Saarland Informatics Campus

Script of Lecture 12, Approximation Algorithms
Summer term 2017

Tobias Mömke, Hang Zhou
`http://www-cc.cs.uni-saarland.de/course/61/`

Written by Tobias Mömke

1 Summary of Results

In the previous lecture, you have seen some key results that we will apply to bounded degree spanning trees (see also [1]).

Lemma 1 (Rank Lemma). *Let $P = \{x : Ax \geq b, x \geq 0\}$ and let x be an extreme point solution of P such that $x_i > 0$ for each i. Then every maximal number of linearly independent tight constraints of the form $A_i x = b_i$ for some row i of A equals the number of variables.*

We considered the spanning tree LP (T)

$$\text{Minimize} \quad \sum_{e \in E} x_e c_e \tag{1}$$

$$\text{s.t.} \quad \sum_{e \in E[S]} x_e \leq |S| - 1 \quad \text{for all } \emptyset \neq S \subset V \tag{2}$$

$$\sum_{e \in E[V]} x_e = |V| - 1 \tag{3}$$

$$x_e \geq 0 \quad \text{for all } e \in E \tag{4}$$

The set of tight constraints is $\mathcal{F} = \{S \subseteq V : x(E[S]) = |S| - 1\}$ and we obtained the following lemma.

Lemma 2. *There is a laminar family $\mathcal{L} \subseteq \mathcal{F}$ such that both $\mathcal{L}$ and $\mathcal{F}$ have the same maximal number of linearly independent constraints.*

You have seen in the exercise that every laminar family of subsets of V is composed of at most $2|V| - 1$ sets. This number can only be attained if $|V|$ of the sets are the singletons $\{v\}$ for $v \in V$. Observe that (T) does not contain constraints for singleton sets. Therefore Lemma 2 implies that there are at most $|V| - 1$ tight linearly independent constraints. By the rank lemma we obtain the following.

Lemma 3. *Every extreme point solution x of (T) has at most $|V| - 1$ non-zero variables.*

2 MST Algorithm

We run the following algorithm which iteratively removes edges and vertices from G.

a) Initialize $F \leftarrow \emptyset$;

b) While $|E(G)| > 0$

   (i) Find an optimal extreme point solution x to (T) for G. Remove all edges e from G with $x_e = 0$;

   (ii) Find a leaf v with edge $e = \{u, v\}$. Set $F \leftarrow F \cup \{e\}$, remove v from G.

c) Output: F

We first note that each e in line (ii) has $x_e = 1$ since $x(E[V]) = |V(G)| - 1$ and $x(E[V \setminus \{v\}]) \leq |V(G)| - 2$. Therefore $x_e \geq 1$. At the same time $x(e) \leq E[\{u, v\}] - 1 = 1$.

To see that F is the edge set of a spanning tree, let us consider the edges in reverse order of when they were added. Then we start building a tree with a single edge and always add leaves. Adding a leave cannot disconnect the graph or close a cycle.

Minimality follows since removing e can only relax the LP and thus decrease the cost.

To finish the proof, we have to argue that the algorithm terminates.

Lemma 4. *In line (ii), the algorithm always finds a leaf.*

Proof. Suppose the contrary that every vertex has a degree of at least 2. Then $\sum_{v \in V} d(v) \geq 2|V|$ which implies $|E| \geq |V|$. By Lemma 3, however, $|E| < |V|$, a contradiction. □

Our insides will allow us in the following to only concentrate on degree constraints. As soon as all of them disappear, the above results apply.

3 Degree Bounded Spanning Tree

We now add degree constraints to (T) and obtain the following LP (B).

$$\begin{aligned}
\text{Minimize} \quad & \sum_{e \in E} x_e c_e && && (5)\\
\text{s.t.} \quad & \sum_{e \in E[S]} x_e \leq |S| - 1 && \text{for all } \emptyset \neq S \subset V && (6)\\
& \sum_{e \in E[V]} x_e = |V| - 1 && && (7)\\
& x(\delta(v)) \leq B_v && \text{for all } v \in W && (8)\\
& x_e \geq 0 && \text{for all } e \in E && (9)
\end{aligned}$$

The notation B_v allows us to solve a more general problem: instead of requiring a degree bound b for each vertex, we can specify an *individual* degree bound for each vertex. Then the set W is the set of vertices with degree bound. Thus in our old setting, $B_v = b$ for each $v \in V$ and $W = V$.

We now run the following algorithm which successively removes all degree constraints.

a) While $W \neq \emptyset$ do
   (i) Find an optimal extreme point solution x to (B), remove every edge e with $x_e = 0$.
   (ii) If there is a $v \in W$ with $d(v) \leq B_v + 1$, then $W \leftarrow W \setminus \{v\}$.

b) Output: the support of an optimal extreme point solution for the remaining instance, i.e., the edges e with $x_e > 0$.

We first note that by our previous analysis, the solution is a spanning tree since without degree constraints we are left with only the MST constraints. We have seen that an optimal extreme point solution is integral therefore gives an MST on the remaining solution.

Let us now consider Line (ii). If we find such a vertex v, it is save to remove v from W since in the subsequent algorithm we only remove edges and therefore the degree constraint cannot be violated. Removing constraints from an LP can never increase the minimum cost since we relax the LP. Therefore the computed solution cannot be larger then a minimum cost bounded degree-B bounded spanning tree.

To finish the proof, we have to show that we always find a vertex v in Line (ii).

We extend our analysis of extreme point solutions by pointing out the following properties which follow from directly from our lemmas.

Observation 1. *There is a set $T \subseteq W$ and a laminar family $\mathcal{L} \subseteq \mathcal{F}$ such that*

a) $x(\delta(v)) = B_v$ for each $v \in T$, $x(E[S]) = |S| - 1$ for each $S \in \mathcal{L}$

b) The vectors in the set $\{\chi(E[S]) : S \in \mathcal{L}\} \cup \{\chi(\delta(v)) : v \in T\}$ are linearly independent

c) $|\mathcal{L}| + |T| = |E|$

To analyze the algorithm, we first note that as soon as T is empty, we obtain an integral solution (due to our analysis of the MST algorithm). All remaining vertices in W then satisfy the degree constraints.

We therefore are left with showing the following.

Lemma 5. *Let x be an extreme point solution to (B) with $x_e > 0$ for all $e \in E(G)$. Let $\mathcal{L}$ and $T \subseteq W$ correspond to the tight set constraints and the tight degree constraints from Observation 1. If $T \neq \emptyset$ then there is a vertex v with $d(v) \leq B_v + 1$.*

Proof. We use a local fractional token argument. We give one token to each edge $e \in E(G)$. Now the edges distribute their tokens fractionally to $\mathcal{L}$ and T such that each set constraint and each degree constraint obtains at least one token. Let us assume by contradiction that there is no vertex v as claimed in the lemma. We then show that there is a constraint that obtains more then one token. This contradicts that $|\mathcal{L}| + |T| = |E|$ and we conclude that v must exist.

We assign the tokens to constraints as follows. Each edge e assigns an x_e fraction of its token to the smallest set $L \in \mathcal{L}$ that contains both ends of e. If there is no such set, e keeps the fractional token.

Then $e = \{u, v\}$ assigns $(1 - x_e)/2$ to u if $u \in T$ and additionally $(1 - x_e)/2$ to v if $v \in T$, for the degree constraints.

Let us first argue that each constraint receives at least one token. Let us consider $L \in \mathcal{L}$ and let $R_1, R_2, \ldots, R_k$ be the children of L (the maximal subsets); possibly $k = 0$. We first claim that L receives a non-zero amount of tokens. Otherwise $x(E[L]) = \sum_i x(E[R_i])$ and thus a linear combination of other constraints – contradicting that all R_i and L are in $\mathcal{F}$.

Since all R_i are tight, all $x(R[E_i])$ are integer. Therefore the amount of fractional tokens received by L must be integer, too, and L receives at least one full token.

To see that also each $u \in T$ receives a full token, recall that $x(\delta(u)) = B_u$. Therefore

$$\sum_{e \in \delta(u)} \frac{1 - x_e}{2} = \frac{d(u) - B_u}{2} \geq 1$$

The inequality holds since we assume that $d(u) > B_u + 1$ and therefore $d(u) \geq B_u + 2$.

We now argue that we have not assigned all tokens (and thus there are more then $|E|$ tokens, giving the contradiction).

If $V \notin \mathcal{L}$, there is an edge e between two maximal sets of $\mathcal{L}$ which did not assign x_e of its token and we are done. We also directly get a contradiction if there is an edge e with $x_e < 1$ and one end not in T, because at least $(1 - x_e)/2 > 0$ of its token were not yet assigned.

This implies that for each vertex $u \notin T$, all incident edges e have $x_e = 1$.

We show that if we did not get a contradiction, the constraint for $x(E[V])$ is a linear linear linear combination of other constraints, again a contradiction. We have

$$2\chi(E[V]) = \sum_{v \in V} \chi(\delta(v)) = \sum_{v \in T} \chi(\delta(v)) + \sum_{v \in V \setminus T} \chi(\delta(v)) = \sum_{v \in T} \chi(\delta(v)) + \sum_{v \in V \setminus T} \sum_{e \in \delta(v)} \chi(e)\,.$$

We have seen before that for all $e = \{u, v\}$ with $x_e = 1$, $\{u, v\} \in \mathcal{F}$ and therefore linear combinations of constraints in $\mathcal{L}$. Since $V \in \mathcal{L}$, we can express the constraints for T as linear combinations constraints in $\mathcal{L}$ and other constraints in T, contradicting the linear independence of T and $\mathcal{L}$. □

4 Outlook

We can generalize the 2-approximation for the Steiner forest problem/proper functions to weakly supermodular functions.

Consider the LP

$$\begin{aligned} \min \sum_{e \in E} w_e x_e & \\ s.t. \quad x(\delta(S)) \geq f(S) & \quad \text{for all } S \subseteq V \\ x_e \geq 0 & \quad \text{for all } e \in E \end{aligned}$$

from Lecture 9.

But instead of a proper function, here the function f is a weakly supermodular. This means

$$f(U) + f(V) \leq \max\{f(Ucup V) + f(U \cap V), f(U \setminus V) + f(V \setminus U)\}\,.$$

The integer problem modeled here is called generalized network design.

Theorem 1. *There is a 2-approximation algorithm for generalized network design.*

We only sketch the proof. Similar to our analysis in Lecture 11, we can apply an uncrossing technique to restrict our focus to a laminar family of tight linearly independent constraints.

We then use a local token argument to show that there is always an edge with LP value at least $1/2$. The algorithm is as follows.

a) Initialize $F \leftarrow \emptyset$

b) While there are edges in G

  (i) Compute an optimal extreme point solution x, remove all edges e with $x_e = 0$

  (ii) Find an edge e with $x_e \geq 1/2$, $F \leftarrow F \cup \{e\}$

  (iii) Remove e from G, update f

c) Output: F

We have two tokens for each edge E and aim to assign two tokens to each set in $\mathcal{L}$.

The basic idea is that unless we find e with $x_e \geq 2$, each vertex has a degree of at least 3. We then assign at least 3 tokens to minimal sets from $\mathcal{L}$ and re-assign these to super-sets.

References

[1] Lap-Chi Lau, R. Ravi, and Mohit Singh. *Iterative Methods in Combinatorial Optimization.* Cambridge University Press, New York, NY, USA, 1st edition, 2011.