CS642
(Circuit Complexity Theory)
End Semester Examination
Maximum Marks : 60

Starting date: 23 Apr 2013
Submission Date: 30 Apr 2013

NOTE: In the classes below, uniformity conditions are not specified. You may assume whatever suits the context.

We have concentrated on *boolean* circuits in this course. That is, circuits having gates whose input/output lines carry only 0-1 values. An interesting variation of this is *arithmetic* circuits. This is, circuits having addition, subtraction, multiplication and division gates where these arithmetic operations are over a specified ring or field (division gates make sense only when the operations are over a field). Notice that subtraction can be achieved by multiplication and addition gates provided we have the constant -1 available. Also, division can be achieved by addition and multiplication gates if the operations are over a finite field (a division gate is always of fanin-2).

The question naturally arises as to what is the power of arithmetic circuits as compared to their boolean counterparts. Since arithmetic circuits in general compute functions, we shall compare them with functions computed by their boolean counterparts.

You have already shown in the midsem that polynomial-size, constant-depth, unbounded fanin arithmetic circuits where the operations are over a fixed finite field capture precisely the class ACC. Boolean counterparts of such circuits are AC^0 circuits. Therefore, the arithmetic circuits in constant-depth, unbounded fanin settings are strictly more powerful than their boolean counterparts.

If, instead of choosing the underlying field to be a fixed finite field, we choose the field of rationals, then we get another class of circuits. What is the power of this class?

Show that this class of circuits precisely captures the class of functions in TC^0. (10 marks)

Now, let us turn our attention to the counterpart of the class NC^1. Define the class $\#\mathrm{NC}^1$ to be the class of functions computed by families of polynomial-size, logarithmic depth, fanin-2 arithmetic circuits. Let us investigate the power of this class. For the sake of simplicity, *we shall assume that the circuits in $\#NC^1$ contain only addition and*

multiplication gates over integers. We simulate subtraction by supplying constant -1 to the circuits.

First, show that all functions in NC^1 are also in $\#\mathrm{NC}^1$. (5 marks)

Somewhat surprisingly, the class $\#\mathrm{NC}^1$ almost equals the class NC^1. Almost, because it can only be shown that any function in the class $\#\mathrm{NC}^1$ can be computed by a family of polynomial-size, fanin-2 boolean circuits of depth $O((\log n) * (\log^* n))$. (Here, $\log^* n$ is the number of times one needs to take logarithm of n to reach 1. For example, $\log^* 2 = 1$ since one application of logarithm on 2 yields 1. $\log^* 4 = 2$ since $\log \log 4 = 1$. $\log^* 16 = 3$, $\log^* 65536 = 4$, and $\log^*(2^{65536}) = 5$. Notice that 2^{65536} is larger than the number of atoms in the universe, and therefore, for all practical purposes, $\log^* n \leq 5$!!).

The proof of this is in two parts. We only show the first part of this: given any $\#\mathrm{NC}^1$ circuit, one can construct a sequence of polynomially many 3×3 matrices such that

- the entries of each matrix is either constants available to the circuit (i.e., 0, 1, or -1) or an input variable,
- letting P be the product of all these matrices,

$$P = \begin{bmatrix} 1 & X & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

  where X is equal to the output of the $\#\mathrm{NC}^1$ circuit.

Prove the above statement (the proof of this has the same overall structure as the Barrington's proof—only now you need to work with 3×3 matrices instead of 5-cycles). (15 marks)

As an aside, show that the problem of computing product of 3×3 matrices over integers is AC^0-complete for $\#\mathrm{NC}^1$. (5 marks)

We now turn our attention to investigating the structure of AC^0-complete sets for NC^1. The same proof will carry over to other classes. Let A be an AC^0-complete set for NC^1 and $B \in \mathrm{NC}^1$. We aim to show that there exists an NC^0 reduction of B to A. Define set $\hat{B}$ as accepted by the following algorithm:

> On input x, let $x = 1^d 0 1^k 0 y$. Reject if k does not divide $|y|$. Otherwise, let $y = u_1 u_2 \cdots u_q$ with $|u_i| = k$ for every i. Let $v_i = 1$ if the number of ones in u_i equals zero modulo d, otherwise $v_i = 0$. Let $v = v_1 v_2 \cdots v_q$. Accept iff $v \in B$.

Show that $\hat{B} \in \mathrm{NC}^1$. (5 marks)

Let the AC^0-circuit family $\{C_n\}_{n \geq 1}$ compute the reduction f of $\hat{B}$ to A. Define a reduction h of B to $\hat{B}$ so that $f \circ h$ can be computed by an NC^0-circuit family. (20 marks)

Hint: see http://www.cse.iitk.ac.in/users/manindra/isomorphism/non-uniform-ac0-iso.pdf for ideas on how to prove it. The definition of $\hat{B}$ above is different from the one in the paper. This is deliberate! You must use this definition.