Today

- Parity does not have constant depth AND-OR-NOT circuits.
- Few words on Non-uniform models of computing.

Circuit complexity: Big goal

- Would like to show exponential lower bounds on circuit size for functions in NP.
- Best we've been able to show is exponential lower bounds on constant depth circuits.
- References:
  - Furst, Saxe, Sipser '83.
  - Yao '85.
  - Hastad '87.
  - Smolensky '88.
- Today: Smolensky's proof.

Circuit depth

- Depth of a circuit is the length of the longest path from input to output.
- Today we consider AC_0: the class of circuits with unbounded fan-in OR, and AND gates, and constant depth.
- Depth represents parallel time. Unbounded fan-in represents concurrent writing on shared memory cells.
- "Lowest level of complexity".

Parity function

For every n, $\bigoplus_n : \{0,1\}^n \to \{0,1\}$ represents the parity of n bits (or sum modulo two).

Goal for today:

Theorem: If $\bigoplus_n$ has a circuit of depth d then it must have size $2^{n^{\Omega(1/d)}}$.

Main tools

- Vector spaces over $\mathbb{Z}_3^n$.
- Polynomials over $\mathbb{Z}_3^n$.
- Randomness.

Parity and polynomials

- $\mathbb{Z}_3 = \{-1, 0, +1\}$ (Arithmetic mod 3, but think of 2 as -1.)
- Two representations of the Boolean world: $\{0,1\}$ and $\{+1,-1\}$. $(0 \leftrightarrow 1;\ 1 \leftrightarrow -1.)$
- $x \mapsto 1 - 2x$ and $(1-y)/2 \leftarrow y$.
- Then $\bigoplus_n : \langle x_1, \ldots, x_n \rangle \mapsto \prod_{i=1}^n x_i$.
- In general think of $f : \{0,1\}^n \to \{0,1\}$ and $f : \{+1,-1\}^n \to \{+1,-1\}$ as functions mapping $\mathbb{Z}_3^n \to \mathbb{Z}_3$.

Polynomials over $\mathbb{Z}_3$

Fact: For every $f : \{0,1\}^n \to \{0,1\}$, can find polynomial $q : \mathbb{Z}_3^n \to \mathbb{Z}_3$ such that q has degree 1 in each variable and agrees with f on $\{0,1\}^n$.

Similar fact for $f : \{-1,1\}^n \to \{-1,1\}$.

Main Lemmas

Lemma 1: If $f : \{0,1\}^n \to \{0,1\}$ is computed by a depth d circuit of size s, then there exists a set $S \subseteq \{0,1\}^n$ of size $|S| \geq 3/4 2^n$ such that $f : S \to \{0,1\}$ computed by a polynomial over $\mathbb{Z}_3$ of degree $(\log s)^{O(d)}$.

Lemma 2: If there exists a degree polynomial D $p : \mathbb{Z}_3^n \to \mathbb{Z}_3$ such that $p(x) = \bigoplus(x)$ for all $x \in S$, then every Boolean function $f : S \to \{0,1\}$ is computed by polynomials of degree $n/2 + D$.

Lemma 3: Any set of functions generating all $f : S \to \{0,1\}$ must have at least $|S|$ members.

Using lemmas to prove theorem

- Assume parity has depth d, size s circuit.
- By Lemma 1, parity is computed by polynomial of degree $(\log s)^{O(d)}$ on set S of size $3/42^n$.
- By Lemma 2, every Boolean function on S is a polynomial of degree $n/2+(\log s)^{O(d)}$. Thus this set of functions is contained in a vector space over $\mathbb{Z}_3$ of dimension at most $\sum_{i=0}^{n/2+(\log s)^{O(d)}} \binom{n}{i} \leq 2^{n-1} + (\log s)^{O(d)} 2^n/\sqrt{n} < 3/42^n$. (Provided $s \leq 2^{n^{\Omega(1/d)}}$.)
- By Lemma 3, this space of functions has dimension at *least* $|S| \geq 3/42^n$.

- We have a contradiction

Proof of Lemma 3

- Let $\delta_x(y) = 1$ if $x = y$ and 0 o.w..
- The functions $\{\delta_x : S \rightarrow \{0,1\} | x \in S\}$, are linearly independent.
- Simple linear algebra.

Proof of Lemma 2

- Will switch back and forth between 0/1 and ± 1.
- Suppose $\bigoplus : S \rightarrow \{0,1\}$ is represented by a polynomial $q : \mathbb{R}^n \rightarrow \mathbb{R}$. Let $T \subseteq \{+1,-1\}^n$ be the associated set. Then $\prod_{i=1}^n x_i = 1 - 2q((1-x_1)/2, \ldots, (1-x_n)/2)$ on the set T.
- Consider Boolean function $f : S \rightarrow \{0,1\}$. Let $g : T \rightarrow \{+1,-1\}$ be associated function. Represent g by a polynomial in its arguments. $p(\mathbf{x}) = \sum_i \alpha_i A_i + \sum_j \beta_j B_j$ where A_i are terms of degree less than $n/2$ and B_j's are terms of degree greater than $n/2$. Let $C_j = \prod_{i=1}^n x_i / B_j$. Then $p'(\mathbf{x}) =$

$\sum_i \alpha_i A_i + q(\mathbf{x}) \sum_j \beta_j C_j$ also represents g and is a polynomial of degree at most $n/2 + D$.

- The polynomial $r(\mathbf{x}) = (1 + p(1 - 2\mathbf{x}))/2$ represents f.

Proof of Lemma 1

- This is the hard lemma. (Though the linear algebra is also very novel.)
- But is seen again and again in complexity.
- Basic idea: Fix input $x_1, \ldots, x_n$ and randomly replace every gate by a polynomial of low-degree. Show the resulting circuit still computes the original value with probability at least $3/4$.
- Use the probabilistic method to conclude there exists a collection of polynomials which computes the original function on $3/4$ths of the input.

Prob. polynomial for the OR function

Naive answer: $OR(y_1, \ldots, y_k) = 1 - \prod_{i=1}^{k}(1 - y_i)$. Answer is always right. But degree is k - too much.

Step 1: Get the answer right w.p. $1/2$ with polynomials of degree 2.

Basic idea: pick $a_1, \ldots, a_k \in \mathbb{Z}_3$ at random. $p_{\mathbf{a}}(\mathbf{y}) = \sum_{i=1} a_i y_i$.

Claim 1: $p_{\mathbf{a}}(\mathbf{0}) = 0$.

Claim 2: $\Pr_{\mathbf{a}}[p_{\mathbf{a}}(\mathbf{y}) = 0] \leq 1/3$.

Proof: Let $Q(\mathbf{z}) = \sum_{i=1}^{k} y_i z_i$. Q is a non-zero polynomial of degree 1 in its argument. Evaluation at random $\mathbf{z} = \mathbf{a}$ leaves it non-zero.

Prob. polynomial for the OR function (contd.)

The polynomial $p_{\mathbf{a}}^2$ is always 0 or 1 and computes the OR function on any fixed input w.p. $2/3$.

Pick $\mathbf{a}_1, \ldots, \mathbf{a}_l$, and take the OR of polynomials $p_{\mathbf{a}_i}$.

Gives degree 2ℓ polynomial that is right w.p. $1 - (2/3)^{\ell}$.

What we gained? Will pick $\ell = \log s$ to make degrees logarithmically smaller than fan-in.

What we lost? Not guaranteed to be right.

Prob. polynomial for circuit

- Replace every gate by degree 2ℓ poly randomly.
- Resulting circuit computes a polynomial of degree $(2\ell)^d$.
- Prob. it gets the output wrong (for fixed input) is at most $s(1/3)^\ell$.
- Lemma follows.

Conclusions

- Algebra, arithmetization, randomness very powerful tools.
- Work in situations where there's no mention of them in problem statement.
- Many more examples in course.
- Unfortunately, know little else?

Non-Uniform Models

- Many in existence
  - Circuits
  - Formulae
  - Branching Programs
  - "Rectifier and Switching Networks".
- Circuits corr. to Time
- Branching program width corr. to Space.