Approximation Algorithm

Prof. Palash Dey

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

Week – 10

Lecture 49

Lecture 49 : Primal-dual Algorithm for Steiner Forest Contd.

Welcome. So, from the last two lectures we have been studying the primal dual method based approximation algorithm for the generalized Steiner tree or Steiner forest problem. In the last class we have seen the pseudo code of our algorithm where we are ah uniformly increasing multiple dual variables and at the end we claim that ah our algorithm achieves an approximation factor of 2. So, let us prove that ah theorem now. So, generalized Steiner tree or Steiner forest .

So, we wrote down one lemma which let us first assume and using that lemma let us prove the approximation guarantee. So, the lemma for any calci in any iteration of the algorithm. sum over connected components C in **C**, cardinality $\delta(C) \cap F'$, F' is the final Steiner tree output by the algorithm this is less than equal to twice cardinality **C**. Let us assume this lemma and prove this theorem that our algorithm has an approximation factor of at most 2 proof.

So, let us begin with standard primal dual analysis, ALG which is the sum of costs of the edges in F'. ok these are only tight edges. So, for this edge the inequality dual inequality is tight. So, c_e equal to S such that $e \in \delta(S)$ y_S. ok and then we swap the double sum this is $\sum_S |\delta(S) \cap F'| y_S$.

ok. And what we will show is that this is as a sum not individually uniformly this is less than equal to twice cardinality twice sum of $\sum_S y_S$. So, this will show and big sum of y_S is a dual feasible solution. So, this is less than equal to twice LP-opt by weak duality which is less than equal to twice opt. So, all we need to show is this inequality that $\sum_S |\delta(S) \cap F'| y_S$ is less than equal to $2 \sum_S y_S$.

So, let us write to show $\sum_S |\delta(S) \cap F'| y_S$ is less than equal to $2 \sum_S y_S$. So, we will prove it by induction on the number of iterations. So, initially all y_S is 0 and thus the inequality holds ok. So, as an inductive hypothesis let us assume that it holds in the at the beginning of some iteration.

Suppose the inequality holds at the beginning of some iteration, in that iteration some dual variables are increased. We will show that the inequality holds after the iteration at the end of that iteration we will show that the inequality holds at the end of that iteration also ok. So, let **C** be the set of connected components C such that there exists $i \in [k]$ with $|C \cap \{s_i, t_i\}| = 1$. So, let **C** be such a set of these current components at the beginning of the iteration. So, in that iteration suppose we increase the dual variables y_C uniformly Suppose we increase ah dual variables y_C C in **C** uniformly that means, by same amount ah by say epsilon greater than 0 ok.

So, let us see how this increase affects the left hand side and right hand side of the inequality. So, LHS increases by how much? So, for every c conected component c only those variables are touched other variables are not touched. So, and So, the LHS in its increases by only those connected component C in **C** y_C increases by ϵ and this $|\delta(C) \cap F'|$ and y_C is increased by epsilon. So, this times epsilon.

So, LHS increases by this amount. On the other hand RHS increases by what is RHS $2\sum_S y_S$. So, increases by $2\sum_{C \in C} \epsilon$. So, LHS increases by this RHS increases by this. Now, from lemma So, this is the lemma we know that in any iteration in particular in this iteration $\sum_{C \in C} |\delta(C) \cap F'|$ is less than equal to $2|C|$.

So, this should be $2|C|$. So, what is this? This is also twice cardinality calc times epsilon. So, from here we get if I multiply both side with $\sum_{C \in C} |\delta(C) \cap F'| \epsilon$ is less than equal to $2|C|\epsilon$. So, you see the increase in LHS is at most the increase or in RHS. the increase in LHS is at most the increase in RHS.

and in the beginning of the iteration the inequality hold ah ah inequality holds that means, LHS is less than equal to RHS in the beginning of the iteration. we had LHS less than equal to RHS and in this iteration the effect net effect is RHS increase at least as much as increase in LHS. Hence LHS remains less than equal to RHS after the iteration which is exactly what we need to prove. So, what we have shown is this inequality that it is a two factor approximation algorithm, but we have to we have used this lemma. So, let us prove the lemma now.

For proving the lemma, we need to make an observation that at any point of the algorithm (V, F), F is the set of edges we are picking we have not considered yet the set of edges we will remove that will be the F'. So, even with (V, F) is a forest ok. This is a easy observation because the proof follows from the fact that whenever we pick an edge it is a boundary edge of a connected component. So, we pick only boundary edges we pick only some only ah one boundary edge of any connected component or any one connected component in every iteration ok. Hence, in (V, F) we cannot have a cycle because only we are only picking boundary edges good.

Now, let us prove the lemma proof of lemma. Let us write what do we need to prove? To prove that sum over current connected component C in **C**. $\sum_{C \in C} |\delta(C) \cap F'|$ is less than equal to $2|C|$ for every iteration ok. So, consider any iteration consider any iteration i arbitrary iteration for this iteration we will show this inequality. So, let $F_i = \{e_1, \ldots, e_i\}$ is the set of edges that are picked till iteration i.

or just before the beginning of iteration i, i - 1. So, this is F_i. So, this we will show at the beginning of every iteration ok and let us call the remaining edges to be H of F_i, $F' \setminus F_i$ that is H ok. So, note that $F_i \cup H$ is same as $F_i \cup F'$ and both are feasible solutions of the problem, both are feasible solution and they are same not both they are actually same. So, let us write this is a feasible solution ok.

Now, first an easy observation is that if we remove any edge $e \in H$ from this feasible solution $F_i \cup H$, then the solution becomes infeasible. why because of our clean up process this H, e was introduced or added to set F after i-th iteration or after (i - 1)-th iteration and hence it was not removed by the clean up process that means, H e is essential for maintaining feasibility of the solution ok. So, all the edges of H are necessary, all the edges of H are necessary for feasibility of the solution. So, now, let us draw the conected components the C. So, suppose these are the conected components.

So, this is C what you do we first contract each of the conected components into a single we contract each connected component of C into single vertex let V' be the set of contracted vertices. this there can be edges connecting these two and so on, but this will be a forest. So, after contraction it look like each set will be an one vertex, this is how it will look like if the graph is this. So, this is V'. Now, we see that you know in V' there are two kinds of vertices, one whose connected component has exactly one of s_i and t_i.

So, we call those vertices red vertices. So, we color the vertices of V' red or we color a vertex red. its corresponding connected component has exactly 1 of s_i and t_i for any $i \in [k]$ ok. So, some vertices are colored red vertices, other vertices are colored blue. Now, you can see that this inequality $\sum_{C \in C} |\delta(C) \cap F'|$ is less than equal to $2|C|$

is equivalent to showing something about this forest $V^{'}$ these are forest. So, there could be some isolated vertices there could be some other trees and so on. So, in terms of the forest the RHS is number of red vertices in $V^{'}$ ok. These are the exactly the number of nitrate components which is exactly one vertex in C and the left hand side the LHS is at most summation sum of degrees of red vertices. Each c is the degree of red vertex $\delta(C)\cap F^{'}$ is the is at most the degree of the red vertex.

So, $v\in R$ degree v. So, it is enough to show enough to show that sum of degrees of red vertices is less than equal to twice number of red vertices which is exactly what we will show now ok. To prove the claim first observe that there we cannot have a degree 1 blue vertex. observe that we cannot have any blue vertex of degree 1 you prove it as a homework easy proof. So, the degree of every blue vertex is at least 2. So, then we can write summation sum of degrees of red vertices is sum of degrees of all vertices minus sum of degrees of blue vertices ok.

Now, sum of degrees of all vertices because it is a forest this is at most twice the number of vertices because it is a forest. So, this is less than equal to twice cardinality r plus cardinality b. minus because we cannot have a blue vertex of degree 1 each degree of every blue vertex is at least 2. So, this is less than equal to twice cardinality B. So, this is twice cardinality R which is exactly what we need to prove ok.

So, you prove this claim that we cannot have a blue vertex of degree 1 very easy and with this we have we conclude the proof of the lemma and thus the proof of approximation factor of our algorithm ok. So, let us stop here. Thank you.