CSS.414.1: Polynomial Methods in Combinatorics

Instructor: Mrinal Kumar

TIFR 2024, Aug-Dec

Scribe: Soham Chatterjee

sohamchatterjee999@gmail.com
Website: sohamch08.github.io

2024

Contents

1 Introduction and Targets

The content of this course will be the followings:

- Polynomial Methods in Combinatorics/Geometry

  1. Kakeya/Nikodym Problem over finite fields
  2. Joints Problem
  3. Combinatorial Nullstellensatz (CN)
  4. CN proof of Cauchy-Devenport, Erdös-Heilbronn Conjecture

- Polynomial Methods in Algebraic Algorithms

  1. Noisy Polynomial Interpolation (Sudan, Guruswami-Sudan)
  2. Multiplicative noise (Von zur Gathen-Shparlinski)
  3. Coppersmith's Problem (Given an univariate $f(x)\mathbb{Z}[x]$, compute all 'small' integer roots modulo a composite)

- Polynomial Methods in Circuit Complexity

  1. Razborov-Smolensky (Lower Bound for constant depth AND, OR, NOT, $\bmod p$ gates)
  2. Algorithmic consequences (all pairs shortest paths)
  3. Upper bounds on matrix rigidity (Alman-Williams '2015, Dvir-Edelman '2017)

- Polynomial in Property Testing: Polischuk-Speilman Lemma/Variants

- Weil Bounds (Stepanov, Schmidtm Bombieri)

- Rational Approximations of Algebraic Numbers (Thue[1907] - Siegel - Roth[1954])

2 Joints Problem

3 Combinatorial Nullstellensatz

3.1 Chevally-Warning Theorem

4 Sum Sets

4.1 Sum Sets over Finite Fields

4.1.1 Cauchy-Davenport Theorem

4.2 Restricted Sum Sets

4.2.1 Erdös-Heilbronn Conjecture

5 Arithmetic Progression Free Sets in $\mathbb{F}_3^n$

5.1 3AP Free sets in $\mathbb{F}_q$

6 3-Tensors and Slice Rank

6.1 Rank

6.2 Generalization to 3-Dimension

6.3 Slice Rank of Diagonal 3D Tensor

7 Kakeya and Nikodym Problem

Definition 7.0.1: Kakeya Sets

In a finite field $\mathbb{F}_q$, $K \subseteq \mathbb{F}^n$ is a Kakeya Set if $\forall\, a \in \mathbb{F}^n$, $\exists\, b \in \mathbb{F}^n$ such that

$$L_{a,b} = \{b + at : t \in \mathbb{F}_q\} \subseteq K$$

i.e. informally it has a line in every direction

Now notice that we can take the whole $\mathbb{F}_q^n$ as the Kakeya Set. We can also remove a point from $\mathbb{F}_q^n$ and it will still be a Kakeya Set. Having defined the Kakeya sets the biggest question which is studied is:

Question 7.1

How small can a Kakeya Set be?

7.1 Lower Bound on Nikodym Sets

7.2 Lower Bound on Kakeya Sets

7.2.1 Hasse Derivative

8 Razborov Smolensky Lower Bound

The result we will discuss the result that majority is strictly harder than the parity for AC^0, since there is no polynomial-size AC^0 circuit to compute majority even if we are given parity gates. The result is Razborov's, and the proof technique uses ideas due to both Razborov and Smolensky.

Consider the class AC^0 of polynomial size circuits with constant depth with unbounded fan-in. We consider the class $AC^0(\oplus)$ where we are give the parity gates $\oplus$ which outputs 1 if an odd number of its inputs are 1. The main theorem which we will prove in this section is:

Theorem 8.1 Razborov-Smolensky

For any $d \in \mathbb{N}$ any any depth d $AC^0(\oplus)$ circuit for Majority has size $\geq 2^{\Omega\left(n^{\frac{1}{2d}}\right)}$

8.1 Two Parts of Proving Lower Bound

The proof of the above theorem requires two lemmas:

Lemma 8.1.1

$\forall\ \epsilon > 0$ and $d \in \mathbb{N}$ the following is true:

If $f : \{0,1\}^n \to \{0,1\}$ can be computed by a size s depth d $AC^0(\oplus)$ circuit then $\exists$ a polynomial g in n variables and deg $O\left(\log \frac{s}{\epsilon}\right)^d$ such that

$$\mathop{\mathbb{P}}_{a\in\{0,1\}^n}[f(a) = g(a)] \geq 1-\epsilon$$

Lemma 8.1.2

For all polynomials $p(x_1, \ldots, x_n)$ with deg $p = t$,

$$\mathop{\mathbb{P}}_{a\in\{0,1\}^n}[g(a) = \textsc{Maj}(a)] \leq \frac{1}{2} + O\left(\frac{t}{\sqrt{n}}\right)$$

Now first we will show that with these two lemmas we can prove Razborov-Smolensky Lower Bound for Majority function

Proof of Theorem 8.1: Suppose Maj has a $AC^0(\oplus)$ circuit of size $< 2^{n^{\frac{1}{2d}-\delta}}$

$\xRightarrow{\text{Lemma 8.1.1}} \exists$ polynomial g of degree $n^{\frac{1}{2d}-\delta}$ that approximates Maj with error 0.1.

$\xRightarrow{\text{Lemma 8.1.2}} \forall$ polynomial g of deg $n^{\frac{1}{2d}-\delta}$ the error is $\geq 1 - \left[\frac{1}{2} + O\left(\frac{n^{\frac{1}{2d}-\delta}}{\sqrt{n}}\right)\right] \geq \frac{1}{2} - - \left[\frac{1}{2} + O\left(\frac{n^{\frac{1}{2d}-\delta}}{\sqrt{n}}\right)\right] \geq \frac{1}{2} - o(1)$

But $\frac{1}{2} - o(1) < 0.1$ is contradiction. ∎

Alternate Proof Theorem 8.1: Suppose C be an $AC^0(\oplus)$ circuit of size s and depth d computing Majority

$\xRightarrow{\text{Lemma 8.1.1}} \exists$ polynomial g of degree $O\left(\log \frac{s}{\epsilon}\right)^d$ with error probability $\leq \epsilon$.

$\xRightarrow{\text{Lemma 8.1.2}} \forall$ polynomial g of deg $O\left(\log \frac{s}{\epsilon}\right)^d$ the error is $\geq \frac{1}{2} + O\left(\frac{\left(\log \frac{s}{\epsilon}\right)^d}{\sqrt{n}}\right)$.

Hence from these two results and setting $\epsilon = 0.1$ we have

$$\frac{1}{2} + O\left(\frac{\left(\log \frac{s}{\epsilon}\right)^d}{\sqrt{n}}\right) \geq 1-\epsilon \implies (\log 10s)^d \geq \sqrt{n} \implies s \geq 2^{\Omega\left(\frac{1}{2d}\right)}$$

∎

Now that we proved our main objective theorem we will focus on proving the 2 lemmas in the following two sections.

8.2 Approximating Boolean Function with Polynomials

We first state and prove a lemma showing that every $AC^0(\oplus)$ circuit can be approximated by a low degree polynomial i.e. Lemma 8.1.1. But to prove that we will show a more stronger lemma and then the lemma follows as a simple corollary of this stronger result.

Lemma 8.2.1

For all $\mathrm{AC}^0(\oplus)$ circuits C of size s of depth d and $\forall\, \epsilon > 0$ there exists a distribution $\mathscr{D}$ of polynomials $p(x_1, \ldots, x_n) \in \mathbb{F}_2[x_1, \ldots, x_n]$ such that for all $a \in \{0,1\}^n$

$$\mathop{\mathbb{P}}_{p \in \mathscr{D}}[p(a) = C(a)] \geq 1 - \epsilon$$

where $\mathscr{D}$ is supported on polynomials of degree $\leq \left(\log \frac{s}{\epsilon}\right)^d$

First we will show that this lemma implies Lemma 8.1.1.

Proof of Lemma Lemma 8.1.1: Consider the $|\{0,1\}^n| \times |\operatorname{supp} \mathscr{D}|$ table for each $a \in \{0,1\}^n$, a represents a row in the table. In the table at $(a,i)^{th}$ entry put 1 if i^{th} polynomial p in $\mathscr{D}$ satisfies $p(a) = C(a)$. For rest of the positions put 0.

$\xRightarrow{\text{Lemma 8.2.1}}$ $\forall\, \epsilon > 0$ there exists a distribution $\mathscr{D}$ such that for all $a \in \{0,1\}^n$ such that $\mathop{\mathbb{P}}_{p \in (\mathscr{D})}[p(a) = C(a)] \geq 1 - \epsilon$. Hence in the table for each $a \in \{0,1\}^n$, at least $1 - \epsilon$ many fraction of $|\operatorname{supp}(\mathscr{D})|$ entries in a^{th} row have 1. Therefore there are total at least $(1-\epsilon) \cdot |\{0,1\}^n| \cdot |\operatorname{supp}(\mathscr{D})|$ many 1's in total in the table.

Hence by pigeon hole principle there is at least one column which has at least $(1-\epsilon) \cdot |\{0,1\}^n|$ many 1's. Therefore there is a polynomial $p \in \operatorname{supp}(\mathscr{D})$ which agrees with C in at least $1 - \epsilon$ fraction of total inputs. Hence

$$\mathop{\mathbb{P}}_{a \in \{0,1\}^n}[p(a) = C(a)] \geq 1 - \epsilon$$

■

Now we will prove the Lemma 8.2.1. Now before diving into the proof first let's see how can we approximate the gates in $\mathrm{AC}^0(\oplus)$ circuits with low-degree polynomials. That way we can approximate any $\mathrm{AC}^0(\oplus)$ circuit with low-degree polynomial.

So to for a $\neg x_i$ gate we can have the polynomial $1 - x_i$. For a $\bigoplus_{i=1}^{k} x_i$ we can use the polynomial $\sum_{i=1}^{k} x_i$. So only $\wedge$ and $\vee$ gates are remaining. Now notice if we have a low degree polynomial for $\wedge$ we also have a low degree polynomial for $\vee$ since

$$\bigvee_{i=1}^{n} x_i = \neg\left(\bigwedge_{i=1}^{n} (\neg x_i)\right)$$

So we will try to find a polynomial approximating an $\wedge$ gate of degree $\leq \left(\log \frac{1}{\epsilon}\right)^d$. We can't approximate $\wedge$ by outputting 0 every time since the desired correctness probability must hold for all inputs x. Multiplying a random constant-size subset of the bits will not work either, for the same reason.

Naive way to have a polynomial for $\bigvee_{i=1}^{n} x_i$ would be $1 - \prod_{i=1}^{n}(1 - x_i)$. But with this the degree becomes very large.

Idea. *Check parity of random subset of $[n]$. So we take a random subset $S \subseteq [n]$ then we take the polynomial $p_S = \sum_{i \in S} x_i$.*

Lemma 8.2.2

If S is a random subset of $[n]$ then

$$\mathbb{P}\left[p_S(x_1, \ldots, x_n) = \bigwedge_{i=1}^{n} x_i\right] = \frac{1}{2}$$

8.3 Degree-Error Trade of to Approximate MAJORITY