CSE 532 Spring 2008
Computational Complexity II
Problem Set #2
Due: June 10, 2008

Problems:

1. In this problem you will show that for any prime q, any function f computable in $\mathsf{AC}^0[q]$ can be computed by a depth 3 threshold circuit of size $n^{\log^{O(1)} n}$.

   (a) A *probabilistic circuit* is a circuit C that in addition to its regular input x takes as input a vector r of bits. The values of the bits r is chosen uniformly at random. It computes a function f with error at most ϵ if $Pr_r[C(x,r) \neq f(x)] \leq \epsilon$. A special example is to consider probablistic depth-2 $\mathsf{AC}^0[q]$ circuits that consist of a MOD_q gates of AND gates, i.e. multivariate polynomials p over $\mathbb{F}_q$ that on input x and random string r output $p(x,r) \in \{0,1\}$.

   Use the construction given in class to show that for any constant ℓ there is a polynomial p of degree $O(\log n)$ that computes the OR of n-bits with error at most $1/n^\ell$ (and similarly for the AND of n bits).

   (b) Use part (a) and induction on the depth (using the distributive law) to show that for any $\ell' > 0$, any function computed by polynomial-size AC^0_q circuit of depth k can be computed by a probabilistic multivariate polynomial p over $\mathbb{F}_q$ of degree $O(\log^k n)$ and $n^{O(\log^k n)}$ monomials with error at most $1/n^{\ell'}$.

   (c) Now apply the construction which showed that $\mathsf{BPP} \subseteq \mathsf{P/poly}$ to the polynomials in part (b) to compute any $\mathsf{AC}^0[q]$ function of depth k using a circuit consisting of a $MAJORITY$ gate applied to $n^{O(1)}$ polynomials over $\mathbb{F}_q$ of degree $O(\log^k n)$ each having $n^{O(\log^k n)}$ monomials.

   (d) Finally, take the result of part (c) and convert it to a circuit of size $n^{O(\log^k n)}$ consisting of a $MAJORITY$ of $MAJORITY$ gates whose inputs are AND gates of fan-in $O(log^k n)$.

2. We showed two different methods for obtaining lower bounds on deterministic communication complexity. One was via fooling sets: We showed that $D^{cc}(f) \geq \log_2 |A|$ where $A = \{(x_1,y_1),\ldots,(x_m,y_m)\}$ is a set of input pairs such that $f(x_j,y_j) = 1$ but for any $i \neq j$ at least one of $f(x_i,y_j)$ or $f(x_j,y_i)$ is 0. We also showed that $D^{cc}(f) \geq \log_2 rank(M_f)$. Show that for any fooling set A, $|A| \leq rank(M_f)^2$ and therefore the rank lower bound is always at least half the fooling set lower bound. To do this define a new matrix M^* which is the outer product of $M_f \otimes M_f^T$ and look at the submatrix of M^* whose rows and columns are indexed by elements of A. (The matrix $M \otimes N$ is the matrix with $rows(M) \cdot rows(N)$ rows and $cols(M) \cdot cols(N)$ columns that replaces each entry m_{ij} of M with the matrix $m_{ij}N$. You will need the fact that $rank(M \otimes N) = rank(M) \cdot rank(N)$.)

3. A Boolean formula F is *read-once* if and only if each variable labels at most one leaf of F. Suppose that function $f : \{0,1\}^n \to \{0,1\}$ is computed by a read-once formula. Define $g(x,y) = f(x_1 \oplus y_1, x_2 \oplus y_2, \ldots, x_n \oplus y_n)$. Use induction and the rank lower bound to prove that $D^{cc}(g) \geq n$. Hint: Use the rank property of $\otimes$ as above and the fact that the all 1 J-matrix has rank 1.

4. We can define a distribution on restrictions $R_{p,n}$ in which each bit is unset with probability p and each bit that is set is chosen independently and uniformly at random. We can apply restrictions to formulas simplyfing them by propagating the values.

   (a) Show that given a De Morgan formula F with s leaves the expected number of leaves of $F|_\rho$ for ρ chosen from $R_{p,n}$ is at most a constant times $p^{3/2}s + 1$. (One might expect ps but one can do better.)

   (b) Use part (a) to show that $Parity_n$ requires formula size $\Omega(n^{3/2})$.

   (c) Define the function $g : \{0,1\}^{n+\log_2 n} \to \{0,1\}$ to be $g_{x_1 \ldots x_n}(x_{n+1}, \ldots, x_{n+\log_2 n})$ where $g_{x_1 \ldots x_n} : \{0,1\}^{\log_2 n} \to \{0,1\}$ is the function whose truth table has x_i as its i-th entry. Use Shannon's Theorem for formulas to derive that for almost all choices of $x_1, \ldots x_n$ we have that $L(g_{x_1,\ldots,x_n})$ is $\Omega(n)$.

   (d) Using g we can define an explicit function f on $(1+\log_2 n)n$ bits denoted as $x_{i,j}$ for $0 \leq i \leq \log_2 n$ and $1 \leq j \leq n$ where

   $$f(x) = g_{x_{0,1} \ldots x_{0,n}}(\oplus_{j=1}^n x_{1,j}, \cdots, \oplus_{j=1}^n x_{\log_2 n, j}).$$

   Now, using ideas similar to part (b) above, apply part (a) to derive that $L(f)$ is $\Omega(n^{5/2-\epsilon})$ for any $\epsilon > 0$.