E0 309: Topics in Complexity Theory **Spring 2015**

Lecture 10: Feb 25, 2015

Lecturer: Neeraj Kayal *Scribe: Sumant Hegde*

10.1 Determinant is irreducible

Claim 10.1 *Let $X = \{x_{ij}\}_{i,j\in[n]}$ and $f(X), g(X)$ be two polynomials such that $f(X) \cdot g(X) = DET(X)$. Let $var(f)$ denote the set of variables appearing in the polynomial f. Then $var(f)$ and $var(g)$ are disjoint.*

Proof: Suppose for contradiction that $y \in X$ is in both $var(f)$ and $var(g)$. Then we can view f and g as polynomials in y as shown below.

$$f = \alpha(Z) \cdot y^{d_f} + \text{lower order terms w.r.t. } y$$
$$g = \beta(Z) \cdot y^{d_g} + \text{lower order terms w.r.t. } y$$

where $Z = X \setminus \{y\}$ and α, β are polynomials not equal to 0, and $d_f, d_g > 0$. Clearly

$$f \cdot g = \alpha(Z)\beta(Z) \cdot y^{d_f+d_g} + \text{lower order terms w.r.t. } y$$

Since $\alpha(Z), \beta(Z) \not\equiv 0$, $f \cdot g$ has a monomial in which degree of y is $d_f + d_g > 1$. However degree of y is equal to 1 in $DET(X)$ as $DET(X)$ is a multilinear polynomial, leading to a contradiction. ∎

Remarks
1. The above claim easily extends for more than two polynomials. The polynomials will be pairwise variable-disjoint.
2. It follows that on expanding $f \cdot g$ into sum-of-products form no cancellations occur.

Claim 10.2 *Let $X = \{x_{ij}\}_{i,j\in[n]}$, $n \geq 2$. Then $DET(X)$ cannot be expressed as a product of linear polynomials.*

Proof: Assume that there do exist linear polynomials $l_1(X), \ldots, l_m(X)$ such that $DET_n = \prod_{i\in[m]} l_i$.

Consider generating the monomial $x_{11}x_{22}\ldots x_{nn}$ present in $DET(X)$. From the variable-disjointness property mentioned above, each x_{ii} should be present in a distinct linear form. Say w.l.o.g. that x_{ii} is present only in l_i.

Consider generating another monomial, say $x_{12}x_{21}x_{33}\ldots x_{nn}$ present in $DET(X)$. If $x_{12} \in var(l_1)$ then on multiplication we get $x_{12}x_{22}x_{33}\ldots x_{nn}$ as one of the monomials which is invalid in the determinant. Cancellations cannot occur, so this invalid monomial remains in the final product. On the other hand, if $x_{12} \in var(l_i)$ for some $i \geq 2$, then on multiplication we get $x_{11}\ldots x_{12}\ldots$ as one of the monomials which again is invalid. Thus the product always has some invalid monomials, a contradiction. ∎

Claim 10.3 *$DET(X)$ cannot be expressed as a product $f(X) \cdot g(X)$ where $deg(f(X)), deg(g(X)) \neq 0$.*

Proof: *(sketch)* The variable-disjointness property holds for $f(X), g(X)$ as well. The proof is along the lines of that of claim 10.2. ■

Claim 10.4 *$DET(X)$ cannot be expressed in the form $l_1(X)P_1(X)+l_2(X)P_2(X)$ where $deg(l_1(X)), deg(l_2(X)) = 1$ and $deg(P_1(X)), deg(P_2(X)) > 0$.*

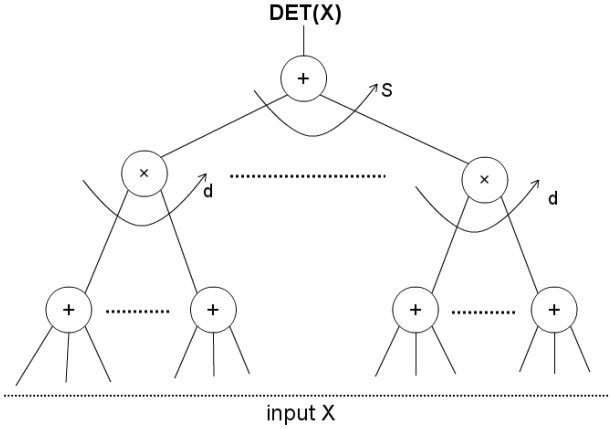
Proof of this claim is left as an exercise. We are also asked to verify if it is impossible to express $DET(X)$ in the form $l_1(X)P_1(X) + l_2(X)P_2(X)$ where $deg(l_1(X)), deg(l_2(X)), deg(P_1(X)), deg(P_2(X)) > 0$.

10.2 Lower bounds for depth three arithmetic circuits

The determinant when expressed in the form

$$DET_n = \sum_{i=1}^{s} \prod_{j=1}^{d} l_{ij} \qquad \deg(l_{ij}) = 1 \ \forall \ i, j$$

corresponds to a depth three arithmetic circuit, as shown:



Naturally, we call it a $\Sigma\Pi\Sigma$ circuit. We could have a $\Pi\Sigma\Pi$ circuit to compute the determinant. In that case, however, since the determinant is irreducible, there would be at most two input edges to the product gate at the top. The first input would be a linear combination of products of variables, say $\sum_{i=1}^{s} \alpha_i \prod_{j=1}^{n} x_j^{j_i}$ and the second input would be a constant, say β. This β can be "pushed down" to the subcircuit rooted at the first input, to make it $\sum_{i=1}^{s} \alpha_i\beta \prod_{j=1}^{n} x_j^{j_i}$. This makes the product gate at the top irrelevant, reducing the circuit depth to two. Most of the polynomials we consider will be irreducible and therefore we will focus only on $\Sigma\Pi\Sigma$ circuits in the rest of the lecture.

Admittedly, our knowledge on the lower bounds for depth three circuits is very limited. For instance, consider the lower bound on the size of a depth three circuit computing the determinant.

Conjecture 10.5 *The size of any $\Sigma\Pi\Sigma$ circuit computing DET_n must be super-polynomial, i.e. $n^{\omega(1)}$.*

The best known lower bound on the size of any $\Sigma\Pi\Sigma$ circuit computing DET_n, in terms of the number of edges, is $\Omega(n^4)$ [1].

A recent result shows that if we can prove a "strong enough" lower bound on the size of any $\Sigma\Pi\Sigma$ circuit computing the $n \times n$ permanent $PERM_n$ then we get super polynomial lower bound for general arithmetic circuits. Thus it seems that proving lower bounds for depth three circuits is as nontrivial as for general circuits.

10.3 What lower bounds can we hope for?

Before proceeding further we try to answer a basic question: What lower bounds can we hope for? There are two queries implicit in the question:
1. showing the existence of functions that are "hard" to compute, and
2. showing an explicit function and proving that it is hard to compute.
We answer the queries first in the boolean world and then the arithmetic world. As we will see we arrive at the same conclusion in both.

10.3.1 In the Boolean World

There exist boolean functions that are hard to compute. Moreover, most boolean functions are hard to compute. This can be shown by a counting argument, as follows.

Consider the set of boolean functions on n inputs

$$f_n = \{f(x_1, \ldots, x_n) : \text{each } x_i \in \{0, 1\}\}.$$

Total number of such functions $= |f_n| = 2^{2^n}$.

Now we want to upper-bound the total number of functions computable by circuits of size s. We assume that the $\wedge$ and $\vee$ gates have fanin two.

Lemma 10.6 *Total number of boolean circuits of size s is less or equal to 2^{s^2}.*

Proof: For every boolean circuit ϕ of size s we define a straight line program (SLP) as follows. The SLP has s lines. Any line l_i is of the form

$$l_i = x_i \text{ for } i \in [n] \text{ (i.e. the first } n \text{ lines are inputs.)}$$
$$l_i = l_j \wedge l_k \text{ or } l_i = l_j \vee l_k \text{ or } l_i = \neg l_j \text{ for } i \geq n + 1$$

where $j, k < i \leq s$. There is a one-to-one mapping between the lines of the SLP and the nodes in ϕ. This equivalence between boolean circuits and straight line programs (SLPs) implies that it suffices to count the number of SLPs of s lines.

Clearly the right hand side (RHS) of l_i has at most $3s^2$ choices. There are s such lines.
So the number of SLPs of size s is less or equal to $(3s^2)^s = 3^s s^{2s} \leq 2^{s^2}$ ■

It follows that the fraction of boolean functions computed by circuits of size s is less or equal to $2^{s^2}/2^{2^n}$. Let us say $s = 2^{n/3}$. Still, the fraction of functions computed by the circuits of size $s = 2^{2n/3}/2^{2^n} << 1$. That is, with high probability, even large circuits cannot compute a random boolean function (on n inputs).

Thus we have showed the existence of functions hard to compute. The other task, that is, showing an explicit function and proving it is hard to compute, has remained unaccomplished by the research community. Nevertheless, plenty of functions, including all NP-complete problems, have been considered to be candidates.

10.3.2 In the Arithmetic World

There exist polynomials that are hard to compute. Moreover, most n-variate degree d polynomials are hard to compute. This can be verified by a counting argument, as follows. Consider the set f_n of n-variate degree d polynomials over a finite field $\mathbb{F}$.

Total number of such polynomials $= |f_n| = \mathbb{F}^{\binom{n+d}{d}}$.

We want to upper-bound the total number polynomials computable by arithmetic circuits of size s. We assume that the fanin of sum gates is unbounded and the fanin of product gates is two.

Lemma 10.7 *The total number of arithmetic circuits of size s is less or equal to $\mathbb{F}^{s^2}$.*

Proof: It suffices to upper bound the total number of circuits with s product gates (and any number of sum gates). Accordingly, for any circuit ϕ with n inputs and s product gates we define a straight line program with $n + s$ lines, as follows.

$$\begin{aligned} l_i =& x_i && \text{for } i \in [n] \\ l_i =& (\alpha_{i,0} + \alpha_{i,1}l_1 + \cdots + \alpha_{i,i-1}l_{i-1}) \cdot (\beta_{i,0} + \beta_{i,1}l_1 + \cdots + \beta_{i,i-1}l_{i-1}) && \text{for } n+1 \le i \le n+s \end{aligned}$$

where $\alpha_{i,j}$ and $\beta_{i,j}$ are field constants.
Clearly, the first n lines represent the input variables. Any other line l_i, we claim, represents a distinct product gate g_i in ϕ (and vice versa). This claim can be proved by inducting on the maximum number of product gates along the path from any leaf to g_i. (Also notice that the linear combinations present on the RHS of l_i represent the potential sum gates feeding to g_i.)

Now let us count the number of possible SLPs on n inputs. We ignore the first n lines as they are fixed. Among the remaining s lines, for any line l_i, there are i many α's and β's on the RHS. For each of these α's and β's there are $\mathbb{F}$ many choices of values. Therefore the number of circuits of size s is less or equal to $\prod_{i=1}^{s} \mathbb{F}^i\mathbb{F}^i = \mathbb{F}^{\sum_{i=1}^{s} 2i} \le \mathbb{F}^{s^2}$ ■

Let us allow the circuit size to be as large as $(\sqrt{\binom{n+d}{d}})/2$. In other words, say that the polynomial is "hard" to compute if it requires circuit size greater than this size. (Recall that $\binom{n+d}{d}+1$ is the trivial upper bound on the size of a circuit (with product gate fanin d) computing any n-variate degree d polynomial. Furthermore, it is shown by Lovett that for any n-variate degree d polynomial f there exists a circuit computing f having at most $(\sqrt{\binom{n+d}{d}})(nd)^{O(1)}$ multiplications.) So $s = (\sqrt{\binom{n+d}{d}})/2$. Yet, the fraction polynomials computed by circuits of size $s = \mathbb{F}^{s^2}/\mathbb{F}^{\binom{n+d}{d}} = 1/\mathbb{F}^{\frac{3}{4}\binom{n+d}{d}} << 1$. Thus we conclude that the lower bound size is close to the trivial circuit size.

For fields with characteristic zero. An n-variate degree d polynomial has $\binom{n+d}{d}$ monomials and thus its coefficient vector is of dimension $\binom{n+d}{d} = N$(say). Hence the set of all n-variate degree d polynomials forms

an N dimensional vector space.

Referring to the straight line program described above, we can view the operation of a $\Sigma\Pi\Sigma$ circuit as a mapping from α's and β's into a polynomial (which is output). Referring again to the analysis in the previous section, the number of α's and β's is at most s^2. Therefore, the mapping is from a point in an s^2 dimensional space to a point in an N dimensional space. The image of such mapping is an s^2 dimensional object (variety). In general, if $s^2 < N$, then most of the points in the N dimensional space (n-variate degree d polynomials) will be outside this image (set of polynomials computed by $\Sigma\Pi\Sigma$ circuits). In our case indeed $s^2 < N$, as we chose $s = (\sqrt{\binom{n+d}{d}})/2$. Thus we arrive at the same conclusion, namely, that most polynomials are hard to compute.

10.4 References

[1] Amir Shpilka, Avi Wigderson, Depth-3 Arithmetic Circuits over Fields of Characteristic Zero. *Computational Complexity*, 10(1):1-27, 2001.

[2] Ankit Gupta, Pritish Kamat, Neeraj Kayal, Ramprasad Saptharishi Arithmetic circuits: A chasm at depth three. *FOCS 2013*

[3] Shachar Lovett, Computing polynomials with few multiplications. *Theory of Computing, 7(13):185188, 2011.*