Complexity of Boolean and Arithmetic Circuits - 236603

Exercise 3

Lecturer: Amir Shpilka *Hand in date: July 6th*

Please write your solution in LaTeX and submit them by email to shpilka@cs.technion.ac.il.

1. For a function $f : \{0,1\}^n \to \{0,1\}$ the *k-th slice* is the function

$$f^{(k)}(x) = \begin{cases} 0 & \sum_{i=1}^n x_i < k \\ 1 & \sum_{i=1}^n x_i > k \\ f(x) & \sum_{i=1}^n x_i = k \end{cases}$$

   Verify that each $f^{(k)}$ is a monotone function. Prove that for every f and $1 \le k \le n$ we have that

$$S^M(f^{(k)}) = O(S(f) + n^2 \log(n)).$$

   In other words, there is almost no gap between the monotone complexity and the non-monotone complexity of slice functions.

2. Let $maj(x_1, \ldots, x_n)$ be the majority function.
   (a) Prove that any boolean circuit of depth d, where d is a constant, for maj requires size $2^{n^{\Omega(1/d)}}$.
   (b) Prove that for every constant d there exists a depth d boolean circuit of size $2^{\Omega(n^{1/(d-1)})}$ computing $\bigoplus_n$ (the parity on n variables).
   (c) Prove that there exists a depth 2 circuit of polynomial size over the basis $\{\wedge, \vee, \neg, maj\}$ that computes $\bigoplus_n$.

3. (a) Show that there exist a circuit over the basis $\{\wedge, \vee, \neg\}$ with only $O(\log(n)$ negation gates (and any number of $\{\wedge, \vee\}$ gates) that on input $x_1, \ldots, x_n$ computes $\neg x_1, \ldots, \neg x_n$. (the circuit should of course be defined over the positive literals only). Hint: compute the number of 1's in the input.
   (b) Prove that any such circuit must use $\Omega(\log(n))$ negation gates.