Approximation Algorithm

Prof. Palash Dey

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

Week – 10

Lecture 47

Lecture 47 : Primal-dual Algorithm for Steiner Forest

Welcome. So, in the last class we have seen a $4 \log n$ factor approximation algorithm for minimum weighted feedback vertex set problem and we have used primer dual method and we have seen that unlike set cover algorithm the f factor approximation algorithm for set cover using primer dual method. for minimum weighted feedback vertex set it is necessary to carefully select some dual variable to increase. So, today we will see another very interesting problem which is called generalized Steiner tree or Steiner forest and here we will see that it is sometimes useful in a primal dual method to increase multiple dual variables simultaneously. So, let us begin. the problem is generalized Steiner tree also called Steiner forest.

So, what is the problem? The input is an undirected graph $G=(V,E)$ non-negative costs $c_e \geq 0$ for all edges E and k $s_i - t_i$ pairs k pairs of vertices $\{s_1,t_1\},\{s_2,t_2\},\ldots,\{s_k,t_k\}$ goal. Select a set of set F of edges of minimum total cost such that in the subgraph V , F there is a path between s_i and t_i for every $i \in [k]$. Select a set of edges of minimum total cost such that s_i and t_i every pair is connected.

So, for this problem we will see a primal dual algorithm with small approximation factor and I want to remind you that this is the algorithm who which we used as a black box as a subroutine in price collecting Steiner tree problem. in the deterministic rounding and randomized rounding of linear programs for the price collecting standard tree problem we have used this problem this algorithm as a black box. So, as the first step we first write the ILP formulation of the problem. ILP formulation ok. So, we have a variable for every edge.

$e \in E$ which takes value 1 if we pick e otherwise it takes value 0. So, obviously, we want to minimize $\sum c_e x_e$. So, minimize $\sum c_e x_e$ subject to. Now, I need to ensure that there is a path between every in the subgraph that I select. Now, we have seen this sort of constraints in price collecting standard tree also.

So, we use similar constraints that for every subset of vertices which has one s_i, but not t_i among the boundary edges I must pick at least one vertex. So, that is the idea. So, let us introduce the notation. So, let s_i be all the subset of vertices such that $|S \cap \{s_i, t_i\}| = 1$ means, S has either s_i or t_i, but not both and for s_i and t_i to get connected there must we must pick at least one boundary vertex. So, let us recall what is boundary vertex we denote by $\delta(S_i)$ this is the set of edges in E such that $e \cap S_i$ this cardinality is 1 here also the cardinality ok.

So, let us write down the constraint here subject to for all or let me write this way for all $S \subseteq V$ such that S belongs to S_i for some i. If I look at the boundary $e \in \delta(S)$ at least one of these edges must be picked x_e greater than equal to 1 and the usual constraint that for every edge $e \in E$ $x_e \in \{0, 1\}$. So, this is the exact formulation of the problem ILP opt equal to opt. Now, the LP relaxation standard LP relaxation I replace x_e in between 0, 1. Again because it is a minimization problem and the objective function is minimize $\sum_{e \in E} c_e x_e$ are non-negative.

We can again safely drop $x_e \leq 1$ this constraint. So, only this constraint is enough that $x_e \geq 0$. So, this is our primal LP. So, what we will do? We will write the dual LP. So, this is the primal LP.

So, let us write the dual LP. So, for every set we have a variable every set we have a constraint this will correspond to dual variables. So, for every set let us introduce a variable y_S corresponding to this constraint. So, we want to minimize maximize $\sum_{S \subseteq V : S = S_i \, for \, some \, i \in [k]} y_S$ this we want to maximize subject to the constraint for every edge we have a constraint for every edge $e \in E$ sum over y_S such that $e \in \delta(S)$ this is less than equal to c_e and for all S such that there exist an i with S belongs to S_i y_S all these variables are non-negative.

So, this is the dual and again let us begin with trying the standard primal dual framework. So, recall in the standard primal dual framework, we start with a dual feasible solution and a trivial dual feasible solution is setting all y_S to be 0. but before that here again we see that we have an exponentially many variables, but like feedback vertex set problem all these variables will be 0 except polynomially many. So, storing a solution dual solution will not be computationally challenging. So, start with dual feasible solution y equal to 0.

That means, y_S equal to 0 for all such S. So, these are dual feasible solution and partial primal solution F equal to empty set. So, it is infeasible. So, what we do? We pick some dual variable, what are the natural dual variables? Natural dual variables are those sets in G comma F, which has one which has exactly one of s_i or t_i for some i, but initially this F is empty set. So, the graph is disconnected it just has n isolated vertices.

So, in the first iteration these sets are we have only n sets because those are the connected components not n sets, but a natural let me write a natural primal dual algorithm considers the set **C** of connected components of $G^{'}=(V,F)$. which contains exactly one of s_i and t_i for sum $i\in[k]$ ok. So, these are the variables and we pick any corresponding dual variables and such dual variable and increase its value until one constraint become tight one dual constraint become tight C in **C** be any such component, we increase y_C until some dual constraint becomes tight.

and dual constraints corresponds to edges of the graph here you see for every edge we have a dual constraint. So, we pick such a tight edge in F and repeat until F is a Steiner forest ok. So, this is the standard primal dual framework. Let us write down its pseudocode y equal to 0 and F equal to empty set while not all s_i-t_i pairs are connected in (V,F), what do we do? Let C be a connected component of (V,F) such that there exists an $i\in[k]$ such that $|C\cap\{s_i,t_i\}|=1$ That means, it contains exactly 1 of s_i and t_i, then increase dual variable y_C until there is an edge $e\in\delta(C)$ such that summation the corresponding dual constraint becomes tight $\sum_{S\in S_i:e\in\delta(S)}y_S=c_e$. for any $i\in[k]$ ok. So, then next what we do we include this tight edge in our solution is $F\cup\{e\}$ and while loop ends then return F. Now, let us try to analyze using standard primal dual analysis. So, ALG is the sum of weights of the edges in F that is $\sum_{e\in F}c_e$.

Now, all these edges the corresponding dual constraint is tight. So, this is the corresponding dual constraint. So, we replace c_e with $\sum y_S$. So, this is equal to $\sum_{e\in F}\sum_{S:e\in\delta(S)}y_S$ as usual we exchange the double sum the order of the summations summation s and then which edges for every S. how many edges I am picking in my solution which are in $\delta(S)$.

So, that is delta s intersection f that many time y_S is appearing. Now, again to have an approximation algorithm we need this is we need or let me write this way need $\delta(S)\cap F$ if this is less than equal to alpha then this is alpha times summation sum over s y s, but this is dual of this is a dual solution by weak duality this is less than equal to LP opt which is less than equal to opt, but this is problematic. In the next lecture we will see that we do not hope to get a good bound uniformly for every s which bounds cardinality of $\delta(S)\cap F$ ok. So, let us stop here.