TU Berlin
Institut für Mathematik
Prof. David Williamson
Jannik Matuschke

Approximation Algorithms (ADM III)
Winter term 2010

Problem Set 5

(due date: *January 12, 2011*)

Exercise 5.1 **5 points**

An important quantity in combinatorial optimization is called the *Lovász Theta Function.* The theta function is defined on undirected graphs $G = (V, E)$. One of its many definitions is given below as a semidefinite program:

$$\begin{aligned} \vartheta(\bar{G}) = \text{maximize} \quad & \sum_{i,j} b_{ij} \\ \text{subject to} \quad & \sum_i b_{ii} = 1, \\ & b_{ij} = 0, \qquad \forall i \neq j, (i,j) \notin E, \\ & B = (b_{ij}) \succeq 0, \qquad B \text{ symmetric.} \end{aligned}$$

Lovász showed that $\omega(G) \leq \vartheta(\bar{G}) \leq \chi(G)$, where $\omega(G)$ is the size of the largest clique in G and $\chi(G)$ is the minimum number of colors needed to color G.

(a) Show that $\omega(G) \leq \vartheta(\bar{G})$.

(b) The following is a small variation in the vector program we used for graph coloring:

$$\begin{aligned} \text{minimize} \quad & \alpha \\ \text{subject to} \quad & v_i \cdot v_j = \alpha, \qquad \forall (i,j) \in E, \\ & v_i \cdot v_i = 1, \qquad \forall i \in V, \\ & v_i \in \mathbb{R}^n, \qquad \forall i \in V. \end{aligned}$$

Its dual is

$$\begin{aligned} \text{maximize} \quad & -\sum_i u_i \cdot u_i \\ \text{subject to} \quad & \sum_{i \neq j} u_i \cdot u_j \geq 1, \\ & u_i \cdot u_j = 0, \qquad \forall (i,j) \notin E, i \neq j, \\ & u_i \in \mathbb{R}^n \qquad \forall i \in V. \end{aligned}$$

Show that the value of the dual is $1/(1 - \vartheta(\bar{G}))$. By strong duality, this is also the value of the primal; however, see the chapter notes in the book for a discussion of conditions under which strong duality holds.

The value of this vector program is sometimes called the *strict vector chromatic number* of the graph, and the value of original vector programming relaxation used in class is the *vector chromatic number* of the graph.

Exercise 5.2 **5 points**

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, we'd 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 $(\frac{3}{4} + \epsilon)$-approximation algorithm for the maximum satisfiability problem, for some $\epsilon > 0$. How large an ϵ can you get?

Exercise 5.3 **5 points**

Consider the *multicut problem in 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:

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} c_e x_e \\ \text{subject to} \quad & \sum_{e \in P_i} x_e \geq 1, \qquad & 1 \leq i \leq k, \\ & x_e \in \{0, 1\}, \qquad & e \in E. \end{aligned}$$

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.

Exercise 5.4 **5 points**

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.