Advanced Approximation Algorithms **(CMU 15-854B, Spring 2008)**

Lecture 13: Int. Gap and Hardness of Priority Steiner Tree

February 26, 2008

Lecturer: Anupam Gupta *Scribe: Evan Danaher*

1 Integrality Gap

1.1 LP Relaxation

The LP relaxation for the Priority Steiner Tree is similar to the relaxation used before for the Steiner Tree. Again, say a set $S \subseteq V$ crosses the set R if $S \cap R \neq \emptyset$ and $R \setminus S \neq \emptyset$ - there is at least one terminal in S and one not in S. For $j \leq k$, let $\mathcal{S}_j$ be the collection of all sets which cross R_j. Finally, for a set $S \subseteq V$, let ∂S be the set of edges with one endpoint in S and one in $V \setminus S$.

Define $E_{\leq j}$ to be $\bigcup_{k=1}^{j} E_j$, the set of edges of priority j or higher; the set of edges available to vertices of level j.

The LP relaxation of Priority Steiner Tree then has a variable for each edge, representing the extent to which it is used by the fractional solution. The goal is to minimize the cost of edges used, while ensuring that every vertex in any R_j is connected to the root using only edges in $E_{\leq j}$; this can be formulated in terms of crossing sets, giving the following LP relaxation:

$$\min \sum_e c_e x_e$$

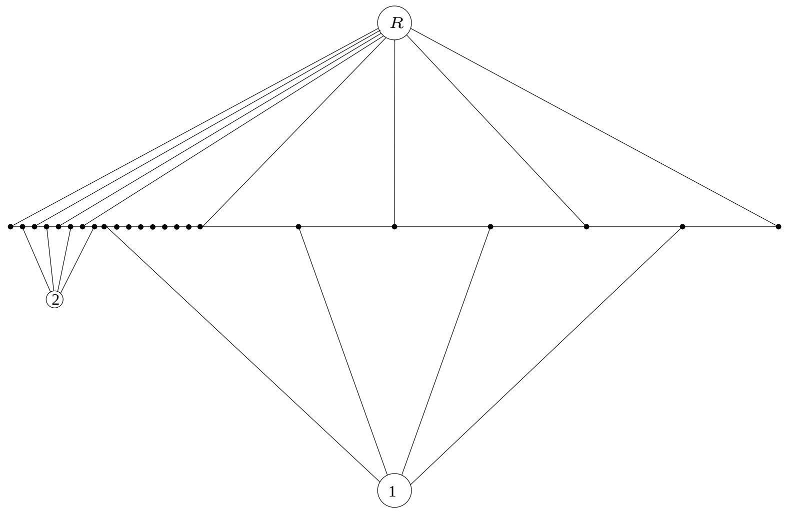
$$\text{subject to} \sum_{e \in \delta S \cap E_{\leq j}} x_e \geq 1 \qquad \forall j \leq k \forall S \in \mathcal{S}_j$$

1.2 Integrality Gap Instance

The gap instance is suggested by the following picture:

All edges except for the center line have weight 0, The center line is initially divided into $2z$ edges, where z is a constant to be determined. The root vertex connects to every other one of these (including the ends) by level 1 edges with cost 0, and the remaining vertices are connected by level 1 cost 0 edges to the sole level 1 vertex.

This basic structure is repeated recursively: each of the $2z$ segments is split into $2z$ subsegments, (so $2z-1$ new vertices are added to the segment) with an associated level 2 vertex. Every other new vertex is connected to the root by a cost 0 level 2 edge, with the remaining vertices connected to the associated level 2 vertex by cost 0 level 2 edges, resulting in $2z$ new level 2 vertices and $2z(2z-1)$ new vertices along the center line. This pattern is repeated up to level k, resulting in $(2z)^{k-1}$ vertices of level k, along with $(2z)^{k-1}z$ level k cost 0 edges connecting them to the center line, and $(2z)^{k-1}(z-1)$ level k cost 0 edges connecting the center line to the root.



Finally, the $(2z)^k$ edges along the center line are each given cost $1/(2z)^k$ and level k, so that the entire line has cost 1 and is available to every vertex.

1.3 Gap

The integral solution is fairly straightforward: first, note that the level 1 vertex must follow some edge up to the center line, then reach one of the level 1 edges to the root, which requires traversing edges of at total cost $1/(2z)$. Each level 2 vertex must likewise go up to the center line, then along the center line to a level 2 edge to the root. The level 2 edge connected to the segment used by the level 1 vertex already has been used, so this vertex costs no extra. The remaining $2z-1$ level 2 vertices each incur a cost of $1/(2z)^2$ to reach a level 2 edge traveling to the root. And clearly, at any level j, each of the $(2z-1)^{j-1}$ vertices not covered by previous paths each incur a cost of $1/(2z)^j$.

As z gets large, the total cost then tends towards

$$\sum_{j=1}^{k} \frac{(2z-1)^{j-1}}{(2z)^j} \approx \sum_{j=1}^{k} \frac{1}{(2z)} = \frac{k}{2z}$$

In particular, if we define $z = k$, then this is a constant.

Now consider the cost of a fractional solution which avoids choosing a particular path to fully invest in at each level, but rather fractionally takes each center edge by $1/z = 1/k$. At each level, each vertex has z possible paths it can follow, each of which is $1/z$ fractionally chosen, so it overall has a complete path to the root. Thus, this is a solution to the LP relaxation with total cost $1/k$ (since the sum of costs of all the center edges is 1).

Thus, the integrality gap for this instance is k. The size of the problem is $O((2k)^k)$, so for a given size n, the gap obtainable by an instance of size n is roughly $\log\log n/logn$.

2 Hardness

The reduction is from a special collection of Set-Cover instances that can be split into two groups: YES instances have a solution of size X, and any attempted covering of a NO instance using at most hX sets leaves at least a $2^{c \cdot h}$ fraction uncovered. Further, every instance in this collection has universe size u, and s sets each of size z.

2.1 Reduction

The reduction from Set-Cover to PST is similar to that used for the integrality gap. This time, there are s center lines, one corresponding to each set. Each line is broken into $2z$ segments, and every other vertex along the lines is connected to the root with a cost 0 level 1 edge. There are u level 1 vertices, corresponding to the ground elements. For each center line, the z vertices along not connected to the root are connected by cost 0 level 1 edges to the z level 1 vertices corresponding to the elements of the corresponding set. Thus, each level 1 vertex has degree equal to the number of sets is corresponding element is in.

This structure is then repeated recursively: each segment of each centerline is split into $2z$ subsegments, and $(2z)u$ level 2 vertices are added, corresponding to $2z$ copies of the ground elements. Each copy is associated with a collection of segments of the center lines: one copy corresponds to the first segment of each line, one to the second, and so on. These level 2 vertices are then connected to the vertices of their associated subsegments according to containment of their corresponding elements in the corresponding sets, as before.

This pattern is repeated k times, resulting in $u \cdot (2z)^{k-1}$ level k vertices. The center edges are again level k with cost $1/(2z)^k$, so that the each center line has total cost 1.

Claim 2.1 (Soundness)**.** *A YES Set-Cover instance is mapped to a PST instance with* $\mathrm{Opt} \leq X$.

Proof. As a YES instance, there is a set cover solution with at most X sets. For each set in this solution, buy the entire center line corresponding to that set.

This solution is feasible, since every vertex needing to be connected to the root corresponds to some element in the set cover; this element is in some set in the cover, and the corresponding center line is available in the PST solution. So the vertex can take the free edge to that center line, then travel along the available center line to an edge leading to the root.

The solution has cost at most X, since each center line has total cost 1, and at most X were bought. Thus, the optimum for this instance is at most X. ☐

To establish completeness, we use the notion of a *minimal solution*: one for which any vertex goes a single step along a center line. It is clear that any optimal solution is of this form, since additional steps along a center line can be eliminated; if they were needed for later steps, then those steps can add them back in as necessary. We may thus assume that all solutions under discussion use only minimal paths.

Further, suppose that each center edge is duplicated at each level, and bought at the first level where that edge is used. This does not affect the cost of the solution, since a single copy of each needed edge is bought, but it allows us to talk about the level at which an edge is bought, and the cost incurred by edges bought at each level.

Claim 2.2. *For a NO Set-Cover instance, if the total cost incurred by edges at levels less than j is at most $h/2 \cdot X$, then level j incurs a "large" cost $\Theta(u \cdot 2^{-c \cdot h}/z)$.*

Proof. Suppose we are considering paths of level j. Any previous paths bought an edge along the center at its level, which now corresponds to an entire set at level j.

The total cost at higher values is at most $h/2 \cdot X$ by assumption. There are $(2z)^{j-1}$ subinstances corresponding to the original set cover, and by Markov's inequality, at least $1/2$ of them have at most $2(h/2 \cdot X) = h \cdot X$ paths bought by lower levels. Because the Set-Cover instance was a NO instance, this lets us conclude that these $1/2$ of the subinstances have at least a $2^{-c \cdot h}$ fraction of elements uncovered, corresponding to level j vertices which need to add new edges each of cost $1/(2z)^j$ to connect to the root. The cost at level j is thus

$$\frac{(2z)^{j-1}}{2} \cdot (2^{c \cdot h} u) \cdot \left(\frac{1}{2z}\right)^j = \frac{u \cdot 2^{-c \cdot h}}{4z}$$

□

Now suppose that the final cost of the PST solution was at most $h/2 \cdot X$. Then each level must have incurred the large cost. In particular, if this happens with $k \geq 4hXz2^{c \cdot h}/u)$, then $ku \cdot 2^{-c \cdot h}/z \geq h/2 \cdot X$, contradicting the assumption that the final cost was at most $h/2 \cdot X$. Thus, in this case the total cost cannot be less than $h/2 \cdot X$, so we have a gap of $h/2$ between the YES and NO instances.

2.2 The Set-Cover Instance and Conclusion

The special Set-Cover instance needed above can be constructed as a reduction from SAT to Label-Cover to Set-Cover. Starting from a SAT instance of size n, construct a Label-Cover instance with n nodes in each set, each node of degree d, label set L and key set K with $n = n'^{-\log \eta}$, and $d, |L|, |K|$ all $\text{poly}(1/\eta)$.

This Label-Cover instance reduces to a Set-Cover instance with universe size $u = nd2^{|K|}$, $s = n|L|$ sets of size $z = d2^{|K|-1}$, and optimal solution $X = 2n$, with the h-factor property being satisfied if $1/h^2 > O(\eta)$.

Thus, to get the h-factor property, we need $h^2 < O(1/\eta)$, so $\eta \approx h \approx \log \log n$.

The size of the construction is $N \approx s(2z)^k = n^{-\log \eta} \cdot \text{poly}(1/\eta)^{2^h} \approx n^{\log \log \log n}$, and the gap between YES and NO instances is $h = \Omega(\log \log n) = \Omega(\log \log N)$, so we have proven that PST is hard to approximate to better than $\Omega(\log \log n)$.

References

[1] J. Chuzhoy, A. Gupta, J. Naor, and A. Sinha. On the approximability of some network design problems. *SODA*, V(N):1–19, 2007.