

### Problem 1

$ITERADD_{poly(\log n),n} \in AC^0$  where  $ITERADD_{poly(\log n),n}$  is problem of adding polylog many  $n$ -bit numbers.

**Solution:** We already know  $ITERADD_{\log n,n} \in AC^0$ . So we will use this in order to show

$$ITERADD_{poly(\log),n} \in AC^0$$

Number of bits after adding  $k$  many  $n$ -bit numbers is  $\log k + n$  bits. We will solve it with a tree like circuit with depth  $k$ .

Suppose we are given  $\log^k n$  many  $n$ -bit numbers. Let the numbers are  $a_1, \dots, a_{\log^k n}$ . Now we group the numbers where each group contains  $\log n$  many numbers. So there will be total  $\log^{k-1} n$  many groups. In each group we will add the  $\log n$  many  $n$ -bit numbers which we know we can do in  $AC^0$ . Now from each group we get numbers with  $\log \log n + n = O(n)$  bits.

Now from these  $\log^{k-1} n$  many  $O(n)$  bits numbers we again form groups where each group contains  $\log n$  many numbers. So we will have  $\log^{k-2} n$  many groups. Again in each group we add the  $\log n$  many  $n$  bit numbers which we can do in  $AC^0$ . The numbers we get from each group has  $\log \log n + O(n) = O(n)$  bits.

We continue this and notice each time the power of the log is reducing by 1. So we have to do this process  $k$  times. And then we will have 1 number which is the addition of all the  $\log^k n$  many  $n$ -bit numbers. In  $i$ -th step we use the  $AC^0$  circuit for adding  $\log n$  many  $n$ -bit numbers and get  $\log^{i-1} n$  numbers where each number has  $\log \log n + O(n) = O(n)$  many bits.

Since each step is using a  $AC^0$  circuit parallelly  $\log^{i-1} n$  many times and there are total  $k$  many such iterations. So the total circuit will have constant depth and polynomial size. Hence  $ITERADD_{poly(\log n),n} \in AC^0$

□

### Problem 2

**Defintiion 1:** It is the class of constant depth polynomial size circuits with AND, OR, NOT, MAJORITY gates.

**Defintiion 2:** It is the class of constant depth polynomial size circuits with NOT, MAJORITY gates.

Equivalence of The two Definitions of  $TC^0$

**Solution:** If we can construct the AND and OR gate with MAJORITY gate in constant depth and polynomial size then we can say these two definitions are equivalent.

Now if  $a_0, \dots, a_{n-1}$  are  $n$  bits. Then

$$\bigwedge_{i=0}^{n-1} a_i = MAJORITY\left(a_0, \dots, a_{n-1}, \underbrace{0, 0, \dots, 0}_{n-1 \text{ times}}\right)$$

Similarly

$$\bigvee_{i=0}^{n-1} a_i = MAJORITY\left(a_0, \dots, a_{n-1}, \underbrace{1, 1, \dots, 1}_{n-1 \text{ times}}\right)$$

Now the AND and OR gate both can be replaced with  $O(1)$  depth  $O(n)$  size subcircuits using only MAJORITY gates in a circuit and still the total depth of the circuit will be constant and size is polynomial. Hence the two definitions are equivalent.

□