KTH Numerical Analysis
and Computer Science

Homework

Due on Jan 20 at 10.15. Many of the problems are of such nature that the solutions can be found in the literature. Please solve the problems by yourself unless specifically asked to proceed in a different manner. Problems can be discussed in groups of up to three people but solutions should be written and handed in on an individual basis. It goes without saying that everybody handing in a solution should understand it fully. Also please specify with whom you have been collaborating.

1. We discussed Zwick's lowerbound $4n - O(1)$ for circuits with fanin-two gates but not allowing parity or equivalence gates. It applied to many symmetric functions but not to parity. Prove the exact bound on the size of circuits computing parity in this model. You are supposed to prove lower and upper bounds that match exactly.

2. Find an oracle A such that $\mathrm{NP}^A = P^A$ and a different oracle B such that $\mathrm{NP}^B \neq P^B$. Do not consult the literature on this problem, but instead consider the following hints. To construct A we need to make sure that ordinary NP is contained in P^A and thus one needs A to be at least NP-hard and the first thought might be to make A NP-complete. This does not quite work. To construct B we just have to make sure that there is a language L in NP^B such that none of a countable number of polynomial time oracle Turing machines M_i^B recognize L. For each i we only need to find one input such that M_i^B makes a mistake an hence it is sufficient to fix a finite part of B to make it go wrong.

3. Prove that there are functions that can be computed by a large weight threshold gate but which, assuming integral weights, require weights of exponential size. What upper bound can you give for the weights required for any such function?

4. Prove that the following two given definitions of Σ_k are equivalent.

1. $\Sigma^0 = P$, $\Sigma^{k+1} = \mathrm{NP}^{\Sigma^k}$.
2. The set of languages L that can be defined as $x \in L$ iff $\exists y_1 \forall y_2 \ldots Q y_k R(x, y_1, y_2, \ldots y_k)$ for some relation R which can be decided in polynomial time and where $|y_i| \leq |x|^c$ for some fixed constant c. Here Q is a quantifier that is $\exists$ if k is odd and $\forall$ if k is even.

5. In a permutation branching program of length l and width w we have l levels each of w nodes. For each level i we have two permutations π_i and σ_i and an associated literal l_i. If l_i is true for a certain input then π_i is chosen for that level and otherwise σ_i is chosen. The result of the program is the composition of the chosen permutations. If this permutation is the identity the input is accepted and otherwise rejected.

Consider permutation branching programs of widths 2 and 3. As a warmup characterize exactly the functions computable by permutation branching programs of width 2. Then show that languages recognizable by polynomial size width-3 permutation branching programs can be recognized by polynomial size constant-depth circuits containing certain types of modular gates. Can any Boolean function be computed by a width-3 branching program?

6. We know that majority can be computed by depth $O(\log n)$ circuits. Design a Karchmer-Wigderson game that solves the corresponding game with $O(\log n)$ bit communication.

7. Suppose we are interested in whether we have a path of length $\log n$ in a graph between s and t in a graph G. What is the monotone complexity of this problem? Give upper and lower bounds that are within a constant of each other!

8. Let C be a depth-d circuit of size 2^s. The maximal fraction of inputs for which C agrees with parity can be written on the form $\frac{1}{2} + P(d, s, n)$. Find a good upper bound on P and try to give examples of circuits getting as close as possible to this bound.

9. Let F be a finite field and consider all functions $f : \{0,1\}^n \mapsto F$. Show that any such function f can be represented as a polynomial in which each monomial is multilinear. Show that this representation is unique.

10. We showed that if p is a prime then each function f computed by a polynomial size, depth-d circuit containing $\wedge$, $\vee$ $\neg$ and Mod_p gates can be well approximated by a low degree polynomial modulo p. To be more exact there is a polynomial of degree $O((\log n)^d)$ that agrees with f on a fraction $1 - \frac{1}{n}$ of all points. Is the same true for $p = 6$?

11. Construct an NC^1 circuit that, given an n bit number in binary notation in the closed interval $[0, 1]$ computes $\sin x$ with an accuracy of 2^{-n}. The answer might contain more than n bits.

12. Consider the predicate $x \stackrel{?}{>} y$ for two n bit numbers x and y. Show that this can be computed by a polynomial size depth-2 circuit of majority gates. It might be easier to make the circuit have a majority gate at the output and xor-gates next to the inputs. This is most easily done by a probabilistic construction. For each i make a set of gates that has the property that if the ith bit of x and y are equal then the net contribution of these gates is 0 while if they are unequal there is a large contribution if this is the most significant bits in which the numbers differ while the contribution is small if there are more significant bits in which they differ.

You need to prove that such a constructed circuit can be converted into a depth-2 majority circuit, and you may wish to prove this general fact even if you are unable to complete the construction in this particular case.

Also, as a warm-up (or for partial credit) consider the same question for the predicate $x \stackrel{?}{=} y$.