15-859(J): Metric Embeddings and Methods **Lecturer:** R. Ravi
Topic: Embeddings and Approximation Algorithms **Date:** Sept 4, 2003
Scribe: Hubert Chan

1 Introduction

The aim of the lecture is to show how metric methods can be exploited to give approximate solutions to some NP-hard problems. We investigate three themes in such applications:

1. **Metric Data**
The input data for the given data is a metric or can be pre-processed to form a metric. The idea is to embed the metric into (a combination of) simpler metrics, on which the problem can be solved more easily.

2. **Metric Relaxations**
The given problem is formulated as a mathematical program which can be relaxed such that an optimal solution can be viewed as a metric. Rounding techniques based on embeddings can give rise to approximate solutions.

3. **Problems on Metrics**
Those are the problems in which metrics are the objects of study. For example, given an arbitrary metric, the goal is to find a tree metric that is closest (in some sense) to it. This has applications in building evolutionary trees in computational molecular biology.

We give an example for each theme and show how solutions can be obtained via metric methods.

2 Metric Data

The example we will use to illustrate this paradigm is the Buy at Bulk Network Design Problem (BBNDP) (see the papers by Salman et al. [6], and by Awerbuch and Azar [1] for more details).

2.1 The Buy at Bulk Network Design Problem

Suppose $G = (V, E)$ is a connected undirected graph and $l : E \to \mathbb{R}_+$ is a length function defined on the edges of the graph. We can view the graph as a network in which vertices communicate with one another through cables installed on the edges. There is a demand function $dem : V \times V \to \mathbb{Z}_+$ that specifies the amount of communication between any pair of vertices. We are given k types of cables such that cable type i has capacity $u_i > 0$ and cost per unit length $c_i > 0$. Moreover, for $1 \leq i < k$, $u_i \leq u_{i+1}$, $c_i \leq c_{i+1}$ and $\frac{c_{i+1}}{u_{i+1}} \leq \frac{c_i}{u_i}$ [1].

[1]Otherwise, cable type-$(i+1)$ can be substituted by type-i. This is the property reflecting economies of scale, leading to the "buy-at-bulk" in the name of the problem.

A solution to the problem consists of specifying, for every pair of vertices (u, v), a path P_{uv} from u to v in G. Moreover, we have to assign for each edge e and each i, the number of type-i cables, n_e^i, to be installed on edge e. Note that the cables installed on an edge have to satisfy the demand going through it in both directions, i.e. for each edge e,

$$\sum_{(u,v):e \in P_{uv}} dem(u,v) \leq \sum_{i=1}^{k} n_e^i u_i. \tag{2.1}$$

The cost of the solution is:

$$\sum_{e \in E} (\sum_{i=1}^{k} n_e^i c_i) l_e,$$

which we aim to minimize.

Remark 2.1 Note that the length function $l : E \to \mathbb{R}_+$ naturally induces a metric $dist$ on V, where $dist(u, v)$ is the length of a shortest path from u to v. The metric $dist$ is also known as the *metric completion* of the length function l. Observe that if $l(u, v) > dist(u, v)$, no cables would be installed on edge (u, v) in an optimal solution. Suppose some cable is installed on edge (u, v) in a solution. Then, a solution with strictly smaller cost can be obtained by removing that cable on (u, v) and installing the same type of cable along a shortest path from u to v. Hence, without loss of generality, we assume from now on $dist(u, v) = l(e)$, for every edge $e = (u, v)$.

2.2 Special Instances of BBNDP

Several classic problems can be formulated as special instances of BBNDP.

1. **Minimum Spanning Tree Problem**
   Pick any $v_0 \in V$. For each $v \in V \setminus \{v_0\}$, set $dem(v, v_0) := 1$; set the demand for all other pairs of vertices to be 0. We only have one type of cable, each of which has very large capacity and cost 1 per unit length. We can retrieve from an optimal solution of this instance of BBNDP a minimum spanning tree under the length function l, namely the set of edges on which a non-zero number of cables are installed.

2. **Shortest Path Tree Problem**
   Suppose we want to find a shortest path tree rooted at some vertex v_0. We assign the same demand function as above and provide only one type of cables, which also has cost 1 per unit length. The only difference is that each cable has capacity 1. The optimal solution for this instance of BBNDP gives a shortest path tree rooted at v_0.

3. **Steiner Tree Problem**
   Given a graph $G = (V, E)$, a set of required vertices $R \subseteq V$, and a length function $l : E \to \mathbb{R}_+$, the goal is to find a minimum length tree in G that spans R (i.e. the vertices in R are connected to one another in the tree). The Steiner Tree Problem can be formulated as an instance of

BBNDP. We only use one type of cable, which is the same as the one for the minimum spanning tree problem. The demand function is modified in the following way. Pick any $v_0 \in R$. For each $v \in R \setminus \{v_0\}$, set $dem(v, v_0) := 1$; set all other demands to be 0.

4. **Generalized Steiner Forest Problem**
   Instead of a set of required vertices, we are given a list of pairs of vertices $\{(s_j, t_j)\}_{j=1}^p$. The goal is to find a minimum length subgraph in which for each i, s_i is connected to t_i. In the formulation of the problem, we use the same infinite-capacity-unit-cost cable, and the only pairs of vertices that have non-zero demands are exactly $\{(s_j, t_j)\}_{j=1}^p$, each of which has unit demand.

Since the last two problems are NP-hard, we can conclude that BBNDP is also NP-hard.

2.3 BBNDP on Trees

Suppose the given graph G is a tree. Then, for every pair of vertices (u, v), there exists a unique (simple) path from u to v in G. Hence, we set P_{uv} to be this unique path. It remains to install cables to satisfy the demand.

We define the *aggregate demand* (with respect to the path assignment P_{uv}'s) on an edge e to be $f_e := \sum_{(u,v):e \in P_{uv}} dem(u, v)$. Note that this is the quantity on the left hand side of the inequality (2.1).

For every non-negative integer m, define the minimum cost per unit length to install cables to satisfy m units of aggregate demand on an edge to be [2]:

$$c(m) := \min\{\sum_{i=1}^{k} n^i c_i : \sum_{i=1}^{k} n^i u_i \geq m, \quad n^i \in \mathbb{Z}_+,\ i = 1, \dots, k\}. \tag{2.2}$$

It follows that the cost of an optimal solution is $\sum_{e \in E} c(f_e) l(e)$. However, in general, computing the function c is NP-hard, since this is an instance of the Min-Knapsack Problem. Fortunately, there exists a PTAS for approximating the function c. We give a simple heuristic to obtain a 2-approximation.

Theorem 2.2 *Let m be a non-negative integer. Suppose i_0 is the type of cable that attains $\min_i \left\lceil \frac{m}{u_i} \right\rceil c_i$. Then, using only cable type i_0 to satisfy m units of aggregate demand gives a 2-approximate solution for estimating $c(m)$, i.e., $\left\lceil \frac{m}{u_{i_0}} \right\rceil c_{i_0} \leq 2c(m)$.*

Proof: Suppose $c(m)$ is attained by installing n^i copies of cable type i for each $i = 1, \dots, k$. Let $I := \{i : n^i > 0\}$. If $|I| = 1$, then only one type of cable is installed and so the simple heuristic attains $c(m)$. We assume $|I| \geq 2$.

Let $\hat{i} := \max I$. Note that there exists $m_i > 0$ for each $i \in I$ such that $n_i = \left\lceil \frac{m_i}{u_i} \right\rceil$ and $\sum_{i \in I} m_i = m$.

[2]Note that for any non-negative integers m and n, $c(m + n) \leq c(m) + c(n)$.

Therefore, we have

$$c(m) = \sum_{i \in I} \left\lceil \frac{m_i}{u_i} \right\rceil c_i \geq \sum_{i \in I} \frac{c_i}{u_i} \cdot m_i \geq \sum_{i \in I} \frac{c_{\hat{i}}}{u_{\hat{i}}} \cdot m_i = \frac{c_{\hat{i}}}{u_{\hat{i}}} \cdot m.$$

Notice that we have used the fact that $\{\frac{c_i}{u_i}\}$ is non-increasing. Hence,

$$\left\lceil \frac{m}{u_{i_0}} \right\rceil c_{i_0} \leq \left\lceil \frac{m}{u_{\hat{i}}} \right\rceil c_{\hat{i}} \leq (\frac{m}{u_{\hat{i}}} + 1) c_{\hat{i}} \leq \frac{2m}{u_{\hat{i}}} \cdot c_{\hat{i}} \leq 2c(m).$$

Note that we used $m \geq u_{\hat{i}}$ in the penultimate inequality. Since $|I| > 1$, it follows $m \geq u_{\hat{i}}$; otherwise, we can just use one copy of type $\hat{i}$ cable to achieve a lower cost. ■

Corollary 2.3 *There exists a 2-approximation algorithm for BBNDP on trees with n nodes, that runs in time $O(nk)$, where k is the number of cable types.*

2.4 Approximating BBDNP via Tree Embeddings

Recall Theorem 3.4 from the previous lecture:

Theorem *Any n-point metric $O(\log n)$-probabilistically embeds into a distribution of trees; furthermore, samples from this distribution can be generated in polynomial time.*

Remark 2.4 Note that by [3], the distribution contains $O(n \log n)$ trees, each of which we can assume is a spanning tree of the original graph [5]. Moreover, for each tree T in the distribution and for each edge (u, v) in T, $dist_T(u, v) = dist(u, v)$.

We apply the theorem to our metric $dist$ induced by the length function l to obtain the desired distribution of trees $\mathcal{D}$.

Lemma 2.5 *Suppose we are given a solution to the BBDN problem on some tree T from distribution $\mathcal{D}$ with cost c_T. Then, we can construct a solution to the problem on G (with the same demand function and the same types of cables) with cost c_T in polynomial time.*

Proof: For every edge (x, y) in T, fix some shortest path p_{xy} between x and y in G. For each cable installed on an edge (x, y) in T, we install one copy of the same type of cable on each edge on the path p_{xy} in G. Since the length of the path p_{xy} equals $dist_T(x, y)$, it follows the cost of installing cables in G is c_T.

It remains to specify a path P_{uv} in G for every pair of vertices (u, v). Note that the unique path $\mathcal{P}$ from u to v in T naturally specifies a walk Q_{uv} in G from u to v, namely by concatenating the paths p_{xy} for each (x, y) in $\mathcal{P}$. Observe that the cables installed on G can satisfy all the demands when the demand for each pair (u, v) is routed along the walk Q_{uv}. Hence, we can pick P_{uv} to be a simple path from u to v contained in Q_{uv}. ■

Theorem 2.6 *Suppose T is a tree sampled from distribution $\mathcal{D}$ and $OPT(T)$ is the cost of an optimal solution to BBDNP on T. Suppose further that the cost of an optimal solution to BBDNP on G is OPT. Then,*

$$E_T[OPT(T)] \leq O(\log n) OPT,$$

where expectation is taken over sampling T from distribution $\mathcal{D}$.

Proof: Suppose an optimal solution on G assigns path P_{uv} in G to each pair of vertices (u, v) and installs n_e^i cables of type i on edge e. Hence, $OPT = \sum_{e \in E}(\sum_{i=1}^{k} n_e^i c_i) l(e)$.

Suppose T is a tree sampled from distribution $\mathcal{D}$. We use the optimal solution on G to construct a solution for the problem on T.

For every pair of vertices (u, v), suppose p_{uv} is the unique path between u and v in T. For each cable installed on edge $e = (u, v)$ in G, we install one copy of the same cable on each edge in p_{uv} in T. It is immediate that installing cables on T this way would satisfy all the demands if we route the demand between any two vertices using the unique path between them. The cost of the solution is

$$c_T := \sum_{e \in E}(\sum_{i=1}^{k} n_e^i c_i) dist_T(e).$$

Observe that $OPT(T) \leq c_T$ and for each edge e, $E[dist_T(e)] \leq O(\log n) l(e)$. Hence, by linearity of expectation, we have the following result.

$$\begin{aligned}
E[OPT(T)] &\leq E[c_T] \\
&= E[\sum_{e \in E}(\sum_{i=1}^{k} n_e^i c_i) dist_T(e)] \\
&= \sum_{e \in E}(\sum_{i=1}^{k} n_e^i c_i) E[dist_T(e)] \\
&\leq \sum_{e \in E}(\sum_{i=1}^{k} n_e^i c_i) O(\log n) l(e) \\
&= O(\log n) OPT.
\end{aligned}$$

■

By combining Theorem 3.4 from Lecture 1, Corollary 2.3, Lemma 2.5 and Theorem 2.6, we have:

Theorem 2.7 *There exists a randomized polynomial algorithm for BDDNP that gives $O(\log n)$-approximation in expectation.*

Remark 2.8 Since the number of trees in the distribution is polynomial, we can solve BDDNP (approximately) on each of the trees and output a solution among these of minimum cost. This would give the claimed $O(\log n)$-approximation algorithm.

3 Metric Relaxation

We illustrate how the method of metric relaxation can be used to solve problems involving cuts.

3.1 Metrics and Cuts

Suppose V is some finite set. For $S \subseteq V$, let $\partial(S) := \{\{i,j\} : i \neq j \in V, |S \cap \{i,j\}| = 1\}$.

Definition 3.1 A metric d on V is an elementary cut metric, if there exists a subset $S \subseteq V$ such that for all $i, j \in V$,

$$d_{ij} = \begin{cases} 1, & \{i,j\} \in \partial S; \\ 0, & \text{otherwise.} \end{cases}$$

Definition 3.2 A metric d on V is a cut metric, if there exists $y : 2^V \to \mathbb{R}_+$ such that for all $i, j \in V$,

$$d_{ij} = \sum_{S \subseteq V : \{i,j\} \in \partial S} y(S).$$

In other words, the set of cut metrics is exactly the set of non-negative linear combinations of elementary cut metrics.

3.2 Solving Min-Cut via Metric Relaxation

Given a connected undirected graph $G = (V, E)$, a capacity function $c : E \to \mathbb{R}_+$, and two distinct vertices s and t, the Min-s-t-Cut Problem asks for a minimum capacity cut that separates s and t.

In the language of cut metrics, the capacity of an s-t minimum cut is:

$$\min\{\sum_{e \in E} c_e d_e : d \text{ is an elementary cut metric such that } d_{st} = 1\}$$

We shall relax the condition and only require d to be a metric such that $d_{st} \geq 1$. This can be formulated as a linear program.

$$\begin{aligned} \min \sum_{e \in E} c_e d_e \\ \text{subject to} d_{ij} \leq d_{ik} + d_{kj}, \forall i, j, k \\ d_{st} \geq 1 \\ d_{ij} \geq 0, \forall i, j. \end{aligned} \tag{3.3}$$

Suppose d^* is an optimal solution to (3.3). If d^* is fractional, we need to round the solution in order to obtain a cut. We order the vertices in V into level sets based on their distances from s in d^* (Figure 3.1). Suppose $\{x_0, x_1, \dots, x_l\} := \{d^*(s,v) : v \in V, d^*(s,v) \leq d^*(s,t)\}$, where $\{x_j\}$ is strictly increasing. It follows $x_0 = 0$ and $x_l = d^*(s,t)$.

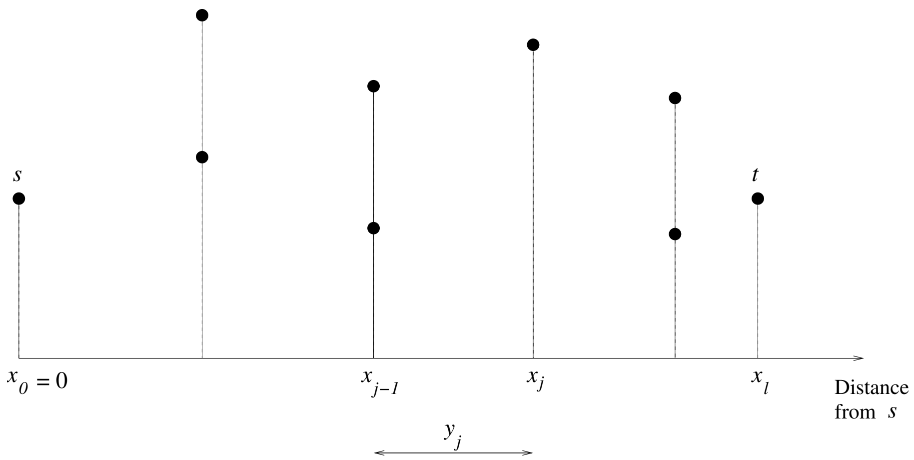

Figure 3.1: Ordering vertices into level sets

For each $j = 1, \dots, l$, define

$$E_j := \{(u,v) \in E : d^*(s,u) \le x_{j-1}, d^*(s,v) \ge x_j\},$$

$$y_j := x_j - x_{j-1},$$

and

$$C_j := \sum_{e \in E_j} c_e.$$

Lemma 3.3 *For each edge $e \in E$,*

$$d_e^* \ge \sum_{j : e \in E_j} y_j.$$

Proof: By the triangle inequality, we have for each edge $e = (u,v) \in E$,

$$d_e^* \ge |d^*(s,u) - d^*(s,v)| \ge \sum_{j : e \in E_j} y_j$$

■

Theorem 3.4 *Suppose $C^* = \min_j C_j$ is attained by some j_0. Then, E_{j_0} is an s-t minimum cut.*

Proof: Suppose Z^* is the capacity of a minimum cut. Note that since Z^* is the value of an optimal integral solution, it can be larger than the value of an optimal solution to the LP. Using

Lemma 3.3, we have

$$
\begin{aligned}
Z^* \geq \sum_{e\in E} c_e d_e^* \quad & \geq \sum_{e\in E} c_e \sum_{j:e\in E_j} y_j \\
= \sum_{j=1}^{l} \sum_{e\in E_j} c_e y_j \quad & = \sum_{j=1}^{l} C_j y_j \\
\geq \sum_{j=1}^{l} C^* y_j \quad & = C^*(x_l - x_0) \\
= C^* d^*(s,t) \quad & \geq C^*.
\end{aligned}
$$

■

In retrospect, note that we have provided a simple embedding of d^* into ℓ_1^1 that preserves the distance between s and t in d^*.

3.3 Extension to Multi-way Cut Problem

Given a connected undirected graph $G = (V, E)$, a capacity function $c : E \to \mathbb{R}_+$ and a set of vertices $\{s_r\}_{r=1}^k$, the goal of the multi-way cut problem is to find a minimum capacity set of edges whose removal separates the s_i's from one another. Unlike the Min-s-t-Cut Problem, the Multi-way Cut Problem is NP-hard. We reinterpret a simple 2-approximation [4] via a metric relaxation.

The problem can be formulated as an integer programming, which has the following LP relaxation:

$$
\begin{aligned}
\min \quad & \textstyle\sum_{e\in E} c_e d_e & & (3.4)\\
\text{subject to} \quad & d_{ij} \leq d_{ih} + d_{hj}, & \forall i, j, h \quad & (3.5)\\
& d_{s_p s_q} \geq 1, & \forall p \neq q \quad & (3.6)\\
& d_{ij} \geq 0, & \forall i, j. \quad & (3.7)
\end{aligned}
$$

Suppose d^* is an optimal solution to the LP. In general, d^* is fractional, and we use a similar rounding technique as in Theorem 3.4. Define, for $r := 1, \ldots, k$,

$$B_r := \{v \in V : d^*(s_r, v) < \frac{1}{2}\}.$$

Because of the constraint (3.6), the B_r's are disjoint. We also define, in a similar way as before,

$$\{x_0^r, x_1^r, \ldots, x_{l_r-1}^r\} := \{d^*(s_r, v) : v \in V,\, d^*(s, v) < \frac{1}{2}\},$$

where $\{x_j^r\}$ is a strictly increasing sequence. It follows that $x_0^r = 0$; let us define $x_{l_r}^r := \frac{1}{2}$.

For each $j = 1, \ldots, l_r$, define

$$E_j^r := \{(u, v) \in E : d^*(s_r, u) \leq x_{j-1}^r, d^*(s_r, v) \geq x_j^r\},$$

$$y_j^r := x_j^r - x_{j-1}^r,$$

and the *capacity* of the cut corresponding to the j-th level cut

$$C_j^r := \sum_{e \in E_j^r} c_e.$$

Lemma 3.5 *For each edge $e \in E$,*

$$d_e^* \geq \sum_{r=1}^{k} \sum_{j : e \in E_j^r} y_j^r.$$

Proof: Note that $e = (u, v) \in E_j^r$ for some j only if at least one of u and v belongs to B_r. Recall that the B_r's are disjoint.

Suppose both u and v belong to some B_r. Then, the result follows from Lemma 3.3. Else, suppose $u \in B_r$ and $v \in B_{r'}$, where $r \neq r'$. Since $d^*(s_r, s_{r'}) \geq 1$, it follows that

$$d_e \geq (\frac{1}{2} - d^*(s_r, u)) + (\frac{1}{2} - d^*(s_{r'}, v)) \geq \sum_{j : e \in E_j^r} y_j^r + \sum_{j : e \in E_j^{r'}} y_j^{r'}.$$

Note that this is the lower bound we seek to prove, since e is not in any $E_j^{r''}$, for $r'' \neq r, r'$. ■

Theorem 3.6 *Suppose for each $r = 1, \ldots, k$, $C_r^* = \min_j C_j^r$ is attained by some j_0^r. Then, the union of the lightest $k-1$ of the $E_{j_0^r}^r$'s gives a $2(1 - \frac{1}{k})$-approximation to the multi-way cut problem.*

Proof: Suppose Z^* is the cost of an optimal solution for the multi-way cut problem. Using Lemma 3.5 as before, we have

$$\begin{aligned}
Z^* &\geq \sum_{e \in E} c_e d_e^* \qquad \geq \sum_{e \in E} c_e \sum_{r=1}^{k} \sum_{j : e \in E_j^r} y_j^r \\
&= \sum_{r=1}^{k} \sum_{j=1}^{l_r} \sum_{e \in E_j^r} c_e y_j^r \qquad = \sum_{r=1}^{k} \sum_{j=1}^{l_r} C_j^r y_j \\
&\geq \sum_{r=1}^{k} \sum_{j=1}^{l_r} C_r^* y_j \qquad = \sum_{r=1}^{k} C_r^* (x_{l_r}^r - x_0^r) \\
&= \frac{1}{2} \sum_{r=1}^{k} C_r^*
\end{aligned}$$

It remains to observe that the union of the lightest $k-1$ of the $E_{j_0^r}^r$'s has capacity at most $(1 - \frac{1}{k}) \sum_{r=1}^{k} C_r^*$. ■

Note that the above result can be interpreted as embedding d^* into a non-negative linear combination of elementary cut metrics (each cut represented by E_j^r), which is dominated by d^* and preserves the distance in d^* between every pair (s_p, s_q).

A more careful metric relaxation leading to an improved $\frac{3}{2}$-approximation can be found in [2].

References

[1] Baruch Awerbuch and Yossi Azar. Buy-at-bulk network design. In *IEEE Symposium on Foundations of Computer Science*, pages 542–547, 1997.

[2] Gruia Călinescu, Howard Karloff, and Yuval Rabani. An improved approximation algorithm for MULTIWAY CUT. *J. Comput. System Sci.*, 60(3):564–574, 2000.

[3] Moses Charikar, Chandra Chekuri, Ashish Goel, Sudipto Guha, and Serge A. Plotkin. Approximating a finite metric by a small number of tree metrics. In *IEEE Symposium on Foundations of Computer Science*, pages 379–388, 1998.

[4] Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiterminal cuts. *SIAM J. Comput.*, 23(4):864–894, 1994.

[5] Anupam Gupta. Steiner points in tree metrics don't (really) help. *SODA*, pages 220–227, 2001.

[6] F. S. Salman, Joseph Cheriyan, R. Ravi, and S. Subramanian. Buy-at-bulk network design. In *SODA Buy-at-Bulk Network Design: Approximating the Single-Sink Edge Installation Problem.*, pages 619–628, 1997.