# On the existence of algebraically natural proofs

Joint work with Prerona Chatterjee, C. Ramya, Ramprasad Saptharishi and Anamay Tengse

# Polynomials

- Main protagonists: multivariate polynomials over a field $\mathbb{F}$
- $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, $\deg(P) = d$
- $\mathbb{F}$ : complex numbers

# Polynomials

- Main protagonists: multivariate polynomials over a field $\mathbb{F}$
- $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, $\deg(P) = d$
- $\mathbb{F}$ : complex numbers

- Algebraic complexity: the cost of computing polynomials as formal objects

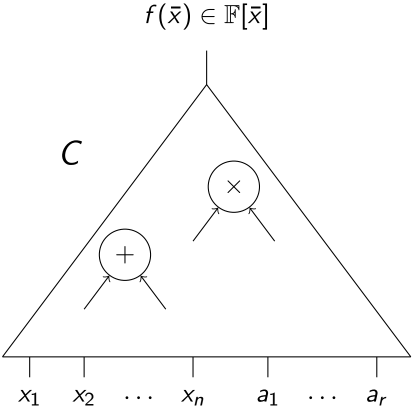
# Polynomials

- Main protagonists: multivariate polynomials over a field $\mathbb{F}$
- $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, $\deg(P) = d$
- $\mathbb{F}$ : complex numbers

- Algebraic complexity: the cost of computing polynomials as formal objects

- Variables - $\bar{x} = \{x_1, \ldots, x_n\}$, constants - $\mathbb{F} = \mathbb{C}$
  Operations - Addition $+$ and multiplication $\times$.

# Algebraic Circuits




**Parameters**:

Size($C$)

- No. of gates

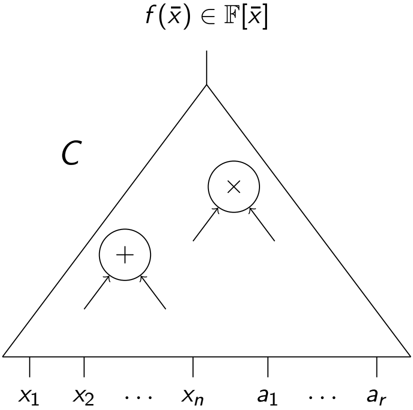
or no. of wires

Depth($C$)

- Longest path from

root to a leaf

# Algebraic Circuits




**Parameters**:

Size($C$)

- No. of gates

or no. of wires

Depth($C$)

- Longest path from

root to a leaf

# Easy and Hard Polynomials [Val79]

Parameters: Number of variables - $n$, degree - $d$
This talk: $d \sim \text{poly}(n)$

# Easy and Hard Polynomials [Val79]

Parameters: Number of variables - $n$, degree - $d$
This talk: $d \sim \text{poly}(n)$

**Definition** (VP - Easy Polynomials):

Class of all $n$-variate, degree $d = \text{poly}(n)$ polynomials, computable by circuits of size $\text{poly}(n)$. E.g. the Determinant.

# Easy and Hard Polynomials [Val79]

Parameters: Number of variables - $n$, degree - $d$
This talk: $d \sim \text{poly}(n)$

**Definition** (VP - Easy Polynomials):

Class of all $n$-variate, degree $d = \text{poly}(n)$ polynomials, computable by circuits of size $\text{poly}(n)$. E.g. the Determinant.

**Definition** (VP - Easy Polynomials):

Class of all $n$-variate, degree $d = \text{poly}(n)$ polynomials, for which it is reasonably easy to compute the coefficient of any given monomial. E.g. Permanent.

# Easy and Hard Polynomials [Val79]

Parameters: Number of variables - $n$, degree - $d$
This talk: $d \sim \text{poly}(n)$

**Definition** (VP - Easy Polynomials):

Class of all $n$-variate, degree $d = \text{poly}(n)$ polynomials, computable by circuits of size $\text{poly}(n)$. E.g. the Determinant.

**Definition** (VP - Easy Polynomials):

Class of all $n$-variate, degree $d = \text{poly}(n)$ polynomials, for which it is reasonably easy to compute the coefficient of any given monomial. E.g. Permanent.

VP **vs** VNP: Lower bounds for explicit polynomials.

# Finding Hard Polynomials

- General Lower Bounds:

# Finding Hard Polynomials

- General Lower Bounds:
  - Circuits: $\Omega(n \log d)$ [BS83,Smo97]
  - Formulas: $\Omega(n^2)$ [Kal85, SY08, CKSV20]

# Finding Hard Polynomials

- General Lower Bounds:
  - Circuits: $\Omega(n \log d)$ [BS83,Smo97]
  - Formulas: $\Omega(n^2)$ [Kal85, SY08, CKSV20]

- Many structured cases:
  - Constant depth circuits [NW95,KST16,GKKS13,...]
  - Multilinear models [Raz09,DMPY12,...]
  - Non-commutative models [Nis91,LMP16,CILM18,...]
  - Monotone models [Yeh19,Sri19]

# Finding Hard Polynomials

- General Lower Bounds:
  - Circuits: $\Omega(n \log d)$ [BS83,Smo97]
  - Formulas: $\Omega(n^2)$ [Kal85, SY08, CKSV20]
- Many structured cases:
  - Constant depth circuits [NW95,KST16,GKKS13,...]
  - Multilinear models [Raz09,DMPY12,...]
  - Non-commutative models [Nis91,LMP16,CILM18,...]
  - Monotone models [Yeh19,Sri19]

**Observation**: Most of the proofs follow a certain template.

# Finding Hard Polynomials

- General Lower Bounds:
  - Circuits: $\Omega(n \log d)$ [BS83,Smo97]
  - Formulas: $\Omega(n^2)$ [Kal85, SY08, CKSV20]
- Many structured cases:
  - Constant depth circuits [NW95,KST16,GKKS13,...]
  - Multilinear models [Raz09,DMPY12,...]
  - Non-commutative models [Nis91,LMP16,CILM18,...]
  - Monotone models [Yeh19,Sri19]

**Observation**: Most of the proofs follow a certain template.

Can proofs based on this template yield strong lower bounds ?

# The Template

# The template: a toy case

**Circuit class**: $\mathcal{C} = \{(\alpha t - \beta)^2 : \alpha, \beta \in \mathbb{C}\}$.

# The template: a toy case

**Circuit class**: $\mathcal{C} = \{(\alpha t - \beta)^2 : \alpha, \beta \in \mathbb{C}\}$.

Finding explicit $h \notin \mathcal{C}$:

# The template: a toy case

**Circuit class**: $\mathcal{C} = \{(\alpha t - \beta)^2 : \alpha, \beta \in \mathbb{C}\}$.

Finding explicit $h \notin \mathcal{C}$:

- An Equation of $\mathcal{C}$:
  If $f(t) = at^2 + bt + c \in \mathcal{C}$, then $b^2 - 4ac = 0$.

# The template: a toy case

**Circuit class**: $\mathcal{C} = \{(\alpha t - \beta)^2 : \alpha, \beta \in \mathbb{C}\}$.

<u>Finding explicit $h \notin \mathcal{C}$</u>:

- An Equation of $\mathcal{C}$:
  If $f(t) = at^2 + bt + c \in \mathcal{C}$, then $b^2 - 4ac = 0$.
- A Hard Polynomial:
  $h(t) = a't^2 + b't + c'$ such that $b'^2 - 4a'c' \neq 0$.

# Waring rank: a real world example

**Theorem**

*If* $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ *for linear forms* $L_1, L_2, \ldots, L_s$*, then* $s$ *is at least* $\exp(\Omega(n))$.

# Waring rank: a real world example

**Theorem**

*If* $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ *for linear forms* $L_1, L_2, \ldots, L_s$*, then* $s$ *is at least* $\exp(\Omega(n))$*.*

$\mathcal{C} \equiv$ Polynomials with small waring rank. The goal is to show that the monomial $x_1 x_2 \cdots x_n$ is not in $\mathcal{C}$.

# Waring rank: a real world example

**Theorem**
*If* $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ *for linear forms* $L_1, L_2, \ldots, L_s$*, then* $s$ *is at least* $\exp(\Omega(n))$.

$\mathcal{C} \equiv$ Polynomials with small waring rank. The goal is to show that the monomial $x_1 x_2 \cdots x_n$ is not in $\mathcal{C}$.

Partial derivatives complexity: dimension of the linear space spanned by partial derivatives

# Waring rank: a real world example

**Theorem**
*If $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ for linear forms $L_1, L_2, \ldots, L_s$, then $s$ is at least $\exp(\Omega(n))$.*

$\mathcal{C} \equiv$ Polynomials with small waring rank. The goal is to show that the monomial $x_1 x_2 \cdots x_n$ is not in $\mathcal{C}$.

Partial derivatives complexity: dimension of the linear space spanned by partial derivatives

- For $\mathcal{C}$: dimension $\leq O(sn)$ [Chain rule + sub-additivity]

# Waring rank: a real world example

**Theorem**
*If* $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ *for linear forms* $L_1, L_2, \ldots, L_s$*, then* $s$ *is at least* $\exp(\Omega(n))$*.*

$\mathcal{C} \equiv$ Polynomials with small waring rank. The goal is to show that the monomial $x_1 x_2 \cdots x_n$ is not in $\mathcal{C}$.

Partial derivatives complexity: dimension of the linear space spanned by partial derivatives

- For $\mathcal{C}$: dimension $\leq O(sn)$ [Chain rule + sub-additivity]
- For the monomial: dimension $\geq \exp(\Omega(n))$ [distinct multilinear monomials]

# Waring rank: a real world example

**Theorem**
*If* $x_1 \cdots x_n = L_1^n + L_2^n + \cdots + L_s^n$ *for linear forms* $L_1, L_2, \ldots, L_s$*, then* $s$ *is at least* $\exp(\Omega(n))$.

$\mathcal{C} \equiv$ Polynomials with small waring rank. The goal is to show that the monomial $x_1 x_2 \cdots x_n$ is not in $\mathcal{C}$.

Partial derivatives complexity: dimension of the linear space spanned by partial derivatives

- For $\mathcal{C}$: dimension $\leq O(sn)$ [Chain rule + sub-additivity]
- For the monomial: dimension $\geq \exp(\Omega(n))$ [distinct multilinear monomials]

So, for the monomial to be in $\mathcal{C}$, we must have $sn \geq \exp(\Omega(n))$.

# Waring rank: a real world example

The partial derivative matrix: rows and columns indexed by monomials
$(\alpha, \beta)$ entry $=$ coefficient of the monomial $\beta$ in the partial derivative $\frac{\partial P}{\partial \alpha}$

- Every entry is linear in the coefficients of $P$
- Dim of matrix: $N \times N$ for $N = \binom{n+d}{d}$
- Partial derivative complexity $\equiv$ rank of this matrix over $\mathbb{F}$

# Waring rank: a real world example

The partial derivative matrix: rows and columns indexed by monomials
$(\alpha, \beta)$ entry $=$ coefficient of the monomial $\beta$ in the partial derivative $\frac{\partial P}{\partial \alpha}$

- Every entry is linear in the coefficients of $P$
- Dim of matrix: $N \times N$ for $N = \binom{n+d}{d}$
- Partial derivative complexity $\equiv$ rank of this matrix over $\mathbb{F}$

Previous proof: there exists a submatrix which is full rank for $x_1 x_2 \cdots x_n$ and is rank deficient for polynomials of small Waring rank

# Waring rank: a real world example

The partial derivative matrix: rows and columns indexed by monomials
$(\alpha, \beta)$ entry = coefficient of the monomial $\beta$ in the partial derivative $\frac{\partial P}{\partial \alpha}$

- Every entry is linear in the coefficients of $P$
- Dim of matrix: $N \times N$ for $N = \binom{n+d}{d}$
- Partial derivative complexity $\equiv$ rank of this matrix over $\mathbb{F}$

Previous proof: there exists a submatrix which is full rank for $x_1 x_2 \cdots x_n$ and is rank deficient for polynomials of small Waring rank
In particular: the determinant of this minor vanishes on coefficient vector of every polynomial in $\mathcal{C}$ and is non-zero on the coefficient vector of $x_1 x_2 \cdots x_n$.

# Natural proofs of algebraic lower bounds

# Natural proofs of algebraic lower bounds

Variables $\bar{x} = \{x_1, \ldots, x_n\}$, Degree - $d$, Field $\mathbb{F}$.

# Natural proofs of algebraic lower bounds

Variables $\bar{x} = \{x_1, \ldots, x_n\}$, Degree - $d$, Field $\mathbb{F}$.

$\mathcal{M}$ - monomials in $\bar{x}$ of degree $d$, $N = |\mathcal{M}| = \binom{n+d}{n}$.

# Natural proofs of algebraic lower bounds

Variables $\bar{x} = \{x_1, \ldots, x_n\}$, Degree - $d$, Field $\mathbb{F}$.

$\mathcal{M}$ - monomials in $\bar{x}$ of degree $d$, $N = |\mathcal{M}| = \binom{n+d}{n}$.

$$f(x_1, \ldots, x_n) = \sum_{m \in \mathcal{M}} f_m \cdot m \qquad f_m = \mathrm{coeff}_f(m)$$

# Natural proofs of algebraic lower bounds

Variables $\bar{x} = \{x_1, \ldots, x_n\}$, Degree - $d$, Field $\mathbb{F}$.

$\mathcal{M}$ - monomials in $\bar{x}$ of degree $d$, $N = |\mathcal{M}| = \binom{n+d}{n}$.

$$f(x_1, \ldots, x_n) = \sum_{m \in \mathcal{M}} f_m \cdot m \qquad f_m = \mathrm{coeff}_f(m)$$

Let $\mathrm{coeffs}(f) = [f_{m_1}, f_{m_2}, \ldots, f_{m_N}] \in \mathbb{F}^N$.

# Natural proofs of algebraic lower bounds

Variables $\bar{x} = \{x_1, \ldots, x_n\}$, Degree - $d$, Field $\mathbb{F}$.

$\mathcal{M}$ - monomials in $\bar{x}$ of degree $d$, $N = |\mathcal{M}| = \binom{n+d}{n}$.

$$f(x_1, \ldots, x_n) = \sum_{m \in \mathcal{M}} f_m \cdot m \qquad f_m = \mathrm{coeff}_f(m)$$

Let $\mathrm{coeffs}(f) = [f_{m_1}, f_{m_2}, \ldots, f_{m_N}] \in \mathbb{F}^N$.

**Definition** (Equation)

A non-zero polynomial $P$ is said to be an equation for a class $\mathcal{C}$, if $P(\mathrm{coeffs}(f)) = 0$ for all $f \in \mathcal{C}$.

# Natural proofs of algebraic lower bounds

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$;     let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation,

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- $P$ is "easy" to compute (e.g. circuit size and degree poly$(N)$).

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- $P$ is "easy" to compute (e.g. circuit size and degree poly$(N)$).
- $P(\text{coeffs}(g_0)) \neq 0$ for the candidate hard polynomial $g_0$

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bounds for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- $P$ is "easy" to compute (e.g. circuit size and degree poly($N$)).
- $P(\text{coeffs}(g_0)) \neq 0$ for the candidate hard polynomial $g_0$(in fact, for most polynomials).

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bound for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- **Usefulness:** $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- **Constructivity:** $P$ is "easy" to compute (e.g. circuit size and degree $\text{poly}(N)$).
- **Largeness:** $P(\text{coeffs}(g_0)) \neq 0$ for the candidate hard polynomial $g_0$ (in fact, for most polynomials).

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bound for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- **Usefulness:** $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- **Constructivity:** $P$ is "easy" to compute (e.g. circuit size and degree poly$(N)$).
- **Largeness:** $P(\text{coeffs}(g_0)) \neq 0$ for the candidate hard polynomial $g_0$ (in fact, for most polynomials).

**Q.** Can we hope to prove superpolynomial lower bounds for algebraic circuits via natural proofs ?

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bound for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- **Usefulness:** $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- **Constructivity:** $P$ is "easy" to compute (e.g. circuit size and degree poly($N$)).
- **Largeness:** $P(\text{coeffs}(g_0)) \neq 0$ for the candidate hard polynomial $g_0$ (in fact, for most polynomials).

**Q.** Does VP have an efficiently constructible equations ?[AD,G,FSV,GKSS]

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich:** (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich:** (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

**Rich enough :** Candidate construction of pseudorandom functions in the class.

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich**: (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

**Rich enough :** Candidate construction of pseudorandom functions in the class.

- Unclear if this applies to lower bounds for VP.

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich:** (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

**Rich enough :** Candidate construction of pseudorandom functions in the class.

- Unclear if this applies to lower bounds for VP. Pseudorandom functions via algebraic circuits of small size and degree ?

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich:** (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

**Rich enough :** Candidate construction of pseudorandom functions in the class.

- Unclear if this applies to lower bounds for VP. Pseudorandom functions via algebraic circuits of small size and degree ?
- Only need to fool algebraic circuits.

# Boolean vs Algebraic Natural proofs

**Razborov-Rudich:** (Under standard assumptions) Natural proofs cannot yield lower bounds for *rich enough* classes of Boolean circuits.

**Rich enough :** Candidate construction of pseudorandom functions in the class.

- Unclear if this applies to lower bounds for VP. Pseudorandom functions via algebraic circuits of small size and degree ?
- Only need to fool algebraic circuits.
- Not enough evidence, one way or the other.

# Natural proofs of algebraic lower bounds

For $n$, $d$ and $N = \binom{n+d}{n}$; let $\mathcal{U} = \mathbb{F}^N$, $\mathcal{C}_n \subset \mathbb{F}^N$.

Natural proof of lower bound for $\mathcal{C}$: based on showing that $\mathcal{C}$ has an efficiently constructible equation, i.e. there is a polynomial $P(Z_1, \ldots, Z_N)$ such that:

- **Usefulness:** $P(\text{coeffs}(f)) = 0$ for all $f \in \mathcal{C}_n$.
- **Constructivity:** $P$ is "easy" to compute (e.g. circuit size and degree poly$(N)$).
- **Largeness:** $P(\text{coeffs}(g_0)) \neq 0$ for $g_0$ (in fact, for most polynomials $g$).

**Q.** Does VP have efficiently constructible equations ?[AD,G,FSV,GKSS]

# What do we know ?

- **Natural Proofs** [FSV18]

# What do we know ?

- **Natural Proofs** [FSV18]
  - Reformulate this question as a question about succinct derandomization of polynomial identity testing.

# What do we know ?

- **Natural Proofs** [FSV18]
  - Reformulate this question as a question about succinct derandomization of polynomial identity testing.
  - For more structured notions of constructivity (sparsity/Waring rank), the answer is negative.

# What do we know ?

- **Natural Proofs** [FSV18]
  - Reformulate this question as a question about succinct derandomization of polynomial identity testing.
  - For more structured notions of constructivity (sparsity/Waring rank), the answer is negative.

- **Variety Membership** [BIJL18,BIL+19]
  - Hardness of membership testing rules out efficient equations for certain classes.

# What do we know ?

- **Natural Proofs** [FSV18]
  - Reformulate this question as a question about succinct derandomization of polynomial identity testing.
  - For more structured notions of constructivity (sparsity/Waring rank), the answer is negative.
- **Variety Membership** [BIJL18,BIL+19]
  - Hardness of membership testing rules out efficient equations for certain classes.
- **Rank Methods** [EGOW18,GMOW19]
  - *Rank-based methods* will not show optimal lower bounds.
  - Tensor rank lower bounds do not lift to higher dimensions.

# What do we know ?

- **Natural Proofs** [FSV18]
  - Reformulate this question as a question about succinct derandomization of polynomial identity testing.
  - For more structured notions of constructivity (sparsity/Waring rank), the answer is negative.

- **Variety Membership** [BIJL18,BIL+19]
  - Hardness of membership testing rules out efficient equations for certain classes.

- **Rank Methods** [EGOW18,GMOW19]
  - *Rank-based methods* will not show optimal lower bounds.
  - Tensor rank lower bounds do not lift to higher dimensions.

**Q.** Does VP have efficiently constructible equations ?

# Our results

# Main Theorem

**Q.** Does VP have efficiently constructible equations ??

# Main Theorem

**Q.** Does VP have efficiently constructible equations ??

**A.** For a natural special case: polynomials with small integer coefficients, the answer is YES.

# Main Theorem

**Q.** Does VP have efficiently constructible equations ??

**A.** For a natural special case: polynomials with small integer coefficients, the answer is YES.

**Theorem** (Equations for $\mathrm{VP}'_{\mathbb{C}}$):

For $n$,$d$ and $N = \binom{n+d}{n}$, there exists a nonzero $P(Z_1, \ldots, Z_N)$ in $\mathrm{VP}(N)$ such that

- for all $f \in \mathrm{VP}(n, d)$ *with small integer coefficients*, $P(\mathrm{coeffs}(f)) = 0$

# Main Theorem

**Q.** Does VP have efficiently constructible equations ??

**A.** For a natural special case: polynomials with small integer coefficients, the answer is YES.

**Theorem** (Equations for $\mathrm{VP}'_{\mathbb{C}}$):

For $n$,$d$ and $N = \binom{n+d}{n}$, there exists a nonzero $P(Z_1, \ldots, Z_N)$ in $\mathrm{VP}(N)$ such that

- for all $f \in \mathrm{VP}(n, d)$ *with small integer coefficients*, $P(\mathrm{coeffs}(f)) = 0$
- there exists a polynomial $g$ with small integer coefficients such that $P(\mathrm{coeffs}(g)) \neq 0$

# Main Theorem

**Q.** Does VP have efficiently constructible equations ??

**A.** For a natural special case: polynomials with small integer coefficients, the answer is YES.

**Theorem** (Equations for $\mathrm{VP}'_{\mathbb{C}}$):

For $n$,$d$ and $N = \binom{n+d}{n}$, there exists a nonzero $P(Z_1, \dots, Z_N)$ in $\mathrm{VP}(N)$ such that

- for all $f \in \mathrm{VP}(n, d)$ *with small integer coefficients*, $P(\mathrm{coeffs}(f)) = 0$
- there exists a polynomial $g$ with small integer coefficients such that $P(\mathrm{coeffs}(g)) \neq 0$

Restriction **not on circuits** computing the polynomials.

# To summarize

- A natural, rich and computationally interesting (although finite) subset of VP has an efficiently constructible equation.

# To summarize

- A natural, rich and computationally interesting (although finite) subset of VP has an efficiently constructible equation. Doesn't seem to say anything about all of VP, but is still seems a bit surprising.

# To summarize

- A natural, rich and computationally interesting (although finite) subset of VP has an efficiently constructible equation. Doesn't seem to say anything about all of VP, but is still seems a bit surprising.
- For polynomials with small integer coefficients (e.g Permanent), we might still have a lower bound proof which is via a useful and efficiently constructible algebraic property (a constructible equation). But we cannot guarantee largeness.

# To summarize

- A natural, rich and computationally interesting (although finite) subset of VP has an efficiently constructible equation. Doesn't seem to say anything about all of VP, but is still seems a bit surprising.
- For polynomials with small integer coefficients (e.g Permanent), we might still have a lower bound proof which is via a useful and efficiently constructible algebraic property (a constructible equation). But we cannot guarantee largeness.

# Sketch of the Proofs

# Hitting sets for VP

**Definition** (Hitting Set)

$\mathcal{H} \subset \mathbb{F}^n$ is a *hitting set* for a class $\mathcal{C}$ of $n$-variate polynomials, if for all $0 \neq f \in \mathcal{C}$, there exists an $h \in \mathcal{H}$ such that $f(h) \neq 0$.

# Hitting sets for VP

**Definition** (Hitting Set)

$\mathcal{H} \subset \mathbb{F}^n$ is a *hitting set* for a class $\mathcal{C}$ of $n$-variate polynomials, if for all $0 \neq f \in \mathcal{C}$, there exists an $h \in \mathcal{H}$ such that $f(h) \neq 0$.

**Theorem** [HS80,For14]

There *exist* hitting sets of size $\text{poly}(n, d, s)$ for the class of $n$-variate, degree $d$ polynomials that have circuits of size $s$.

# Hitting sets for VP

**Definition** (Hitting Set)

$\mathcal{H} \subset \mathbb{F}^n$ is a *hitting set* for a class $\mathcal{C}$ of $n$-variate polynomials, if for all $0 \neq f \in \mathcal{C}$, there exists an $h \in \mathcal{H}$ such that $f(h) \neq 0$.

**Theorem** [HS80,For14]

There *exist* hitting sets of size $\text{poly}(n, d, s)$ for the class of $n$-variate, degree $d$ polynomials that have circuits of size $s$.

Moreover, there is a hitting set with small integer points.

# Hitting sets for VP

**Definition** (Hitting Set)

$\mathcal{H} \subset \mathbb{F}^n$ is a *hitting set* for a class $\mathcal{C}$ of $n$-variate polynomials, if for all $0 \neq f \in \mathcal{C}$, there exists an $h \in \mathcal{H}$ such that $f(h) \neq 0$.

**Theorem** [HS80,For14]

There *exist* hitting sets of size $\text{poly}(n, d, s)$ for the class of $n$-variate, degree $d$ polynomials that have circuits of size $s$.

Moreover, there is a hitting set with small integer points.

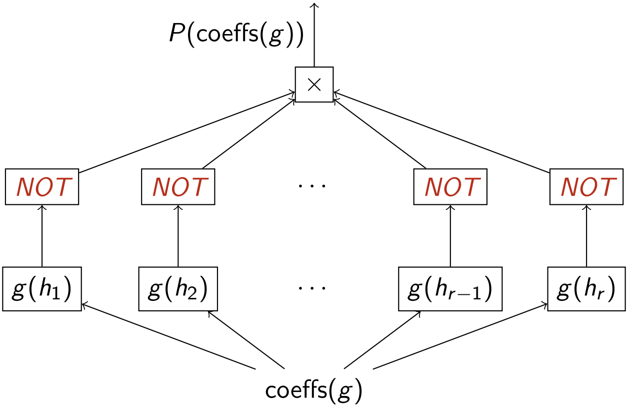
**Observation**: For a nonzero $g$, $g(\mathcal{H}) = 0$ is a proof that $g \notin \mathcal{C}$.

# From hitting set to equations

$\mathcal{H} = \{h_1, \ldots, h_r\}$ hitting set for $\mathcal{C}$, $\qquad 0 \neq g(\bar{x})$ input polynomial.

# From hitting set to equations

$\mathcal{H} = \{h_1, \ldots, h_r\}$ hitting set for $\mathcal{C}$, $\quad 0 \neq g(\bar{x})$ input polynomial.




$NOT(0) =$nonzero $\qquad NOT(\text{nonzero}) = 0$

# Evaluating at a point

**Given**: Vector coeffs$(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

# Evaluating at a point

**Given**: Vector coeffs$(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

coeffs$(g) = [g_{m_1}, g_{m_2}, \ldots, g_{m_N}]$, $\{m_1, \ldots, m_N\} = \mathcal{M}$.

# Evaluating at a point

**Given**: Vector coeffs$(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

coeffs$(g) = [g_{m_1}, g_{m_2}, \ldots, g_{m_N}]$, $\{m_1, \ldots, m_N\} = \mathcal{M}$.

Let eval$(h) = [m_1(h), m_2(h), \ldots, m_N(h)]$.

# Evaluating at a point

**Given**: Vector $\text{coeffs}(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

$\text{coeffs}(g) = [g_{m_1}, g_{m_2}, \ldots, g_{m_N}]$, $\{m_1, \ldots, m_N\} = \mathcal{M}$.

Let $\text{eval}(h) = [m_1(h), m_2(h), \ldots, m_N(h)]$.

Now $g(h) = \langle \text{coeffs}(g), \text{eval}(h) \rangle = \sum_{m \in \mathcal{M}} g_m m(h)$.

# Evaluating at a point

**Given**: Vector $\mathrm{coeffs}(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

$\mathrm{coeffs}(g) = [g_{m_1}, g_{m_2}, \ldots, g_{m_N}]$, $\{m_1, \ldots, m_N\} = \mathcal{M}$.

Let $\mathrm{eval}(h) = [m_1(h), m_2(h), \ldots, m_N(h)]$.

Now $g(h) = \langle \mathrm{coeffs}(g), \mathrm{eval}(h) \rangle = \sum_{m \in \mathcal{M}} g_m m(h)$.

**Note**:

- Linear polynomial in $\mathrm{coeffs}(g)$.

# Evaluating at a point

**Given**: Vector $\text{coeffs}(g) \in \mathbb{F}^N$, point $h \in \mathbb{F}^n$

$\text{coeffs}(g) = [g_{m_1}, g_{m_2}, \ldots, g_{m_N}]$, $\{m_1, \ldots, m_N\} = \mathcal{M}$.

Let $\text{eval}(h) = [m_1(h), m_2(h), \ldots, m_N(h)]$.

Now $g(h) = \langle \text{coeffs}(g), \text{eval}(h) \rangle = \sum_{m \in \mathcal{M}} g_m m(h)$.

**Note**:

- Linear polynomial in $\text{coeffs}(g)$.
- We can "hardwire" $\text{eval}(h)$ in our circuit, for all $h \in \mathcal{H}$.

# Algebraic NOT - Finite Fields

**Given**: Vector $\text{coeffs}(g) \in \mathbb{F}_q^N$, point $h \in \mathbb{F}_q^n$

# Algebraic NOT - Finite Fields

**Given**: Vector coeffs$(g) \in \mathbb{F}_q^N$, point $h \in \mathbb{F}_q^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

# Algebraic NOT - Finite Fields

**Given**: Vector coeffs$(g) \in \mathbb{F}_q^N$,  point $h \in \mathbb{F}_q^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

For all $0 \neq x \in \mathbb{F}_q$, $x^{q-1} - 1 = 0$

# Algebraic NOT - Finite Fields

**Given**: Vector $\mathrm{coeffs}(g) \in \mathbb{F}_q^N$, point $h \in \mathbb{F}_q^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

For all $0 \neq x \in \mathbb{F}_q$, $x^{q-1} - 1 = 0$

**Output**: $(\langle \mathrm{coeffs}(g), \mathrm{eval}(h) \rangle)^{q-1} - 1$.

# Algebraic NOT - Finite Fields

**Given**: Vector $\text{coeffs}(g) \in \mathbb{F}_q^N$, point $h \in \mathbb{F}_q^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

For all $0 \neq x \in \mathbb{F}_q$, $x^{q-1} - 1 = 0$

**Output**: $(\langle \text{coeffs}(g), \text{eval}(h) \rangle)^{q-1} - 1$.

$$P(\text{coeffs}(g)) \approx \prod_{h \in \mathcal{H}} \left( (\langle \text{coeffs}(g), \text{eval}(h) \rangle)^{q-1} - 1 \right)$$

$\text{Degree}(P) \leq |\mathcal{H}| q \leq \text{poly}(N)$, $\text{Size}(P) \leq \text{poly}(N)$.

# Finite Fields: a hard polynomial

**Want**: $f$ with coefficients in $\mathbb{F}_q$ such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

# Finite Fields: a hard polynomial

**Want**: $f$ with coefficients in $\mathbb{F}_q$ such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

# Finite Fields: a hard polynomial

**Want**: $f$ with coefficients in $\mathbb{F}_q$ such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

# Finite Fields: a hard polynomial

**Want**: $f$ with coefficients in $\mathbb{F}_q$ such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

$$P(\mathrm{coeffs}(f)) \approx \prod_{h \in \mathcal{H}} \left( (\langle \mathrm{coeffs}(f), \mathrm{eval}(h) \rangle)^{q-1} - 1 \right) \neq 0$$

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$.

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$. What about the degree ?

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$. What about the degree ?

**Estimating** $|R|$:
Suppose $|\text{coeffs}(g)| \leq L$, $deg(g) = \text{poly}(n)$, and $|h| \leq k$.

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$. What about the degree ?

**Estimating** $|R|$:
Suppose $|\text{coeffs}(g)| \leq L$, $deg(g) = \text{poly}(n)$, and $|h| \leq k$.
Then $|\text{eval}(h)| \leq k^d$, $|g(h)| \approx L \cdot N \cdot k^d$

# Algebraic NOT - Integers

**Given**: Vector $\text{coeffs}(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$. What about the degree ?

**Estimating** $|R|$:
Suppose $|\text{coeffs}(g)| \leq L$, $deg(g) = \text{poly}(n)$, and $|h| \leq k$.
Then $|\text{eval}(h)| \leq k^d$, $|g(h)| \approx L \cdot N \cdot k^d$
For $d \sim n^3$, $N \sim \exp(n \log d)$ and $LNk^d = N^{\omega(1)}$.

# Algebraic NOT - Integers

**Given**: Vector coeffs$(g) \in \mathbb{C}^N$, point $h \in \mathbb{C}^n$

**Goal**: Output zero iff $g(h) \neq 0$, using a polynomial.

$R$ : set of non-zero values that a polynomial in $\mathcal{C}$ takes on $\mathcal{H}$.
Set $Q(y) = \prod_{r \in R}(y - r)$. What about the degree ?

**Estimating** $|R|$:
Suppose $|\,\mathrm{coeffs}(g)| \leq L$, $deg(g) = \mathrm{poly}(n)$, and $|h| \leq k$.
Then $|\,\mathrm{eval}(h)| \leq k^d$, $|g(h)| \approx L \cdot N \cdot k^d$
For $d \sim n^3$, $N \sim \exp(n \log d)$ and $LNk^d = N^{\omega(1)}$.

Cannot directly work with eval$(h)$.

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

**Chinese Remainder Theorem**

For an integer $-2^\ell \le M \le 2^\ell$,
if $M \bmod p_i = 0$ for *distinct* primes $p_1, \ldots, p_{2\ell}$; then $M = 0$.

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

**Chinese Remainder Theorem**
For an integer $-2^\ell \leq M \leq 2^\ell$,
if $M$ mod $p_i = 0$ for *distinct* primes $p_1, \ldots, p_{2\ell}$; then $M = 0$.

Set $\ell = \log(LNk^d) = \text{poly}(d, \log N)$. For primes $p_1, \ldots, p_\ell$,
let $\text{eval}_i(h) = \text{eval}(h) \bmod p_i$

$$= [m_1(h) \bmod p_i, \ldots, m_r(h) \bmod p_i] \in \mathbb{C}^N$$

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

**Chinese Remainder Theorem**
For an integer $-2^{\ell} \leq M \leq 2^{\ell}$,
if $M \bmod p_i = 0$ for *distinct* primes $p_1, \ldots, p_{2\ell}$; then $M = 0$.

Set $\ell = \log(LNk^d) = \mathrm{poly}(d, \log N)$. For primes $p_1, \ldots, p_{\ell}$,
let $\mathrm{eval}_i(h) = \mathrm{eval}(h) \bmod p_i$
$$= [m_1(h) \bmod p_i, \ldots, m_r(h) \bmod p_i] \in \mathbb{C}^N$$
$|\mathrm{eval}_i(h)| = \mathrm{poly}(\ell) = \mathrm{poly}(d, \log N)$.

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

**Chinese Remainder Theorem**
For an integer $-2^\ell \leq M \leq 2^\ell$,
if $M$ mod $p_i = 0$ for *distinct* primes $p_1, \ldots, p_{2\ell}$; then $M = 0$.

Set $\ell = \log(LNk^d) = \mathrm{poly}(d, \log N)$. For primes $p_1, \ldots, p_\ell$,
let $\mathrm{eval}_i(h) = \mathrm{eval}(h) \bmod p_i$
$= [m_1(h) \bmod p_i, \ldots, m_r(h) \bmod p_i] \in \mathbb{C}^N$
$|\,\mathrm{eval}_i(h)| = \mathrm{poly}(\ell) = \mathrm{poly}(d, \log N)$.

For $|\,\mathrm{coeffs}(g)| \leq L$,
$|\langle \mathrm{coeffs}(g), \mathrm{eval}_i(h)\rangle| \leq L \cdot N \cdot \mathrm{poly}(\ell) = \mathrm{poly}(N, L, d) = B$.

# Algebraic NOT - Integers

**Goal**: Check if $g(h) = 0$ using a lower degree polynomial.

**Chinese Remainder Theorem**
For an integer $-2^\ell \leq M \leq 2^\ell$,
if $M \bmod p_i = 0$ for *distinct* primes $p_1, \ldots, p_{2\ell}$; then $M = 0$.

Set $\ell = \log(LNk^d) = \mathrm{poly}(d, \log N)$. For primes $p_1, \ldots, p_\ell$,
let $\mathrm{eval}_i(h) = \mathrm{eval}(h) \bmod p_i$
$= [m_1(h) \bmod p_i, \ldots, m_r(h) \bmod p_i] \in \mathbb{C}^N$
$|\,\mathrm{eval}_i(h)| = \mathrm{poly}(\ell) = \mathrm{poly}(d, \log N)$.

For $|\,\mathrm{coeffs}(g)| \leq L$,
$|\langle \mathrm{coeffs}(g), \mathrm{eval}_i(h)\rangle| \leq L \cdot N \cdot \mathrm{poly}(\ell) = \mathrm{poly}(N, L, d) = B$.

**Note**: Can "hardwire" $\mathrm{eval}_i(h)$ for all $i \in [\ell]$ and $h \in \mathcal{H}$.

# Algebraic NOT - Integers

$$g(h) \neq 0 \iff \exists i \in [\ell] \quad s.t \quad (p_i \nmid \langle \mathsf{coeffs}(g), \mathsf{eval}_i(h) \rangle)$$

# Algebraic NOT - Integers

$$g(h) \neq 0 \iff \exists i \in [\ell] \quad s.t \quad (p_i \nmid \langle \mathsf{coeffs}(g), \mathsf{eval}_i(h) \rangle)$$

$$g(h) \neq 0 \iff \exists i \in [\ell] \quad s.t \quad \prod_{\substack{-B \leq a \leq B \\ p_i \nmid a}} (\langle \mathsf{coeffs}(g), \mathsf{eval}_i(h) \rangle - a) = 0$$

# Algebraic NOT - Integers

$$g(h) \neq 0 \iff \exists i \in [\ell] \quad s.t \quad (p_i \nmid \langle \mathrm{coeffs}(g), \mathrm{eval}_i(h) \rangle)$$

$$g(h) \neq 0 \iff \exists i \in [\ell] \quad s.t \quad \prod_{\substack{-B \leq a \leq B \\ p_i \nmid a}} (\langle \mathrm{coeffs}(g), \mathrm{eval}_i(h) \rangle - a) = 0$$

$$g(h) \neq 0 \iff \prod_{i \in [\ell]} \prod_{\substack{-B \leq a \leq B \\ p_i \nmid a}} (\langle \mathrm{coeffs}(g), \mathrm{eval}_i(h) \rangle - a) = 0$$

# Algebraic NOT - Integers

For $B = \text{poly}(L, N, d) = \text{poly}(N)$.

# Algebraic NOT - Integers

For $B = \text{poly}(L, N, d) = \text{poly}(N)$.

Equation for $\text{VP}'_{\mathbb{C}}$

$$P(\text{coeffs}(g)) \approx \prod_{h \in \mathcal{H}} \prod_{i \in [\ell]} \prod_{\substack{-B \le a \le B \\ p_i \nmid a}} (\langle \text{coeffs}(g), \text{eval}_i(h) \rangle - a)$$

$\text{Deg}(P) \le |\mathcal{H}| \, \text{poly}(n) \, \text{poly}(N) \le \text{poly}(N)$

$\text{Size}(P) \le \text{poly}(N)$.

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

Not enough: Want a solution with small integer coordinates.

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

Not enough: Want a solution with small integer coordinates.

**Siegel :** There exists such a solution!

# Integers: a hard polynomial

**Want**: $f$ with with small coefficients such that $\forall h \in \mathcal{H}$, $f(h) = 0$.

**Linear system in the coefficients of** $f$: $\forall h \in \mathcal{H}, f(h) = 0$

Many more variables than constraints, so there is a non-zero solution.

Not enough: Want a solution with small integer coordinates.

**Siegel :** There exists such a solution!

This ensures non-triviality of the equations obtained earlier.

# Results for VP

**Theorem** (Equations for $\mathrm{VP}'_{\mathbb{C}}$)

For $n$,$d$ and $N = \binom{n+d}{n}$,

There exists a nonzero $P(Z_1, \dots, Z_N) \in \mathrm{VP}(N)$ such that

for all $f \in \mathrm{VP}_{\mathbb{C}}(n, d)$ with coefficients in $\{-N, \dots, N\}$,

$P(\text{coeffs}(f)) = 0$.

Moreover, there is a $g$ with small coefficients such that
$P(\text{coeffs}(g)) = 0$.

# Results for VNP

**Theorem** (Equations for $\mathsf{VNP}'_{\mathbb{C}}$)

For $n$,$d$ and $N = \binom{n+d}{n}$,
There exists a nonzero $Q(Z_1, \ldots, Z_N) \in \mathsf{VP}(N)$ such that
for all $f \in \mathsf{VNP}_{\mathbb{C}}(n, d)$ with coefficients in $\{-N, \ldots, N\}$,
$Q(\text{coeffs}(f)) = 0$.

Moreover, there is a $g$ with small coefficients such that
$P(\text{coeffs}(g)) = 0$.

# To summarize

- Efficiently constructible equations exist for polynomials with "small" coefficients, in both VP and VNP.

# To summarize

- Efficiently constructible equations exist for polynomials with "small" coefficients, in both VP and VNP.
- The restriction is only on the polynomials, circuits can use any constants. Well-studied natural polynomials have small coefficients.

  e.g. Determinant, Permanent, ...

# To summarize

- Efficiently constructible equations exist for polynomials with "small" coefficients, in both VP and VNP.
- The restriction is only on the polynomials, circuits can use any constants. Well-studied natural polynomials have small coefficients.
  e.g. Determinant, Permanent, ...
- We can still hope to prove lower bounds for these polynomial families via constructible equations

# To summarize

- Efficiently constructible equations exist for polynomials with "small" coefficients, in both VP and VNP.
- The restriction is only on the polynomials, circuits can use any constants. Well-studied natural polynomials have small coefficients.

  e.g. Determinant, Permanent, ...
- We can still hope to prove lower bounds for these polynomial families via constructible equations, but cannot guarantee the *largeness* criterion.

# Questions

- Does all of VP have efficiently constructible equations?

# Questions

- Does all of VP have efficiently constructible equations?
  - Unlikely that out proof technique extends.

# Questions

- Does all of VP have efficiently constructible equations?
  - Unlikely that out proof technique extends.
  - How about Constant free versions of VP and VNP.

# Questions

- Does all of VP have efficiently constructible equations?
  - Unlikely that out proof technique extends.
  - How about Constant free versions of VP and VNP.

- How about seemingly simpler models...formulas/constant depth circuits?

# Questions

- Does all of VP have efficiently constructible equations?
  - Unlikely that out proof technique extends.
  - How about Constant free versions of VP and VNP.
- How about seemingly simpler models...formulas/constant depth circuits?
- Limitations on what can be proved via algebraically natural proofs ?

# Thanks!