COL 754

Homework III

Due on 20th November, 2018

All questions are from the textbook.

1. Consier the multi-cut problem on trees. In this problem, we are given a tree $T = (V, E)$, k pairs of vertices s_i-t_i, and costs $c_e \geq 0$ for each edge $e \in E$. The goal is to find a minimum-cost set of edges F such that for all i, s_i and t_i are in different connected components of $G' = (V, E' - F)$. Let P_i be the set of edges in the unique path in T between s_i and t_i . Then we can formulate the problem as the following integer program:

$$\min . \sum_{e \in E} c_e x_e$$

$$\sum_{e \in P_i} x_e \geq 1 \qquad 1 \leq i \leq k$$

$$x_e \in \{0, 1\} \qquad e \in E.$$

   Suppose we root the tree at an arbitrary vertex r. Let $depth(v)$ be the number of edges on the path from v to r. Let $lca(s_i, t_i)$ be the vertex v on the path from s_i to t_i whose depth is minimum. Suppose we use the primal-dual method to solve this problem, where the dual variable we increase in each iteration corresponds to the violated constraint that maximizes $depth(lca(s_i, t_i))$. Prove that this gives a 2-approximation algorithm for the multicut problem in trees.

2. In the 2-approximation primal-dual algorithm for the Steiner forest problem that we discussed in class, we first added all tight edges, then remove unnecessary edges in the order opposite of the order in which they were added. Prove that one can in fact remove unnecessary edges in any order and still obtain a 2- approximation algorithm for the problem. That is, we replace the reverse edge removal steps in the algorithm with a step that checks if there exists any edge $e \in F'$ such that $F' - e$ is feasible (initially F' is the set of all tight edges). If so, e is removed from F' , and if not, F' is returned as the final solution. Prove that $\sum_{e \in F'} c_e \leq 2 \sum_S y_S$ for the dual y generated by the algorithm.

3. In the minimum-cost branching problem we are given a directed graph $G = (V, A)$, a root vertex $r \in V$, and weights $w_{ij} \geq 0$ for all $(i, j) \in A$. The goal of the problem is to find a minimum-cost set of arcs $F \subseteq A$ such that for every $v \in V$, there is exactly one directed path in F from r to v. Use the primal-dual method to give an optimal algorithm for this problem.

4. As with linear programs, semidefinite programs also have duals. The dual of the MAX-CUT SDP (assume the graph is a complete graph) is

$$\begin{aligned} \min. \quad & \frac{1}{2}\sum_{i<j} w_{ij} + \frac{1}{4}\sum_i \gamma_i \\ & W + diag(\gamma) \succeq 0, \end{aligned}$$

where the matrix W is the symmetric matrix of the edge weights w_{ij} and $diag(\gamma)$ is the matrix of zeroes with γ_i as the ith entry on the diagonal. Show that the value of any feasible solution for this dual is an upper bound on the cost of any cut.

5. Semidefinite programming can also be used to give improved approximation algorithms for the maximum satisfiability problem. First we start with the MAX 2SAT problem, in which every clause has at most two literals.

   (a) As in the case of the maximum cut problem, wed like to express the MAX 2SAT problem as a integer quadratic program in which the only constraints are $y_i \in \{1, 1\}$ and the objective function is quadratic in the y_i . Show that the MAX 2SAT problem can be expressed this way. (Hint: it may help to introduce a variable y_0 which indicates whether the value 1 or 1 is TRUE).

   (b) Derive a .878-approximation algorithm for the MAX 2SAT problem.

   (c) Use this .878-approximation algorithm for MAX 2SAT to derive a $(3/4 + \epsilon)$-approximation algorithm for the maximum satisfiability problem, for some $\epsilon > 0$. How large an ϵ can you get ?