Today

- Barrington's theorem and Ben-Or+Cleve Proof.
- Circuit complexity lower bounds: Constant depth circuits.
- Hastad's switching lemma and the Segerlind+Buss+Impagliazzo proof.

Recall Branching Programs and Width

- BP = DAG, with one source, two sinks labelled 0/1, non-sink vertices labelled with variables, and having out-degree 2.
- Layered BP: all edges go from layer i to layer $i+1$.
- Width of (Layered) BP: max size of a layer.
- Pre-80's conjecture: O(1)-width branching programs of poly-size can not compute majority of n bits.
- Barrington's theorem: O(1)-width branching program compute any depth d formula in size $2^{O(d)}$.

- Today will give arithmetic proof.

Register machines and Straightline computation

- Arithmetic circuit: Inputs from field, gates compute addition/multiplication.
- Register machines: Limited memory version of arithmetic circuit. general operation $R_i < -X \circ (Y \cdot Z)$, where $\circ, \cdot$ are field operations; and X, Y, Z are one of field constants, input variables, or other registers.
- Register machine computes $f(x_1, \ldots, x_n)$ if starting with all registers set to zero, some register finally contains $f(x_1, \ldots, x_n)$.
- Register machine with c registers computes

f over $\mathbb{F}_2$, then f can be computed with width 2^c.

Ben-Or+Cleve Theorem

- If f can be computed by depth d arithmetic formula, then f can be computed by 3-register machine with length $2^{O(d)}$.
- Implies width eight braching program for all poly-sized formulae.

Proof

- Inductive claim: If f has depth d, and R_1, R_2, R_3 are arbitrarily initialized, then can leave register machine in state $(R_1, R_2, R_3 + f(x_1, \ldots, x_n) \cdot R_2$ in length $2^{2 \cdot d}$.
- Base Case trivial.
- Induction: If $f = f_1 + f_2$, then draw picture.
- Induction: If $f = f_1 * f_2$, then draw complex picture.
- Verify lengths.

Summary on branching programs

- Major open questions:
  - Are $O(1)$-width poly-sized branching programs equal to unbounded width poly-sized branching programs?
  - Give "explicit" function with super-poly branching program size.
- Till date no success except by limiting width/length.
- Till recently, no technique for even super-linear depth. Recent progress: There exists an explicit function that takes super-linear depth, if size is $2^{\epsilon n}$. [Ajtai].

- Won't cover this result; but will encounter a uniform version with simple proof.

Circuit complexity

- Most general model of non-uniform computing.
- Not suprisingly little known for unrestricted case.
- Restricted cases:
  - Monotone circuits: Exponential lower bounds known.
  - Bounded depth, unbounded fan-in OR/AND (known as AC^0): exponential lower bounds known.
- In these lectures:
  - Combinatorial proof of AC^0 lower bound.

  - Algebraic proof of AC^0 lower bound.
  - Connections between circuit depth and communication complexity.

AC^0 lower bounds for parity

Defn: $\bigoplus(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i (\mod 2)$.

Theorem: Parity of n bits requires exponential size for $O(1)$ depth circuits.

- First super polynomial bounds established by Furst, Saxe and Sipser and independently by Ajtai.
- Exponential bounds given later by Yao.
- Hastad gave a clean exponential lower bound, highlighting the role of the switching lemma.
- Razborov-Smolensky later gave an algebraic proof.

Switching lemma

k-DNF is an OR of terms, where each term is AND of at most k literals.

k-CNF is an AND of clauses, where each clause is the OR of at most k literals.

Random restriction with parameter p: Set each variable to $0/1$ w.p. $(1-p)/2$ each and leaves it unset with probability p. Does this independently for each variable.

Hastad's Switching lemma: Random restriction of k-CNF with parameter $p \leq 1/7$ yields a ℓ-DNF with probability at least $1-(7pk)^{\ell}$.