6.841: Advanced Complexity Theory Spring 2016

Problem Set 4 – PCPs

Due Date: April 21, 2016

Problem 1 – PCP basics

Prove the following statements:

a) $PCP_{c,s}[r, q]_\Sigma \subseteq PCP_{c,s}[r, q \cdot \log |\Sigma|]_{\{0,1\}}$.

b) $PCP_{1,s}[r, q]_{\{0,1\}} \subseteq PCP_{1,1-\frac{1}{q}+\frac{s}{q}}[r + \log q, 2]_{\{0,1\}^q}$.

c) If $|\Sigma| \leq \text{poly}(n)$, then $PCP_{c,s}[O(\log n), 1]_\Sigma \subseteq \mathsf{P}$.

Problem 2 – Algorithms for MAX-SAT

a) Give a randomized algorithm that, given a 2CNF formula ψ with exactly 2 distinct literals per clause, outputs an assignment that satisfies at least a 3/4 fraction of ψ's clauses.

b) Give a *deterministic* algorithm that, given a 3CNF formula ψ with exactly 3 distinct literals per clause, outputs an assignment that satisfies at least a 7/8 fraction of ψ's clauses.

Problem 3: Hardness for CLIQUE

In class, we saw how to prove that it is NP-hard to approximate independent set to within a constant factor by using the [FGLSS] reduction. By complementing the graph, this gives the same hardness of approximation result for clique. In this problem, we will see a slightly different way to prove that it is NP-hard to approximate clique to within a constant factor.

Given a graph $G = (V, E)$ and an integer k, define the kth power of G, $G^k = (V', E')$ to be as follows. The vertex set V' is V^k, the set of k-tuples of vertices from V. Two distinct vertices $(u_1, \ldots, u_k)$ and $(v_1, \ldots, v_k)$ have an edge between them in E' iff $\{u_1, \ldots, u_k, v_1, \ldots, v_k\}$ is a clique in G.

Define $\omega(G)$ to be the size of the largest clique in G.

a) Show that $\omega(G^k) = \omega(G)^k$.

b) We know from the PCP Theorem that it is NP-hard to ρ-approximate CLIQUE for some constant ρ. Use this with part a) to show that, for any constant ρ', there is no ρ'-approximation algorithm for CLIQUE unless P=NP. [1]

[1] Under stronger assumptions, we can use this method to get an even better result. For example, unless NP $\subseteq \bigcup_{c \geq 1} \text{DTIME}(2^{(\log n)^c})$, CLIQUE does not admit a polynomial time $2^{-\log^\gamma(n)}$-approximation algorithm.

Problem 4 – Hardness of Approximation from Håstad

In this problem, we will use a version of the PCP Theorem proved by Håstad: completeness is $1-\varepsilon$, soundness is $1/2+\varepsilon$, the number of queries is 3, and all predicates ψ the verifier uses are of the form $x_{i_1}+x_{i_2}+x_{i_3}=b \mod 2$, where b is 0 or 1, and ε can be taken to be any positive constant.

a) Let MAX-3LIN be the maximization problem where the input is a set of 3-variable linear equations mod 2 and the goal is to find an assignment satisfying as many equations as possible. Show that for any $\varepsilon > 0$, there is no $(1/2+\varepsilon)$-approximation algorithm for MAX-3LIN unless P=NP.

b) Assuming P $\neq$ NP, show that we cannot improve Håstad's PCP Theorem to have completeness 1 while preserving the other parameters.

One of the reasons that we like Håstad's PCP so much is not only that it gives an optimal hardness of approximation result for MAX-3LIN, but also that it allows us to get hardness of approximation results for many other problems, like the following:

- **MAX-E3SAT** is the maximization problem where the input is a CNF where each clause has exactly three literals and the goal is to find an assignment satisfying as many clauses as possible.
- **MAX-3MAJ** is the optimization problem where the input is a set of constraints over 3 boolean literals, where each constraint asserts that the majority of its three literals' values is 1.
- **MAX-2SAT** be the problem of computing the maximum number of satisfiable clauses in a 2-CNF instance, where each clause contains at most 2 literals.

We will now use Håstad's result to prove hardness of approximation results for each of these problems.

c) Show that for any $\varepsilon > 0$, there is no $(7/8+\varepsilon)$-approximation algorithm for MAX-E3SAT unless P=NP. (Hint: Reduce from MAX-3LIN)

d) Show that for any $\varepsilon > 0$, there is no $(2/3+\varepsilon)$-approximation algorithm for MAX-3MAJ unless P=NP. (Hint: Reduce from MAX-3LIN)[2]

e) Show that there is an α such that $.99 > \alpha > 3/4$ such that it is NP-hard to approximate MAX-2SAT within a factor of α. (Hint: Reduce from MAX-E3SAT)

Problem 5 (Optional): A "Long Code" Test

This problem is meant as an introduction to use of Fourier analysis in complexity theory. Although, the problem is optional and will **not** be graded, you are encouraged to work on it for your own benefit and discuss it with us or each other.
Let $[n]=\{1,\ldots,n\}$. For $S \subseteq [n]$, define $\chi_S : \{-1,1\}^n \to \mathbb{R}$ as $\chi_S(x)=\prod_{i\in S} x_i$.[3] It is not hard to see that

$$\forall S \neq T, \qquad \sum_x \chi_S(x)\chi_T(x) = 0,$$

[2] In fact, there is a 2/3-approximation algorithm for MAX-3MAJ, making this result tight.

[3] For this problem we work with the representation of Boolean hypercube as $\{-1,1\}^n$. One could equivalently work with the $\{0,1\}^n$ representation; for this simply change the definition of $\chi_S(x)$ to $(-1)^{\sum_i x_i}$. The $\{-1,1\}^n$ representation however is usually more convenient.

and hence the set of functions $\{\chi_S : S \subseteq [n]\}$ form an orthonormal basis for the vector space of real-valued functions over the Boolean hypercube (with respect to the inner product $\langle f, g\rangle = \frac{1}{2^n}\sum_x f(x)g(x)$). Hence, every function $f : \{-1,1\}^n \to \mathbb{R}$ can be written as linear combinations of χ_S's as

$$f(x) = \sum_{S\subseteq[n]} \hat{f}(S)\chi_S(x), \tag{1}$$

where the $\hat{f}(S)$'s are called the *Fourier coefficients* of f.[4]

i) Show that $\hat{f}(S)$ as defined by Eq. (1) satisfies $\hat{f}(S) = \mathbb{E}_x f(x)\chi_S(x)$, where the expectation is taken with respect to the **uniform distribution**.

ii) Show that $\sum_S \hat{f}(S)^2 = \mathbb{E}_x f(x)^2$.

iii) [BLR test] Let $f : \{-1,1\}^n \to \{-1,1\}$. Define ϵ as

$$\Pr_{x,y}[f(x)f(y) = f(x\cdot y)] = 1-\epsilon.$$

where $x \cdot y$ denotes the entry-wise multiplication of x and y.

Show that $1-2\epsilon = \sum_{S\subseteq[n]} \hat{f}(S)^3$. Conclude that there exists $S \subseteq [n]$ such that $\Pr_x[f(x) = \chi_S(x)] \geq 1-\epsilon$.

Let $\mathcal{C}$ be a set of Boolean functions $\{-1,1\}^n \to \{-1,1\}$. A *local test* for $\mathcal{C}$ works as follows: Given an unknown function $f : \{-1,1\}^n \to \{-1,1\}$ given as a table of values, a local test makes q queries to f. If $f \in \mathcal{C}$ the test should accept with probability 1, and if f is δ-far from every function in $\mathcal{C}$ then the test should reject with probability $\Omega(\delta)$. One example of a local test that we saw in class (and also above) is the BLR test, which is a 3-query test for the class of *linear functions* $\mathcal{L} = \{\chi_S : S \subseteq [n]\}$. In this problem we will develop a 6-query test for "dictator functions" $\mathcal{D} = \{\chi_{\{i\}} : i \in [n]\}$ - i.e. the set of functions of the form $f(x) = x_i$ for some $i \in [n]$.

a) Let $a, b, c \in \{-1,1\}$ be bits. Give an expression in terms of a, b, c which evaluates to 0 if $a = b = c$, and to 1 otherwise. (This is called the Not All Equal (NAE) predicate.)

b) Consider the following 3-query test (the "NAE" test) on a function f: Pick $x, y, z \in \{-1,1\}^n$ in the following way: Pick (x_i, y_i, z_i) at random from $\{-1,1\}^3 \setminus \{(1,1,1), (-1,-1,-1)\}$, i.e. so that x_i, y_i, and z_i are not all equal. Do this for each coordinate $i \in [n]$ to construct x, y, and z. Then, test that $f(x)$, $f(y)$, and $f(z)$ are not all equal.

Show that

$$\Pr[\text{NAE test accepts}] = \frac{3}{4} - \frac{3}{4}\sum_{S\subseteq[n]} \hat{f}(S)^2(-1/3)^{|S|}$$

(As an aside, note that if f is a dictator, the NAE test accepts with probability 1.)

c) Give a 6-query local test for $\mathcal{D}$ (Hint: Combine the BLR and NAE tests).

> **Note**: The "Long Code" was used by Håstad to prove the inapproximability result for MAX-3LIN referenced in problem 4. This code encodes a string $w \in \{-1,1\}^{\log n}$ with the truth table of the dictator function $\chi_{\{w\}} : \{-1,1\}^n \to \{-1,1\}$, incurring a *doubly exponential blowup*. Håstad heavily uses Fourier analysis to analyze the 3-query test of his PCP. The proof of this result is contained in chapter 22 of Arora-Barak.

[4] For further information on the use of Fourier analysis in complexity theory, visit http://www.cs.cmu.edu/~odonnell/boolean-analysis/. Lectures 2 and 3 will provide sufficient background for solving this problem.

18.405J / 6.841J Advanced Complexity Theory
Spring 2016