IOE 691: Approximation Algorithms Date: 03/22/2017

Lecture Notes: Degree-bounded MST (Iterative LP)

Instructor: Viswanath Nagarajan Scribe: Mayank Agrawal

1 Degree-bounded minimum spanning tree (MST) problem

In the last lecture, we discussed the algorithm and constraints for solving bounded degree minimum spanning tree (MST) problem. In this lecture we prove its approximation and validity. The input graph is $G = (V, E_0)$ with edge-costs $\{c_{\geq} 0 : e \in E_0\}$ and degree bounds $\{b_v : v \in V\}$. The goal is to find a minimum cost spanning tree that satisfies all degree bounds.

The algorithm is iterative and it maintains the following:

- $E \subseteq E_0$ the set of floating or undecided edges,
- $F \subseteq E_0 \setminus E$ the set of edges in the current solution and
- W the set of vertices with degree bound.

The LP solved in each iteration $LP(E, F, W)$ is given below.

$$\begin{aligned}
\min \quad & \sum_{e \in E_0} c_e x_e \\
s.t. \quad & \sum_{e \in E(S)} x_e \leq |S| - 1, \qquad \forall S \subseteq V \\
& \sum_{e \in E(V)} x_e = |V| - 1 \\
& x(\delta(v)) \leq b_v, \qquad \forall v \in W \\
& x_e \geq 0, \qquad \forall e \in E \\
& x_e = 1, \qquad \forall e \in F \\
& x_e = 0, \qquad \forall e \in E_0 \setminus (F \cup E).
\end{aligned}$$

The first and second set of constraints are the spanning tree constraints; the third set of constraints are the degree constraints. The last two constraints enforce the fixed values (either 0 or 1) of the edges $E_0 \setminus E$. We use $\delta(v)$ to denote the set of edges of E_0 incident to vertex v.

The algorithm is given as Algorithm 1, which gives Theorem 1.1 as the main result.

Theorem 1.1 *Algorithm 1 generates a solution with cost $\leq OPT$ and $deg_u \leq b_u + 1$ for all $u \in V$. Here OPT denotes the optimal cost of a spanning tree that satisfies all the degree bounds.*

Proof: By Lemma 1.1, the algorithm terminates. It is easy to see that the solution F is a spanning tree as the final LP has $E = \emptyset$: so each variable fixed to zero (for $E_0 \setminus F$) or one (for F). This corresponds to an (integral) spanning tree F.

We now prove the desired guarantee. In the first iteration, we start with $E = E_0$ and $W = V$,

Algorithm 1

1: Start with $E = E_0$, $F = \emptyset$, $W = V$.
2: **while** $E \neq \emptyset$ **do**
3: Solve LP(E,F,W) to get extreme point x
4: **if** $x_e = 0$ for $e \in E$ **then**
5: $E \leftarrow E \setminus e$
6: **if** $x_e = 1$ for $e \in E$ **then**
7: $E \leftarrow E \setminus e$
8: $F \leftarrow F \cup e$
9: **if** $|\delta_E(u)| + |\delta_F(u)| \leq b_u + 1$ for some $u \in W$ **then**
10: $W \leftarrow W \setminus u$
11: Output solution F.

therefore the LP cost is at most OPT. Moreover, in each iteration we only relax constraints in the current LP or fix variables to their optimal (LP) values: so the final LP cost will also be at most OPT. Note that the final LP has cost exactly $c(F)$. So $c(F) \leq OPT$.

For the degree bounds, consider any $u \in V$. If $u \in W$ at the end of the algorithm, it is clear that $deg_u \leq b_u$ as this must be satisfied in the final LP. If $u \notin W$ at the end, consider the iteration when u was dropped from W. If E and F denote the corresponding sets at this point, it is clear that $|\delta_E(u)| + |\delta_F(u)| \leq b_u + 1$ (see Step 9). Since none of the other edges $E_0 \setminus (F \cup E)$ can be in the final solution, it follows that $deg_u \leq b_u + 1$. ■

Theorem 1.2 *For any extreme point x to $LP(E, F, W)$, there is a laminar family $L \subseteq 2^V$ and $U \subseteq W$ such that:*

1. *$\sum_{e \in E(S)} x_e = |S| - 1$ for all $S \in L$ (tight spanning tree constraints).*
2. *$x(\delta(v)) = b_v$ for all $v \in U$ (tight degree constraints).*
3. *All constraints in L and U are linearly independent.*
4. *$|E| = |L| + |U|$.*

Proof: Proved in the last lecture ■

See Fig. 1(a) for an example L with sets $A_i \in L$

Lemma 1.1 *Algorithm 1 terminates in at most $|E_0| + |V|$ iterations.*

Proof: It suffices to show that one of Step 4, 6 or 9 applies in every iteration. Then the algorithm must terminate in $|E_0| + |V|$ iterations because each iteration decreases either W or E.

Consider any iteration (E, F, W). We show by contradiction that one of Step 4, 6 or 9 applies. We use a "token distribution and collection" approach to get the contradiction.

Assign one token to each edge $(u, v) \in E$ and distribute the token as x_e units to the edge and $(1 - x_e)/2$ units to u and v each (shown in Fig. 1(b)). Note that all these units are strictly positive as $0 < x_e < 1$ for all $e \in E$ (recall steps 4,6 do not apply).

Now we re-collect tokens from vertices and edges.

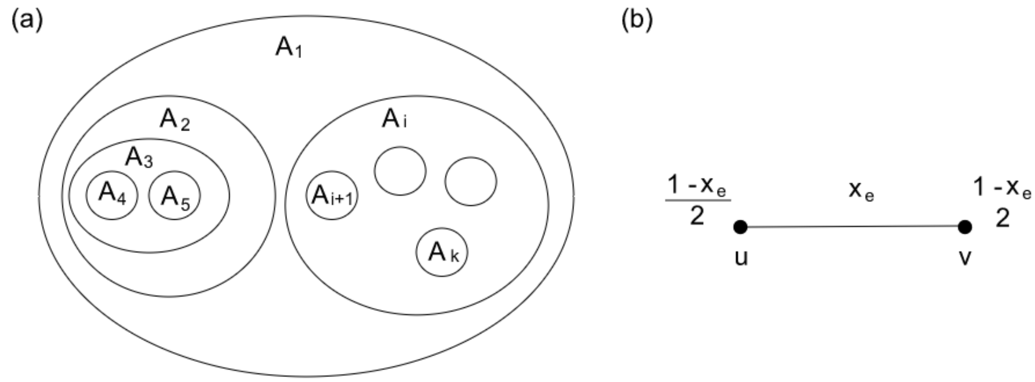

Figure 1: Figure 1. (a) An example of Laminar family with domains representing sets in the family (b) Distribution of the token of an edge u, v.

1. For each vertex $w \in U$ with a tight degree constraint, we get
$$\sum_{v:(w,v)\in E} \frac{1-x_{wv}}{2} = \frac{1}{2}|\delta_E(w)| - \frac{1}{2}x(\delta_E(w))$$
tokens. Now since w has tight degree constraint, $x(\delta_E(w)) + |\delta_F(w)| = b_w$. Also, if Step 9 does not apply then $|\delta_E(w)| + |\delta_F(w)| \geq b_w + 2$. Hence, combining both equations we obtain at least $1/2(b_w + 2 - \delta_F(w)) - 1/2(b_w - |\delta_F(w)|) = 1$ token at w.

2. For any subset $A_i \in L$, we collect all the "edge tokens" x_e from all edges $e = (u, v)$ such that A_i is the minimal set containing both u and v. As L is a laminar family, for any edge e there is at most one minimal set $A_i \in L$ containing both its end points. So each edge token is assigned to at most one subset in L.

   Consider any $A_i \in L$ with children $B_1, \cdots B_k \in L$. The number of children k may be 0 (leaf nodes in L). The edge tokens from $E(A_i) \setminus \left(\cup_{j=1}^{k} E(B_j)\right)$ will be assigned to A_i. As A_i and the B_js are tight constraints, $x(E(A_i)) = |A_i| - 1 - |F(A_i)|$ and $x(E(B_j)) = |B_j| - 1 - |F(B_j)|$ for $j \in [k]$. The number of tokens assigned to A_i is
$$Q = \sum_{e\in E(A_i)\setminus\left(\cup_{j=1}^{k} E(B_j)\right)} x_e \quad = \quad x(E(A_i)) - \sum_{j=1}^{k} x(E(B_j)).$$
   This quantity Q is an integer because all terms in the right hand side are integers (due to the tight constraints). Note that $E(A_i) \setminus \left(\cup_{j=1}^{k} E(B_j)\right) \neq \emptyset$ as the constraints corresponding to A_i and B_js are linearly independent (Theorem 1.2). Moreover, $x_e > 0$ for each $e \in E$. So Q is also positive. So the amount of tokens we collect for A_i is $Q \geq 1$.

Note that the total number of tokens collected so far is at least $|U| + |L| = |E|$ (Theorem 1.2).

We now show that we can collect some extra tokens, which leads to a contradiction as only $|E|$ tokens were distributed.

If any edge $e \in E$ is incident to any $v \in V \setminus U$, we get $(1 - x_e)/2$ token at v which has not been used above- a contradiction. So all floating edges (E) must be incident to vertices with tight degree constraint (U). So we have

$$\mathbf{1}(E) = \frac{1}{2}\sum_{v \in U} \mathbf{1}(\delta_E(v))$$

where $\mathbf{1}(E')$ is the indicator vector of any subset $E' \subseteq E$.

Now consider if any edge $e \in E$ is not contained in any maximal set in L. Then we get x_e amount of extra token that is not used- a contradiction. This implies that $E = \cup_{i=1}^{t} E(M_i)$, where M_i's are maximal sets in L. Naturally, M_i's are disjoint. So

$$\mathbf{1}(E) = \sum_{i=1}^{t} \mathbf{1}(E(M_i)).$$

Using the above two equations, we obtain that the constraints in $\{M_i\}_{i=1}^{t}$ and U are linearly dependent, which contradicts Theorem 1.2. ∎