cs473: Algorithms Assigned: *Tue., Dec. 10, 2019*

Problem Set #11 (**optional**, *not* for submission)

Prof. Michael A. Forbes
Prof. Chandra Chekuri Due: *never*

This pset is **optional**, *not* for submission.

---

1. In the MaxSAT problem we are given a boolean CNF formula φ and the goal is to find an assignment that satisfies the maximum number of clauses. Consider an oblivious randomized algorithm that sets each variable independently to TRUE with probability exactly $\frac{1}{2}$.

   (a) Some clauses such as $(x \vee \neg x)$ are always satisfied by *any* input, which we call *trivially satisfiable*. Reduce the general MaxSAT problem to the case where there are no trivially satisfiable clauses. We will call such φ *non-trivial*.

   (b) Suppose the formula is a non-trivial k-CNF formula where each clause has *exactly* k *distinct* literals. What is the expected number of clauses satisfied by a random assignment?

   Note: For any constant k, obtaining an approximation ratio better (by any fixed constant) than the ratio obtained by this algorithm is NP-hard.

   (c) Prove that for a general non-trivial CNF formula, the expected number of clauses that are satisfied is at least $\frac{m}{2}$ where m is the number of clauses.

2. We saw an LP-based 2-approximation for weighted vertex-cover. Write an LP relaxation for weighted set-cover. Recall that we are given m sets $S_1, S_2, \ldots, S_m$ over a universe of size n, and each set S_i has a non-negative weight $w_i \in \mathbb{Z}_{\geq 0}$. The goal is to find a minimum weight sub-collection of the sets which together cover the universe. Obtain a k-approximation for instances in which each element is contained in at most k sets.

   Note: Vertex-cover is the special case when $k = 2$.

3. Consider the Max-k-Cover problem, which is a variant of the set-cover problem. The input consists of a collection of m sets $S_1, S_2, \ldots, S_m$ of a universe of n elements, and an integer k. The goal is to pick k of the given sets to maximize the number of elements covered.

   (a) Describe a simple greedy algorithm for this problem.

   (b) Use the analysis of the greedy algorithm for set-cover to obtain a $(1 - 1/e)$-approximation for Max-k-Cover.