Advanced Approximation Algorithms **CMU 15-854B, Spring 2008**

HOMEWORK 1
Due: Tuesday, January 29

---

1. Randomized approximation algorithms. Suppose A is a *randomized* algorithm for the NP optimization problem Max-Blah and has the following properties:

i The expected running time of A is at most poly(n).

ii When Opt $\geq c$, with probability at least $1/\text{poly}(n)$ algorithm A outputs a solution of value at least s.

a) Give a randomized algorithm B which runs in poly(n) time with certainty and has the property that when Opt $\geq c$, algorithm B outputs a solution of value at least s with probability at least $1 - 2^{-n}$.

b) Assume Max-Blah solution values are always in the range $[0, \text{poly}(n)]$. Suppose algorithm A' has the property that when Opt $\geq c$, the *expected* value of the solution output by A' is at least s. (Recall that this is our notion of a randomized algorithm solving the c vs. s search problem.) Show that for any constant a, algorithm A' outputs a solution of value at least $s - 1/n^a$ with probability at least $1/\text{poly}(n)$ (and hence part (a) is essentially applicable).

2. Johnson's Algorithm (and a derandomization).

a) Suppose we solve Max-Cut by 2-coloring the graph randomly (vertices' colors are chosen uniformly and independently.) Show that this is an absolute 1/2 approximation algorithm. Deduce that every graph has a Max-Cut of at least 1/2 of the edges.

b) Here is a greedy algorithm for Max-Cut: Order the vertices $v_1, \ldots, v_n$. Color v_1 with color 1. Now for each subsequent vertex, color it 1 or 2 so as to maximize the number of edges cut thus far. Show that this (deterministic) algorithm is also an absolute 1/2 approximation algorithm.

c) Let Max-$\geq k$Sat be the same as Max-kSat except that each clause involves *at least* k literals. Give a randomized absolute $1 - 2^{-k}$ approximation search algorithm.

d) Give a randomized absolute $1/|K|$ approximation search algorithm for Label-Cover(K, L).

3. APX-hardness reductions. The PCP Theorem shows that for some absolute constant $\epsilon_0 > 0$, the 1 vs. $1 - \epsilon_0$ decision problem for Max-3Sat is NP-hard. In fact, it shows this even for Max-E3Sat-6 (see Problem Definitions handout). Using only this fact...

a) Show there is no PTAS for Max-Independent-Set. (Hint: textbook reduction.) Deduce that for all constant $\delta > 0$ the factor-δ decision problem is NP-hard. (Hint: graph products.)

b) Show that for some constant $\epsilon > 0$, the 1 vs. $1-\epsilon$ decision problem for Label-Cover([2], [7]) is NP-hard, even when the following extra condtions on the input hold: the bipartite graph is regular on the left, regular on the right, and $|V|$ is an integer multiple of $|U|$. ($[k]$ denotes $\{1, 2, \ldots, k\}$.)

4. More hardness reductions.

a) Show that for all finite C, the factor-C approximating Min-TSP is NP-hard. (Hint: reduce form Hamiltonian-Path.)

b) Håstad (building on work by Trevisan-Sorkin-Sudan-Williamson) has shown that the 17/21 vs. $16/21 + \epsilon$ decision problem for Max-Cut is NP-hard, for all $\epsilon > 0$. Show that for all constant $c < 5/4$ and all small δ, the $1-\delta$ vs. $1-c\delta$ decision problem is NP-hard.

c) Raz's Theorem shows that for all constant $\eta > 0$, there exists a large enough constant $q = q(\eta)$ such that the 1 vs. η decision problem for Label-Cover(K, L) is NP-hard with $|K|, |L| \leq q$ — even with the extra conditions from (3b) holding. Show that this hardness result still holds even if we additionally require $|U| = |V|$.

5. Greedy algorithm twists.

a) Modify the greedy algorithm for Set-Cover so that it achieves a $(\lceil \ln(n/\text{Opt}) \rceil + 1)$-factor approximation.

b) Show that the greedy algorithm for Max-Coverage is a $(1 - 1/e)$-factor approximation.

6. Greedy for weighted Set Cover. Consider the following "bang-for-the-buck" greedy algorithm for weighted Set Cover: At each stage, choose the set S which minimizes

$$\frac{c(S)}{\text{uncovered elements that } S \text{ would cover}}.$$

a) Show that this gives a H_D-factor approximation algorithm, where $D = \max_S |S| \leq n$ and $H_D = 1 + 1/2 + 1/3 + \cdots + 1/D$. (Hint: introduce the "price" $p(e)$ of each element e, equal to the bang-for-the-buck being achieved when the algorithm first covers e.)

b) Show a matching algorithmic gap instance. (Hint: use $D + 1$ sets over D ground elements.)

7. Optimal 1 vs. $1 - 1/e$ hardness for Max-Coverage. Solve one of the following:

a) Using Raz's Theorem, show that for all constant $\eta > 0$ and integers $k \geq 2$, there exists a large enough constant $q = q(\eta)$ such that given a "regular" k-ary-Consistent-Labeling(K, L) instance H with $|K|, |L| \leq q$, it is NP-hard to distinguish the case that there is a labeling with strong value 1 from the case that every labeling has weak value less than η. Here "regular" means that there is some d such that every $v \in V$ occurs as the ith vertex in a "hyperedge" e exactly d times, $i = 1 \ldots k$. (Hint: given $G = (U, V, E)$, consider all k-tuples from E of the form $[(u, v_1), \ldots, (u, v_k)]$.)

b) Using part (a), show that for all constant $k \geq 2$, $\epsilon > 0$, the 1 vs. $1 - (1 - 1/k)^k + \epsilon$ decision problem for Max-Coverage is NP-hard. (Hint: similar to the reduction from class, using the gadget $\{1, 2, \ldots, k\}^K$.)