarly, we can consider the capacity of an edge to be $c(e_T) = \sum_{e' \in \delta(e_T)} c(e')$ where $\delta(e_T)$ is a cut in the graph defined by the edge e_T.

Thus, in both cases, the structure of the tree is important and not the weights on edges.

Scribe: Manish Purohit
Lecture 8 **Date:** 10/19/2011

2.2 Preliminaries

Recall from the previous lecture, the following functions:

$$m_E : E_T \to E^* \qquad \text{(Maps a tree edge to a path in G)}$$
$$m'_E : E \to E_T^* \qquad \text{(Maps a graph edge to a path in T)}$$
$$m_V : V_T \to V \qquad \text{(Maps a tree vertex to a vertex in G)}$$
$$m'_V : V \to V_T \qquad \text{(Maps a graph vertex to a vertex in T)}$$
$$T_{e',e} = \text{No. of times } e' \text{ appears on } m_E(m'_E(e))$$

We define the length of the edges in T by

$$d_T(e_t) = \sum_{e \in m(e_t)} d_G(e)$$

Now we try to define the stretch (in FRT) and congestion (in Racke) in terms of $T_{e',e}$. The stretch of the mapping is defined as $max_{e\in E} \dfrac{d_T(e)}{d_G(e)}$. Now,

$$d_T(e) = \sum_{e_t \in m'(e)} d_T(e_t) = \sum_{\substack{e' \in \text{unique path in G} \\ \text{corresponding to } e \text{ in } T}} d_G(e')$$
$$\therefore d_T(e) = \sum_{e'} T_{e',e} \cdot d_G(e') \tag{1}$$

The congestion of an edge $e \in E$ is defined as $max_e \dfrac{load_T(e)}{c_G(e)}$. Now,

$$load_T(e) = \sum_{e_T \in E_T : e \in m_E(e_T)} c(e_T)$$

By the definition $c(e_T)$,

$$load_T(e) = \sum_{e_T \in E_T : e \in m_E(e_T)} \sum_{e' \in E : e' \in \delta(e_T)} c(e')$$

Rearranging the summations,

$$load_T(e) = \sum_{e' \in E} \sum_{\substack{e_T \in \text{Unique path}(e') \\ e \in m_E(e_T)}} c(e')$$
$$load_T(e) = \sum_{e' \in E} T_{e,e'} \cdot c(e') \tag{2}$$

2.3 Important Lemmas

Lemma 1 *For every* $\rho \geq 1$ *and every family of trees, there is a probabilistic mapping with distortion at most* ρ *iff for every non-negative co-efficient* α_e*, there is a tree such that*

$$\sum_{e \in E} \alpha_e \frac{d_T(e)}{d_G(e)} \leq \rho \sum_{e \in E} \alpha_e$$

Proof: Consider a zero sum game in which the player MAP chooses an admissible mapping to a tree T, and the player EDGE chooses an edge e. The value of the game for EDGE is the stretch of e in the mapping, and hence EDGE wishes to maximize the stretch whereas MAP wishes to minimize it. A probabilistic mapping, λ, is a randomized strategy for MAP. Choosing nonnegative coefficients α_e (and scaling them so that $\sum_{\alpha_e} = 1$) is a randomized strategy for EDGE.

In such a formulation, let M_{Te} denote the payoff matrix. Now, by the minimax theorem, we have,

$$min_\lambda max_\alpha \sum_T \sum_e \lambda_T \alpha_e M_{Te} = max_\alpha min_\lambda \sum_T \sum_e \lambda_T \alpha_e M_{Te}$$

where λ_T denotes the probability of choosing tree T and α_e denotes the probability of choosing and edge e. The Lemma 1 follows as a direct application of the above minimax theorem and is left as an exercise. ∎

Lemma 2 *For every* $\rho \geq 1$ *and every family of trees, there is a probabilistic mapping with congestion at most* ρ *iff for every non-negative co-efficient* β_e*, there is a tree such that*

$$\sum_{e \in E} \beta_e \frac{load_T(e)}{c_G(e)} \leq \rho \sum_{e \in E} \beta_e$$

Proof: The proof of Lemma 2 is similar to the proof of Lemma 1. ∎

2.4 Main Theorem

Theorem 1 *For every* $\rho \geq 1$ *and family of trees, statements (1) and (2) are equivalent.*
(1) For every collection of lengths $d_G(e)$ *there is a probabilistic mapping with distortion at most* ρ*.*
(2) For every collection of capacities $c_G(e)$ *there is a probabilistic mapping with congestion at most* ρ*.*

Proof: We prove that $(1) \Rightarrow (2)$.
Assume that there is a probabilistic mapping from $G = (V, E)$ having stretch at most ρ. Therefore, from Lemma 1, we have,

$$\sum_{e \in E} \alpha_e \frac{d_T(e)}{d_G(e)} \leq \rho \sum_{e \in E} \alpha_e$$

Substituting from eq. (1),

$$\sum_{e',e} \alpha_e T_{e',e} \frac{d_G(e')}{d_G(e)} \leq \rho \sum_{e \in E} \alpha_e$$

Put $\alpha_e = \beta_e$ and $d_G(e') = \dfrac{\beta_{e'}}{c_G(e')}$,

$$\therefore \sum_{e',e} (\beta_e)(T_{e',e}) \frac{\left(\dfrac{\beta_{e'}}{c_G(e')}\right)}{\left(\dfrac{\beta_e}{c_G(e)}\right)} \leq \rho \sum_{e \in E} \beta_e$$

$$\therefore \sum_{e',e} (\beta_{e'})(T_{e',e}) \frac{c_G(e)}{c_G(e')} \leq \rho \sum_{e \in E} \beta_e$$

Substituting from eq. (2),

$$\sum_{e' \in E} \beta_{e'} \frac{load_T(e')}{c_G(e')} \leq \rho \sum_{e \in E} \beta_e$$

$$\therefore \sum_{e \in E} \beta_e \frac{load_T(e)}{c_G(e)} \leq \rho \sum_{e \in E} \beta_e$$

Therefore, from Lemma 2 there exists a probabilistic mapping with congestion at most ρ.
Similarly, it can be proved that $(2) \Rightarrow (1)$ and hence the two statements are equivalent. ■

3 Minimum Bisection Problem

Given a graph $G = (V, E)$ with $|V| = 2k$, the MINIMUM BISECTION PROBLEM is to partition the graph into two equal parts each having k vertices so that the width of the cut is minimized, where width is the total capacity of all the edges in the cut.

3.1 Algorithm

1. Find a probabilistic mapping into spanning trees with congestion at most δ. By using results from Abraham et al. and above reduction, $\delta = \tilde{O}(\log n)$. Note that number of such trees in support of the distribution is $O(n \log n)$.
2. In each spanning tree, find the optimal bisection using dynamic programming by taking the capacity of each edge to be equal to its load.
3. Of all the bisections found above, choose the one which has the smallest width in G. By the dominance property, the width of the solution is at most $\tilde{O}(\log n) OPT$.

3.2 Analysis

Let OPT be the width of the minimum bisection in the graph. Let OPT_t be the mapping of this optimal bisection on the subtree t i.e. it contains those edges in OPT which are present in t only the capacity of each edge is now taken to be its load.
Now to due to the probabilistic decomposition, we have

$$\tilde{O}(\log n)\mathsf{OPT} \geq \sum_t P_t \mathsf{OPT}_t$$

Now, let C_t be the width of the optimal bisection in tree t. Therefore, we get

$$\tilde{O}(\log n)\mathsf{OPT} \geq \sum_t P_t C_t$$

Let C^* be the width of the best solution among all trees. Therefore, we get

$$\begin{aligned}
\tilde{O}(\log n)\mathsf{OPT} &\geq \sum_t P_t C^* \\
&\geq C^* \sum_t P_t \\
&\geq C^* \\
\therefore C^* &\leq \tilde{O}(\log n)\mathsf{OPT}
\end{aligned}$$

Hence we have a $\tilde{O}(\log n)$ approximation.

4 Other problems with oblivious / incremental algorithms

4.1 Universal Travelling Salesman Problem

Given an undirected graph $G = (V, E)$ with lengths on edges, the UNIVERSAL TRAVELLING SALESMAN PROBLEM is to provide an universal ordering of nodes, so that given any subset of nodes, one can visit all the nodes in the subset by travelling the least distance.

For planar graphs, Platzman and Bartholdi (in JACM, 1989) gave an algorithm with $O(\log n)$ competitive ratio. In 2006, HKL proved an $\Omega\left(\sqrt[6]{\frac{\log n}{\log \log n}}\right)$ lower bound on planar graphs.

For general graphs, the best known bounds are $O(\log^2 n)$ (by GHR, in 2006) and $\Omega(\log n)$ (by GKSS, in 2010).

Scribe: Manish Purohit
Lecture 8 **Date:** 10/19/2011

4.2 Oblivious Network Design

In 2006, GHR studied the problem of Oblivious Network Design which deals with routing multicommodity flows between multiple (source,sink) pairs with minimum cost in an oblivious manner, i.e., without knowing the demands between other pairs and also without knowing the precise cost function.

An algorithm with $O(\log^2 n)$ competitive ratio is introduced by GHR wherein it is assumed that the demands between all pairs are uniform.

4.3 Incremental Algorithms

For a number of problems such as k-median, k-MST and k-Set Cover, incremental algorithms are known which are only a constant factor worse than the corresponding offline algorithms. For a number of practical applications, incremental algorithms are essential as it is not practically feasible to undo previously made decisions.