CS 1511 Homework 24

Mathew Varughese, Justin Kramer, Zach Smith

Wed, Apr 10

48.

A language L has a PCP verifier if there's a polynomial-time probablistic algorithm V with efficiency, completeness, and soundness.

Call a language L one that has a PCP-verifier using r coing and q adaptive queries.

There is an assumption that the number of queries is at most logarithmic in the input size (n), so 2^q will still be polynomial to n.

So our language L with q queries has a logarithmic amount of queries in the input size(n), so our non-adaptive proof wil be fine with 2^q queries.

49.

Let L be the language of pairs $\langle A, k\rangle$ such that A is a 0/1 matrix and k $\in$ Z satisfying perm(A) = k.

To show L $\in$ PCP(poly(n), poly(n)):

In this case, the verifier expects π to contain the answer to if each pair $\langle A, k\rangle$ fits perm(A) = k.

The verifier picks a b $\in \{0, 1\}$ at random and a random A. They then get a value of k, and see if the it b matches what they find in π for the values of A and k. We can run the following protocol for a polynomial amount of times to get a result where we can construct a π that accepts with probability 1 if perm(A) = k and accepts at most 1/2 of the time if perm(A) != k.

50.

We can run Gaussian Elimination on the matrix formed by the equations. This operation is n^3. Now, taking this system, we create a system of equations that are similar to those in the MAXSAT problem. To rid of the rational coefficients, we can simply multiply by a common factor of all coefficients to make them whole numbers. We take the mod 2 of each coefficient. Then, since the equations are linear, we have a system very similar to MAXSAT. If the equations are not satisfiable in this, they will not be satisfiable be the linear equations. To find this p, we perform a gap reduction just as we did in the MAXSAT problem.