Lecture 25 (Nov 3 & 5): Group Steiner Tree

Lecturer: Zachary Friggstad *Scribe: Zachary Friggstad*

25.1 Group Steiner Tree

In this problem, we are given a graph $G = (V, E)$ with edge distances $d(e), e \in E$, a root node $r \in V$, and subsets of *terminal groups* $X_1, \ldots, X_k \subseteq V$. The goal is to find the cheapest $F \subseteq E$ such that in the graph (V, F), r is connected to at least one node in each terminal group. When exach X_t consists of a single node, this is just the usual Steiner Tree problem.

Throughout, we let $n = |V|$ and $g = \max_{1 \leq t \leq k} |X_t|$.

As with Steiner Tree, we can (but do not always) assume that G is a metric by considering the shortest path metric of the original graph. However, Group Steiner Tree is harder to approximate.

Theorem 1 *Unless P = NP, there is no $o(\log k)$ or $o(\log n)$ approximation for* Group Steiner Tree *even if G is a star.*

Proof. We reduce from an instance $(X, \mathcal{S})$ of Set Cover with set costs $c(S) \geq 0, S \in \mathcal{S}$. Let $G = (\mathcal{S} \cup \{r\}, E)$ where r is a new node and $E = \{(S, r) : S \in X\}$ where $d(S, r)$ has distance $c(S)$. For every $i \in X$, let $X_i = \{S : i \in S\}$ be a terminal group. The correspondence between feasible solutions to the Set Cover and Group Steiner Tree instances is immediate.

Since there is no $o(\log |X|)$-approximation for Set Cover unless P = NP, we see that it is also NP-hard to approximate Group Steiner Tree within $o(\log n)$ or $o(\log k)$. ■

In fact, more can be said:

Theorem 2 *[[HK03]] Unless $NP \subseteq DTIME(n^{O(\log \log n)}$, there is no $O(\log^{2-\epsilon} n)$-approximation for* Group Steiner Tree *even if G is a tree for any constant $\epsilon > 0$.*

We will essentially match this lower bound in trees.

Theorem 3 *There is a randomized polynomial-time algorithm that finds a feasible solution for a* Group Steiner Tree *instance in trees whose cost is $O(\log g \cdot \log k) \cdot OPT_{LP}$ with probability at least 1/2. Here, the LP relaxation is given below as (**LP-GST**).*

$$
\begin{aligned}
\text{minimize :} \quad & \sum_{e \in E} d(e) \cdot x_e \\
\text{subject to :} \quad & \sum_{e \in \delta(S)} x_e \geq 1 \quad \text{for each } S \subseteq V - \{r\} \text{ such that } X_t \subseteq S \text{ for some } 1 \leq t \leq k \qquad \textbf{(LP-GST)} \\
& \mathbf{x} \geq 0
\end{aligned}
$$

Using probabilistic embeddings of metrics into tree metrics and the fact that Group Steiner Tree in general graphs reduces to Group Steiner Tree in metrics, this leads to the following bound.

Theorem 4 *The integrality gap of (**LP-GST**) in general graphs is at most $O(\log n \cdot \log g \cdot \log k)$.*

Naturally, since the rounding algorithm for trees runs in polynomial time then this leads to a polynomial-time randomized $O(\log n \cdot \log g \cdot \log k)$-approximation. There are some low-level details to fill in to properly show how the gap bound for trees yields the gap bound in general metrics. The current assignment is asking you to describe them.

For the rest of these notes, we focus on proving Theorem 3.

25.2 Group Steiner Tree in Trees

Suppose that G is a tree, which we now call $T = (V, E)$. Let x^* be an optimal solution to (**LP-GST**) in this instance. We suppose that $c(e) > 0$ for each $e \in E$ and that each node in each terminal group is a leaf in T (this will simplify our discussion, and it is also true for the tree metrics obtained using the tree embedding algorithm we discussed earlier).

Let E_i denote all edges at depth i of the tree (E_1 are the edges incident to the root r) and say that h is the height of T. Finally, for any edge $e \in E$ such that $e \notin E_1$, let $p(e) \in E_{i-1}$ be the parent edge of e.

Claim 1 *In any optimal solution x^* to (**LP-GST**), $x^*_e \leq x^*_{p(e)}$ for each $e \in E - E_1$ and $x^*_e \leq 1$ for each $e \in E$.*

Proof. Consider any terminal group X_t. By the max-flow/min-cut theorem and the constraints of (**LP-GST**), we can send one unit of flow f^t from the nodes in X_t to r such that $f^t_e \leq x^*_e$.

Suppose $x^*_e > x^*_{p(e)}$ where e' is the parent edge of e. We claim that we can reduce x^*_e to $x^*_{p(e)}$ and maintain feasibility. Since all flow originating from nodes in X_t in the subtree under edge e must pass through $p(e)$, then $f^t_e \leq f^*_{p(e)} \leq x^*_{p(e)}$. Thus, reducing x^*_e to $x^*_{p(e)}$ still supports f^t. This holds for every terminal group, so the modified LP solution is still feasible and has no greater cost.

If $x^*_e > 1$ for some edge e, then clearly reducing x^*_e to 1 maintains feasibility. ■

The main subroutine in our algorithm is Algorithm 1, which attempts to connect some terminal groups to r.

Algorithm 1 Main Subroutine for the Group Steiner Tree Rounding Algorithm

$F \leftarrow \emptyset$
for each $e \in E_1$ **do**
  add e to F with probability x^*_e
end for
for $i = 2, \ldots h$ **do**
  for each $e \in E_i$ such that $p(e) \in F$ **do**
    add e to F with probability $x^*_e / x^*_{p(e)}$ (c.f. Claim 1)
  end for
end for
return F

Note that if $x^*_{p(e)} = 0$ for some edge $e \in E - E_1$ then $p(e)$ will not be added to F so we never try to add e to F with probability $x^*_e / x^*_{p(e)}$.

We will prove the following two statements.

Theorem 5 $\mathrm{E}[\mathrm{cost}(F)] = OPT_{LP}$.

Theorem 6 *For any group X_t, the probability that some F contains a path from r to some node in X_t is at least $\frac{1}{4(\log_2(4g)+1)}$.*

For now, let us suppose these are true. Algorithm 2 below is the main algorithm.

Algorithm 2 Main Subroutine for the Group Steiner Tree Rounding Algorithm

---

Solve (**LP-GST**) (a min-cut algorithm can be used to separate the constraints)
$F' \leftarrow \emptyset$
$\Delta \leftarrow \lceil 4(\log_2(4g) + 1) \ln(6k) \rceil$
for Δ iterations **do**
  Run Algorithm 1 and let F be the returned edges.
  $F' \leftarrow F' \cup F$
end for
if F is a feasible solution with cost $\leq 3 \cdot \Delta \cdot OPT_{\mathrm{LP}}$ **then**
  return F
else
  Declare *failure*
end if

Clearly Algorithm 2 runs in polynomial time and the returned solution has cost $O(\log g \cdot \log k)$ if failure is not declared.

Lemma 1 *Algorithm 2 declares failure with probability at most* $1/2$.

Proof. Since F is the union of Δ different calls to Algorithm 1, then by Lemma 5 the expected cost of F is exactly $\Delta \cdot OPT_{\mathrm{LP}}$. By Markov's inequality, $\Pr[\mathrm{cost}(F) > 3 \cdot \Delta \cdot OPT_{\mathrm{LP}}] \leq 1/3$.

By Theorem 6, the probability a particular group is not connected to r in any of the sets F found over all Δ iterations is at most

$$\left(1 - \frac{1}{4(\log_2(4g)+1)}\right)^{\Delta} \leq \left(1 - \frac{1}{4(\log_2(4g)+1)}\right)^{4(\log_2(4g)+1)\ln(6k)} \leq e^{-\ln(6k)} = \frac{1}{6k}.$$

Since there are k groups, by the union bound the probability that some group is not connected is at most $k \cdot 1/(6k) = 1/6$. By the union bound again, the probability that either the cost exceeds $3 \cdot \Delta \cdot OPT_{\mathrm{LP}}$ or that F is not feasible is at most $1/3 + 1/6 = 1/2$. ■

25.2.1 The Cost Bound

Theorem 5 follows immediately from the following observation.

Lemma 2 *Consider Algorithm 1. For any edge $e \in E$,* $\Pr[e \in F] = x_e^*$.

Proof. By induction on the depth of e. If $e \in E_1$, this is immediate from the first loop of Algorithm 1.

Suppose $e \in E_i$ for $i \geq 2$. By induction, we know $\Pr[p(e) \in F] = x^*_{p(e)}$. Now, the events "$e \in F$" and "$e \in F$ and $p(e) \in F$" happen with the same probability because e can only be added to F if $p(e)$ is in F. Therefore we have

$$\Pr[e \in F] = \Pr[e \in F \text{ and } p(e) \in F] = \Pr[e \in F | p(e) \in F] \cdot \Pr[p(e) \in F] = \frac{x^*_e}{x^*_{p(e)}} \cdot x^*_{p(e)} = x^*_e.$$

■

25.2.2 The Connection Probability Bound

Let $Q(t)$ be the set of all paths from a node in X_t to r. Thus, $|Q(t)| = |X_t|$. We now focus on proving Theorem 6, which we do in two steps.

First, we show something slightly different. If $\overline{x}$ is such that $5\overline{x}$ is a feasible LP solution with a few additional properties, then the probability that X_t is connected to r after one call to Algorithm 1 is at least $\frac{1}{4\cdot(h+1)}$ where h is the height of the tree. Then we reduce the general case to this case with $h = \log_2(4g)$.

From now on, we focus on a single terminal group X_t.

Theorem 7 *Let $\overline{x} \in [0,1]^E$ satisfy:*

- $\overline{x}_e \leq \sum_{e' \text{ a leaf edge under } e} \overline{x}_{e'}$.
- $\sum_{e \in \delta(S)} \overline{x}^t_e \geq 1/4$ *for each* $X_t \subseteq S \subseteq V - \{r\}$.

Then running Algorithm 1 with $\overline{x}$ instead of x^* will connect X_t to r with probability at least $\frac{1}{4\cdot(h+1)}$ where h is the height of T.

The key property here is the first property. Intuitively, it says the following. For any edge e, if we are given $e \in F$ then we still expect at most one $r - X_t$ path in F that uses e. This will be made formal in the proof.

Proof. We will show that the set of edges F sampled in Algorithm 1 will contain some $P \in Q(t)$ as a subset with probability at least $1/(4\cdot(h+1))$. Define the random variable $\mathbf{Z}$ to be the number of $P \in Q(t)$ with $P \subseteq F$. Thus, our goal is to show $\Pr[\mathbf{Z} \geq 1] \geq 1/(4\cdot(h+1))$.

As before, we have $\Pr[e \in F] = \overline{x}_e$ when running Algorithm 1 with $\overline{x}$. Thus, for a given path $P \in Q(t)$ where, say, $e(P)$ is the last edge on P we have $\Pr[P \subseteq F] = \Pr[e(P) \in F] = \overline{x}_{e(P)}$. Thus,

$$\mathrm{E}[\mathbf{Z}] = \sum_{P \in Q(t)} \overline{x}_{e(P)} = \sum_{e \in \delta(X_t)} \geq 1/4.$$

The last equality is because X_t is a set of leaf nodes in T.

While the expected number of paths $r - X_t$ paths in F is at least $1/4$ then intuitively one might expect that there should be at least one path in F with reasonable probability (and our goal is quite modest: $\frac{1}{4\cdot(h+1)}$). However, one potential problem is that most of the time there are no paths and just a small amount of the time there are many paths. We will show this is not possible by placing an upper bound on $\mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1]$.

So, we proceed by noting

$$1/4 \leq \mathrm{E}[\mathbf{Z}] = \mathrm{E}[\mathbf{Z}|\mathbf{Z}=0] \cdot \Pr[\mathbf{Z}=0] + \mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1] \cdot \Pr[\mathbf{Z} \geq 1] = \mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1] \cdot \Pr[\mathbf{Z} \geq 1].$$

It suffices to prove $\mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1] \leq h+1$. In fact:

Claim 2 *If for every $P \in Q(t)$ we have* $\mathrm{E}[\mathbf{Z}|P \subseteq F] \leq h+1$ *then* $\mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1] \leq h+1$.

Proof. For every random variable $\mathbf{Y}$ we have $\mathrm{E}[\mathbf{Y}]^2 \leq \mathrm{E}[\mathbf{Y}^2]$. So,

$$\begin{aligned}
\mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1]^2 &\leq \mathrm{E}[\mathbf{Z}^2|\mathbf{Z} \geq 1] \\
&= \sum_{P,P' \in Q(t)} \Pr[P, P' \subseteq F|\mathbf{Z} \geq 1] \\
&= \sum_{P \in Q(t)} \Pr[P \subseteq F|\mathbf{Z} \geq 1] \cdot \left(\sum_{P' \in Q(t)} \Pr[P' \subseteq F|\mathbf{Z} \geq 1 \wedge P \subseteq F] \right) \\
&= \sum_{P \in Q(t)} \Pr[P \subseteq F|\mathbf{Z} \geq 1] \cdot \left(\sum_{P' \in Q(t)} \Pr[P' \subseteq F|P \subseteq F] \right) \\
&\leq (h+1) \cdot \sum_{P \in Q(t)} \Pr[P \subseteq F|\mathbf{Z} \geq 1] \\
&= (h+1) \cdot \mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1]
\end{aligned}$$

The second last equality is because $P \subseteq F$ if and only if both $\mathbf{Z} \geq 1$ and $P \subseteq F$ (i.e. we are conditioning on the same event). Simplifying shows $\mathrm{E}[\mathbf{Z}|\mathbf{Z} \geq 1] \leq h+1$. ■

Now focus on a particular $P \in Q(t)$. Let $P = \{e_1, e_2, \ldots, e_k\}$ (where $k \leq h$) and let $P_i = \{e_1, \ldots, e_i\}$ for $0 \leq i \leq k$ (here P_0 is the trivial path that starts and ends at r). Now consider any $P' \in Q(t)$ and say that $e_i \in P'$ but $e_{i+1} \notin P'$. Then

$$\Pr[P' \subseteq F|P \subseteq F] = \Pr[P' \subseteq F|P_i \subseteq F] \tag{25.1}$$

because the events that $e_i \in F$ is extended to P or to P' are independent. Similar to how we have $\Pr[e \in F] = \overline{x}_e$, we have $\Pr[P' \subseteq F|P_i \subseteq F] = \overline{x}_{e(P')}/\overline{x}_{e_i}$.

Finally, let $Q_P(t,i)$ denote all paths $P' \in Q(t)$ with $e_i \in P'$ and $e_{i+1} \notin P'$. Note that each $P' \in Q(t)$ lies in exactly one of $Q_P(t,0), Q_P(t,1), \ldots, Q_P(t,k)$. So,

$$\begin{aligned}
\mathrm{E}[\mathbf{Z}|P \subseteq F] &= \sum_{i=0}^{k} \sum_{P' \in Q_P(t,i)} \Pr[P' \subseteq F|P_i \subseteq F] \quad \text{(by (25.1))} \\
&= \sum_{i=0}^{k} \sum_{P' \in Q_P(t,i)} \overline{x}_{e(P')}/\overline{x}_{e_i} \\
&\leq \sum_{i=0}^{k} 1 \\
&\leq h+1.
\end{aligned}$$

The first inequality is by the first assumption in the statement of the theorem we are proving (Theorem 7).

Putting this all together, we have shown $1/4 \leq (h+1) \cdot \Pr[\mathbf{Z} \geq 1]$ which completes the proof of Theorem 7. ■

25.2.3 Reducing to the Conditions of Theorem 7

We will consider a series of reductions that eventually lead to an instance T' with values $\overline{x}$ that satisfy the conditions of Theorem 7 with the height h of T' being at most $\log_2(4g)$.

I am attempting to be more precise in these notes than we were in the lectures, which is why the presentation deviates significantly in some parts from what we saw in class.

Step 1 - Making T Binary
First, we convert T to a full binary tree (i.e. each non-leaf contains exactly two children). If some $v \neq r$ contains exactly one child edge e then we merge e and $p(e)$ to a single edge with x^*-value x^*_e. Similarly, if r has one child edge e then simply remove r and make its child the root (we must have $x^*_e = 1$ in this case). If v has more than two child edges $\{e_1, \ldots, e_a\}$, then create a new edge e' and a new vertex v', set $x^*_{e'} = x^*_{p(e_1)}$, detach $e_2, \ldots, e_a$ from v and attach them to v', and finally connect v and v' by e' (if $v = r$ then simply set $x^*_{e'} = 1$). It is easy to check that running Algorithm 1 in either the original tree or in this binary tree does not change the probability that r connects to X_t.

Aside - Reducing the x-Values
We consider the effect of reducing the LP-weight across edges, which will be used in a few subsequent steps.

Lemma 3 *Suppose $\overline{x}$ satisfies $\overline{x} \leq x^*$ (on an edge-by-edge basis) and $\overline{x}_e \leq \overline{x}_{p(e)}$ for each $e \in E - E_1$. Then the probability of connecting r to X_t is no greater when using $\overline{x}$ in Algoritm 1 than when using x^*.*

This seems intuitively true since decreasing the x-values should probably harm our chances of connecting, but it takes a proof since the rounding algorithm divides by some x-values.

Proof. It suffices to prove this when $\overline{x}_e \neq x^*_e$ for exactly one edge e since we can then change the solution from x^*_e to $\overline{x}$ by changing the value of one edge at a time, starting with the lowest edges.

We take a slightly different view of Algorithm 1. For every edge e, we flip a coin C_e that is HEADS with probability $\overline{x}_e/\overline{x}_{p(e)}$ (or simply $\overline{x}_e$ if e is incident to the root r). Then we add e to F if every edge between e and r had its coin come up HEADS. This constructs the same set of paths with the same probability as before.

Consider these events:

- **S**, the event that F connects r to X_t (i.e. the algorithm "succeeds").
- **A**, the event that F includes some $P \in Q(t)$ where $e \notin P$.
- **B**, the event that all coins between $p(e)$ and r come up HEADS. If e is incident to the root then **B** always happens.

Below we subscript probabilities with x^* or $\overline{x}$ to indicate whether we are using x^* or $\overline{x}$ in Algorithm 1. So, our goal is to show $\Pr_{x^*}[\mathbf{S}] \geq \Pr_{\overline{x}}[\mathbf{S}]$.

If e is a leaf edge then we have $\Pr_{x^*}[C_{e'} = \text{HEADS}] \geq \Pr_{\overline{x}}[C_{e'} = \text{HEADS}]$ for every edge e'. In this case, it is easy to see $\Pr_{x^*}[\mathbf{S}] \geq \Pr_{\overline{x}}[\mathbf{S}]$.

However, if e is not a leaf edge then we have $\Pr_{x^*}[C_{e'} = \text{HEADS}] \leq \Pr_{\overline{x}}[C_{e'} = \text{HEADS}]$ for every child edge e' of e. Still, we can show $\Pr_{x^*}[\mathbf{S}] \geq \Pr_{\overline{x}}[\mathbf{S}]$ in this case.

Note that if neither event **A** or **B** happen then **S** does not happen. Furthermore, if event **A** happens then event **S** also happens. Thus,

$$\begin{aligned}
\Pr_{x^*}[\mathbf{S}] &= \Pr_{x^*}[\mathbf{A}] + \Pr_{x^*}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}] \cdot \Pr_{x^*}[\neg\mathbf{A} \wedge \mathbf{B}] \\
&= \Pr_{\overline{x}}[\mathbf{A}] + \Pr_{x^*}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}] \cdot \Pr_{\overline{x}}[\neg\mathbf{A} \wedge \mathbf{B}]
\end{aligned}$$

The second equality is because the corresponding events only depend on the random coins for edges e' that are not e nor a child edge of e. All we have left to show is that $\Pr_{x^*}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}] \geq \Pr_{\overline{x}}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}]$.

Say e_1, e_2 are the two child edges of e. For $i = 1, 2$, let p_i denote the probability that some $P \in Q(t)$ with $e_i \in P$ has all coins between $e(P)$ and its edge just below e_i come up "true". If e_i is a leaf edge then let $p_i = 1$ if $e_i = e(P)$ for some $P \in Q(t)$ and $p_i = 0$ otherwise. Note that this probability is the same whether we are running Algorithm 1 with either x^* or $\overline{x}$.

So, given $\neg\mathbf{A} \wedge \mathbf{B}$, we have $\mathbf{S}$ if and only if some $P \in Q(t)$ with $e \in P$ has all coins from $e(P)$ up to and including e flipped "true". This happens exactly when $C_e = \textsc{Heads}$ and for at least one $i = 1, 2$ we have $C_{e_i} = \text{“}true''$ and some path $P \in Q(t)$ with $e_i \in P$ has all coins up to the edge just before e_i flipped to true. The probability this happens under x^* is exactly as follows, where we use $x^*_{p(e)} = 1$ if e is incident to r:

$$\frac{x^*_e}{x^*_{p(e)}} \cdot \left(\frac{x^*_{e_1}}{x^*_e} \cdot p_1 + \frac{x^*_{e_2}}{x^*_e} \cdot p_2 - \frac{x^*_{e_1} \cdot x^*_{e_2}}{x^*_e \cdot x^*_e} \cdot p_1 \cdot p_2 \right) = \alpha - \frac{\beta}{x^*_e}.$$

Where both α and β are nonnegative terms that do not depend on x^*_e. When x^*_e is reduced to $\overline{x}_e$, we see this probability does not increase, thus

$$\Pr_{x^*}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}] \geq \Pr_{\overline{x}}[\mathbf{S}|\neg\mathbf{A} \wedge \mathbf{B}].$$

∎

Step 2 - Reducing x^* to a Flow

By the constraints of (**LP-GST**), we can send one unit of flow from X_t to r such that the amount of flow f^t_e sent across any edge $e \in E$ is at most x^*_e. Note that since f^t is a flow then $\overline{f}^t_e = \sum_{e' \text{ a leaf edge under } e} \overline{f}^t_{e'}$ for any $e \in E$. Furthermore, since it represents one unit of flow then $\sum_{e \in \delta(S)} f^t_e \geq 1$ for each $X_t \subseteq S \subseteq V - \{r\}$. By Lemma 3, this does not increase the probability of connecting r to X_t.

Now we just have to reduce the height.

Step 3 - Rounding and Deleting Some Edges

For an edge $e \in E$, suppose $2^{-i} \leq f^t_e < 2^{-i+1}$. Round f^t_e down to 2^i. Note that we still have $\sum_{e \in \delta(S)} f^t_e \geq 1/2$ for each $X_t \subseteq S \subseteq V - \{r\}$. Finally, delete every edge whose rounded f^t value is $< 1/(4g)$ (recall $|X_t| \leq g$). Let $\overline{x}'$ denote the resulting values for the remaining edges. Since the scaled f^t values still support $1/2$ of a unit of $X_t - r$ flow and since this flow is sent along at most g paths, then we still have $\sum_{e \in \delta(S)} x^t_e \geq 1/2 - |X_t|/(4g) \geq 1/4$. By Lemma 3, this still does not increase the probability of connecting r to X_t.

Step 4 - Height Reduction

For any edge e such that $\overline{x}'_e = \overline{x}'_{p(e)}$, simply contract e. Since $e \in F$ if and only if $p(e) \in F$ (because $\overline{x}'_e/\overline{x}'_{p(e)} = 1$) then this does not change the probability of connecting r to X_t. Let T' be the resulting tree. However,

Lemma 4 *The height of T' is at most $\log_2(4g)$.*

Proof. For any edge e let $p(e)$ be its parent edge in T'. We have $\overline{x}'_e \leq \overline{x}'_{p(e)}/2$ because Step 4 had distinct edges differing by a factor of at least $1/2$ and the consolidation step ensures no edge has the same value as its parent. Since $1/(4g) \leq \overline{x}'_e \leq 1$ for each edge e then the height of the tree is at most $\log_2(4g)$. ∎

Step 5 - Reducing to a Flow Again
Now we have a tree T' with height at most $\log_2(4g)$ and corresponding $\overline{x}'$ values with $\sum_{e\in S}\overline{x}'_e \geq 1/4$ for any $X_t \subseteq S \subseteq V - \{r\}$. Furthermore, the probability of connecting r to X_t in this tree using $\overline{x}$ is no greater than connecting r to X_t in the original tree T under x^*. We had ensured the first condition of Theorem 7 held under f^t, but we may have lost that in the future reductions. However, we simply let $\overline{x}$ be 1/4 units of $X_t - r$ flow with $\overline{x}_e \leq \overline{x}'_e$, which exists because $\sum_{e\in S}\overline{x}'_e \geq 1/4$. Again, by Lemma 3 this does not increase the probability of connecting r to X_t.

Note: I just noticed that the height reduction step has resulting height at most $4g+1$, not $4g$. Furthermore, the contraction may cause some X_t node to no longer be a leaf node. If we avoid contracting leaf nodes but do everything else the same, then we can still say each node in X_t is a leaf and the height is at most $4g+2$ which suffices to get the $O(\log g \cdot \log k)$ integrality gap bound.

I don't want to go through and make the tiny changes in the analysis everywhere. Really, the only change to the algorithm is that Algorithm 2 should use $\Delta \leftarrow \lceil 4(\log_2(4g)+3)\ln(6k)\rceil$ instead.

25.3 Discussion

The rounding algorithm for trees was described by Garg, Konjevod, and Ravi [GKR98]. In their analysis, they use a concentration bound called *Janson's Inequality* (not to be confused with *Jenson's*), which is like a Chernoff bound in settings with limited dependence. Our presentation follows analysis that is suggested by Rothvoss [R11].

The integrality gap of (**LP-GST**) was shown to be $\Omega((\log n/\log\log n)^2)$ in trees [H+03], which was later followed by the hardness result in Theorem 2 [HK03].

An open problem is to get an $O(\log^2 n)$-approximation in general metrics. An $O(\log^2 k)$-approximation with quasi-polynomial running time is known [CP05].

References

CP05 C. Chekuri and M. Pál, A recursive greedy algorithm for walks in directed graphs, In Proceedings of FOCS, 2005.

GKR98 N. Garg, G. Konjevod, and R. Ravi, A polylogarithmic approximation algorithm for the group Steiner tree problem, In Proceedings of SODA, 1998.

H+03 E. Halperin, G. Kortsarz, R. Krauthgamer, A. Srinivasan, and N. Wang, Integrality ratio for group Steiner trees and directed Steiner trees, In Proceedings of SODA, 2003.

HK03 E. Halperin and R. Krauthgamer, Polylogarithmic inapproximability, In Proceedings of STOC, 2003.

R11 T. Rothvoss, Directed Steiner tree and the Lasserre hierarchy, CoRR, abs/1111.5473, 2011.