COMP 362 - Winter 2017 - Assignment 5

Due: 6pm April 6th

General rules: In solving these questions you may consult your book; you can discuss high level ideas with each other. But each student must find and write his/her own solution. You should drop your solutions in the assignment drop-off box located in the Trottier Building on the 2nd floor.

1. Consider the integer program for vertex cover problem that was discussed in class.

   (a) (5 Points) Suppose G has a triangle (three pairwise adjacent vertices x_1, x_2, x_3). Then we can add to the integer program the constraint $x_1 + x_2 + x_3 \geq 2$. Why is this constraint valid?

   Solution: A vertex cover must pick at least two vertices from every triangle.

   (b) (5 Points) Generalize the above to an odd cycle C. In other words, write a valid inequality associated with C that can be added to the integer program that still describes the original problem, but which will allow for a tighter relaxation when the integer program is relaxed to a linear program.

   Solution: A vertex cover must pick at least $k+1$ vertices from every cycle of lenth $2k+1$. Hence for every such cycle with vertices $x_1, \ldots, x_{2k+1}$, we can add the constraint $x_1 + \ldots + x_{2k+1} \geq k+1$.

   (c) (10 Points) Suppose we add all constraints for all odd cycles. Let H be a graph on n vertices, that has no odd cycle of size at most $\log(n)/10$ and whose minimum vertex cover is of size $n - o(n)$. The existence of such graphs can be shown using probabilistic methods. Consider the integer program for H that is obtained by adding <u>all</u> odd-cycle constraints to the original integer program for the vertex cover. Show that even this integer program has integrality gap[1] of $2 - o(1)$.

[1] integrality gap is the maximum ratio between the optimal solutions to the integer program, and its linear program relaxation.

Solution: Consider the solution $x_i^* = \frac{1}{2} + \frac{10}{\log(n)}$ for every vertex i. Note that this is a feasible solution: For every edge uv we have $x_u^* + x_v^* = 2(\frac{1}{2} + \frac{10}{\log(n)}) \geq 1$. On the other hand the length $2k+1$ of every odd cycle C satisfies $2k+1 \geq \log(n)/10$. Thus for every such cycle

$$x_1^* + \ldots + x_{2k+1}^* \geq (2k+1)\left(\frac{1}{2} + \frac{10}{\log(n)}\right) = k + \frac{1}{2} + (2k+1)\frac{10}{\log(n)} \geq k+1.$$

Note that the cost of this feasible solution is $n(\frac{1}{2} + \frac{10}{\log(n)}) = \frac{n}{2} - o(n)$ while the optimal solution is $n - o(n)$. Hence the integrality gap is at least

$$\frac{n - o(n)}{\frac{n}{2} - o(n)} = 2 - o(1).$$

2. (20 Points) Given a set P of n points on the plane, consider the problem of finding the smallest r such that there exist 10 circles of radius r such that together they contain all the points in P. Design a PTAS algorithm for this problem. In other words, given any fixed $\epsilon > 0$, design an algorithm whose running time is polynomial in n, and its output is at most $1+\epsilon$ times the optimal output.

   Solution: If $n \leq 10$ we are done. Otherwise we first run the 2-factor approximation algorithm for the k-center problem (with $k = 10$). Let t be the output of that algorithm. Then we know that the optimal radius r^* satisfies $t/2 \leq r^* \leq t$.

   Next divide the part of the plane that contains the points into a gird whose cells are $\frac{t\epsilon}{4} \times \frac{t\epsilon}{4}$. In the optimal solution every center is in distance at most t from at least one point in P (if that is not the case, then since $r^* \leq t$, the corresponding circle will not contain any points). Now if we consider all the grid points that are in distance at most t from at least one point in P then in total we have $\frac{8}{\epsilon} \times \frac{8}{\epsilon} \times n$ such points. If we try all the possibilities of choosing 10 centers among them then we have at most $(\frac{8}{\epsilon} \times \frac{8}{\epsilon} \times n)^{10}$ choices. We pick the best one and output it.

   It remains to show that our output radius is not larger than $(1+\epsilon)r^*$. Let $c_1, \ldots, c_{10}$ be the optimal centers. Let $c_1', \ldots, c_{10}'$ be such that c_i' is one of the corners of the grid cell that contains c_i. Note that the distance between c_i and c_i' is at most $\sqrt{2}\frac{t\epsilon}{4} \leq \frac{t\epsilon}{2}$. At some point our algorithm has checked $c_1', \ldots, c_{10}'$, and since $\text{dist}(c_i, c_i') \leq \frac{t\epsilon}{2} \leq r^*\epsilon$, for that particular choice of centers it suffices to consider circles of radius at most $r^* + r^*\epsilon \le (1+\epsilon)r^*$. Hence the output of the algorithm is at most $(1+\epsilon)r^*$.

3. Consider the MAX-SAT problem: Given a 2-CNF ϕ (clauses of the form $(x_i \vee x_i)$ and $(x_i \vee \bar{x_i})$ are not allowed) on n variables $x_1, \ldots, x_n$, we want to find a truth assignment that satisfies the maximum number of clauses.

   (a) (10 Points) As in the case of the MAX-CUT problem, find an "integer quadratic program" formulation of this problem in which the only constraints are $y_i \in \{-1, 1\}$. (Hint: It may help to introduce an auxiliary variable y_0 which indicates whether "True" is associated with -1 or with 1.)

   Solution: Introduce variables $y_i \in \{\pm 1\}$ for each boolean variable x_i and also also introduce an extra variable $y_0 \in \{\pm 1\}$ interpreted in the following way. We will set $x_i = T$ if $y_i = y_0$ and $x_i = F$ otherwise. For every clause involving two variables x_i and x_j we will consider a formula

$$1 - \left(\frac{1 \pm y_i y_0}{2}\right)\left(\frac{1 \pm y_j y_0}{2}\right) = \left(\frac{1}{4} \pm \frac{y_i y_0}{4}\right) + \left(\frac{1}{4} \pm \frac{y_j y_0}{4}\right) + \left(\frac{1}{4} \pm \frac{y_i y_j}{4}\right), \tag{1}$$

   where the $\pm$ signs on the left hand side are picked according to whether the variable is appear as itself or negated in the clause ($-y_i y_0$ if it appears as itself and $-y_i y_0$ if it is negated). For example the clause $(x_i \vee \overline{x_j})$ corresponds to

$$1 - \left(\frac{1 - y_i y_0}{2}\right)\left(\frac{1 + y_j y_0}{2}\right) = \left(\frac{1}{4} - \frac{y_i y_0}{4}\right) + \left(\frac{1}{4} + \frac{y_j y_0}{4}\right) + \left(\frac{1}{4} - \frac{y_i y_j}{4}\right).$$

   Note that this expression is equal to 1 if the clause is satisfied and it is equal to 0 otherwise. This is because $\frac{1 - y_i y_0}{2} = 0$ if x_i is True (i.e. $y_i = y_0$) and it is 1 otherwise, and similarly $\frac{1 + y_i y_0}{2} = 0$ if x_i is False and it is 1 otherwise.

   Now if we sum the expression in the right hand side of (1) over all clauses we obtain an expression $\Phi = \frac{1}{4}\sum_{i<j} \alpha_{ij}(1 + y_i y_j)$ where the coefficients $\alpha_{ij} \in \mathbb{Z}_{\ge 0}$. Note that Φ counts the number of satisfied clauses. Hence the corresponding quadratic program will maximize the number of satisfied clauses.

$$\begin{array}{ll} \max & \frac{1}{4}\sum_{i<j} \alpha_{ij}(1 - y_i y_j) + \sum_{i<j} \beta_{ij}(1 + y_i y_j)) \\ \text{s.t.} & y_i^2 = 1 \qquad\qquad\qquad\qquad\qquad\qquad \forall i = 0, \ldots, n \end{array}$$

(b) (10 Points) Formulate the SDP relaxation of your integer quadratic program, and use it to find a randomized 0.878-factor approximation algorithm for the MAX-2-SAT problem.

Solution: The SDP relaxation is

$$\begin{array}{lll} \max & \frac{1}{4}\left(\sum_{i<j} \alpha_{ij}(1-\langle v_i, v_j\rangle) + \sum_{i<j} \beta_{ij}(1+\langle v_i, v_j\rangle)\right) & \\ \text{s.t.} & \langle v_i, v_i\rangle = 1 & \forall i = 0, \ldots, n \end{array}$$

Now to round the solution of the SDP to a solution for the quadratic program we pick a random unit vector $w \in \mathbb{R}^n$ and for $i = 0, \ldots, n$ we set $y_i = 1$ if $\langle v_i, w\rangle \geq 0$ and $y_i = -1$ if $\langle v_i, w\rangle < 0$. Now from the analysis of the SDP rounding for MAX-CUT we know that

$$\frac{\mathbb{E}[1 - y_i y_j]}{1 - \langle v_i, v_j\rangle} \geq 0.878,$$

and using a similar analysis we can see that

$$\frac{\mathbb{E}[1 + y_i y_j]}{1 + \langle v_i, v_j\rangle} \geq 0.878.$$

This as in in the analysis of MAX-CUT implies

$$\frac{\mathbb{E}[\sum_{i<j} \alpha_{ij}(1 - y_i y_j) + \sum_{i<j} \beta_{ij}(1 + y_i y_j)]}{\sum_{i<j} \alpha_{ij}(1-\langle v_i, v_j\rangle) + \sum_{i<j} \beta_{ij}(1+\langle v_i, v_j\rangle)} \geq 0.878.$$

Since the optimal answer to the SDP is at least as large as the optimal solution to 2-SAT we conclude that the expected value of the rounded solution is at least 0.878 times the optimal solution.

4. (20 Points) Let $v_1, \ldots, v_n$ be *unit* vectors in $\mathbb{R}^n$. Prove that there exists $\epsilon_1, \ldots, \epsilon_n = \pm 1$ such that

$$|\epsilon_1 v_1 + \ldots + \epsilon_n v_n| \leq \sqrt{n}.$$

Hint:

$$|\epsilon_1 v_1 + \ldots + \epsilon_n v_n|^2 = \langle \epsilon_1 v_1 + \ldots + \epsilon_n v_n, \epsilon_1 v_1 + \ldots + \epsilon_n v_n\rangle.$$

Solution: Let X be the random variable $|\epsilon_1 v_1 + \ldots + \epsilon_n v_n|^2$. By the hint we know that

$$X = \sum_{i,j=1}^{n} \epsilon_i \epsilon_j v_i \cdot v_j.$$

Every ϵ_i is uniformly either -1 or 1, hence its expectation is simply zero. On the other hand, ϵ_i^2 is constant one. Thus, using the independence of ϵ_i's and the linearity of expectation we obtain the following.

$$\mathbb{E}[X] = \sum_{i,j=1}^{n} \mathbb{E}[\epsilon_i \epsilon_j] v_i \cdot v_j = \sum_{i \neq j} \mathbb{E}[\epsilon_i \epsilon_j] v_i \cdot v_j + \sum_{i=1}^{n} \mathbb{E}[\epsilon_i^2] v_i \cdot v_i = \sum_{i=1}^{n} v_i \cdot v_i = n.$$

Thus, there will be a choice of ϵ_i's for which $X \leq n$, as desired.

5. (20 Points) Give a polynomial algorithm for the following problem: Let G be a graph with a given proper k-vertex coloring of G. Find a vertex cover whose size is at most $(2 - \frac{2}{k})$OPT, where OPT is the size of the smallest vertex cover for G.

Solution: Consider the LP relaxation for vertex cover.

$$\begin{array}{lll} \max & \sum_{u \in V} x_u & \\ \text{s.t.} & x_u + x_v \geq 1 & \forall uv \in E \\ & x_u \geq 0 & \forall u \in V \end{array}$$

First note that the extreme points of the LPassign only $x_u \in \{0, \frac{1}{2}, 1\}$ to variables. Indeed if x is a feasible solution that does not satisfy this property, then let $V^+ = \{u : 0.5 < x_u < 1\}$ and $V^- = \{u : 0 < x_u < 0.5\}$, and consider

$$z_u = \begin{cases} x_u + \epsilon & u \in V^+ \\ x_u - \epsilon & u \in V^- \\ x_u & \text{otherwise} \end{cases}$$

and

$$y_u = \begin{cases} x_u - \epsilon & u \in V^+ \\ x_u + \epsilon & u \in V^- \\ x_u & \text{otherwise} \end{cases}$$

for sufficiently small $\epsilon > 0$. Note that $x = \frac{y+z}{2}$, but if $\epsilon > 0$ is sufficiently small, then y and z are both feasible. The reason for this is that if $x_u + x_v = 1$, then if $u \in V^+$, we have $v \in V^-$ and vice versa. So in both cases $y_u + y_v = z_u + z_v = 1$.

Now that we know that the extreme points satisfy $x_u \in \{0, \frac{1}{2}, 1\}$, we can state our approximation algorithm.

- Solve the Linear Program and find an point optimal solution x with $x_u \in \{0, \frac{1}{2}, 1\}$ for all u.
- Out of the k colors find the one color j that has largest number of variables with value $\frac{1}{2}$. Pick all the vertices with $x_u = 1$ together with all the vertices with $x_u = \frac{1}{2}$ except those with color j.

Note that this is a vertex cover because every edge uv satisfies $x_u + x_v \geq 1$, and either $x_u = 1$ or $x_v = 1$ in which case one of the end points whill be picked, or $x_u = x_v = \frac{1}{2}$ in which case at least one of u or v is not colored by j and it will be picked.