The quest for a polynomial that is hard to compute

Neeraj Kayal

Microsoft Research
Polynomials

\[ f(x) \in F[x_1, x_2, \ldots, x_n] \text{ of degree } d. \]

*Typically:* \( F = \mathbb{C}, n = d^2 \)
Polynomials

\[ f(x_1, x_2, \ldots, x_n) \in \mathbb{F}[x_1, x_2, \ldots, x_n] \text{ of degree } d. \]

**Typically:** \( \mathbb{F} = \mathbb{C}, n = d^2 \)

**Question:** How many steps required to compute \( f(x) \)?
Polynomials

$f(x_1, x_2, ..., x_n) \in \mathbb{F}[x_1, x_2, ..., x_n]$ of degree $d$.

**Typically:** $\mathbb{F} = \mathbb{C}$, $n = d^2$

**Question:** How many steps required to compute $f(x)$?
Arithmetic Circuits

\[ x_1 \quad x_2 \quad \ldots \quad x_n \]
Arithmetic Circuits

\[ x_1 + x_2, \quad x_1 - x_2, \quad \ldots \]

\[ x_1, \quad x_2, \quad \ldots \quad x_n \]
Arithmetic Circuits

\[ ax_1 + bx_2 \]

\[ + \quad + \quad \cdots \quad + \]

\[ x_1 \quad x_2 \quad \cdots \quad x_n \]

\[ x_1 - x_2 \quad -1 \]
Arithmetic Circuits
Arithmetic Circuits

- $x_1$
- $x_2$
- $\ldots$
- $x_n$

Diagram of arithmetic operations with multiplications and additions.
Arithmetic Circuits

\[ x_1 \times (x_2 + \ldots + x_n) - 1 \]
\[ f(x) = \sum_{i=1}^{n} x_i \cdot x_i \cdot \prod_{i=1}^{n-1} x_i \cdot x_i - 1 \]

\[ \#Steps = \#Edges \]
\[ f(x) = \sum_{i=1}^{n} \left( x_i \prod_{j=1}^{n} x_j \right) \]
Fact: Most polynomials are extremely hard to compute.

#Steps ≈ Number of possible monomials
• **Fact:** Most polynomials are *extremely* hard to compute.

\[
\#\text{Steps} \geq \binom{n+d}{d}^{\Omega(1)}
\]
• **Fact:** Most polynomials are *extremely* hard to compute.

\[
\text{#Steps} \geq \binom{n+d}{d}^\Omega(1)
\]

• **Want:** An *explicit* polynomial that is *moderately* hard to compute.

\[
\text{#Steps} \geq (n \cdot \log d)^\omega(1)
\]
Explicit Polynomials : examples
MAJORITY Polynomial

\[ x = (x_1, x_2, ..., x_n) \]

\[ \text{MAJORITY}_n(x) = \sum_{S \in \binom{\{n\}}{n/2}} \prod_{i \in S} x_i \]
\[ x = (x_1, x_2, \ldots, x_n) \]

\[
\text{MAJORITY}_n(x) = \sum_{S \in \binom{[n]}{n/2}} \prod_{i \in S} x_i
\]

\[ \text{Fact:} \text{ For } x \in \{0,1\}^n \]

\[ \text{MAJORITY}_n = \begin{cases} 
0 & \text{if } \text{wt}(x) < n/2 \\
> 0 & \text{otherwise}
\end{cases} \]
CLIQUE Polynomial

\[ x = (x_{12}, x_{13}, \ldots, x_{(m-1)m}) \]

\[ n = \binom{m}{2} \]
**CLIQUE Polynomial**

\[ x = (x_{12}, x_{13}, \ldots, x_{(m-1)m}) \]

\[ n = \binom{m}{2}, \quad d = \binom{m/2}{2} \]

\[
\text{CLIQUE}_n(x) = \sum_{S \in \binom{[m]}{m/2}} \prod_{\{i,j\} \in \binom{S}{2}} x_{ij}
\]
\[ x = (x_{12}, x_{13}, \ldots, x_{(m-1)m}) \]

\[
\text{CLIQUE}_n(x) = \sum_{S \in \binom{[m]}{m/2}} \prod_{\{i,j\} \in \binom{S}{2}} x_{ij}
\]

- **Fact:** For \( x \in \{0,1\}^{\binom{m}{2}} \)

\[
\text{CLIQUE}_n = \begin{cases} 
0 & \text{if } G_x \text{ has no } \binom{m}{2} - \text{clique} \\
> 0 & \text{otherwise}
\end{cases}
\]
DETERMINANT Polynomial

\[ n = d^2, \quad x = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & \cdots & x_{dd} \end{pmatrix} \]

\[ \text{DET}_n(x) = \sum_{\sigma \in \text{Sym}(d)} \text{sgn}(\sigma) \cdot \prod_{i \in [n]} x_i \, \sigma(i) \]
PERMANENT Polynomial

\[ n = d^2, \quad x = \begin{pmatrix}
   x_{11} & x_{12} & \cdots & x_{1d} \\
   x_{21} & x_{22} & \cdots & x_{2d} \\
   \vdots & \vdots & \ddots & \vdots \\
   x_{d1} & x_{d2} & \cdots & x_{dd}
\end{pmatrix} \]

\[ \text{PERM}_n(x) = \sum_{\sigma \in \text{Sym}(d)} \prod_{i \in [n]} x_{i \sigma(i)} \]
• **Fact 1:** MAJORITY$_n$ is easy.

• **Fact 2:** DET$_n$ is easy.
• **Fact 1:** $\text{MAJORITY}_n$ is easy.

• **Fact 2:** $\text{DET}_n$ is easy.

• **Want to prove:** $\text{PERM}_n / \text{CLIQUE}_n /$ some explicit polynomial is extremely moderately hard
• **Fact 1:** $\text{MAJORITY}_n$ is easy.

• **Fact 2:** $\text{DET}_n$ is easy.

• **Want to prove:** $\text{PERM}_n / \text{CLIQUE}_n /$ some explicit polynomial is extremely moderately hard

**VP vs VNP**
Parallelizing Computation
Can computation be efficiently parallelized?

**Question:** How efficiently can we simulate circuits of size $s$ by circuits of depth $\Delta$?
Can computation be efficiently parallelized?

Example: Iterated Matrix Multiplication

- $X_1, X_2, \ldots, X_d$ are $n \times n$ matrices. Define
  \[
  \text{IMM}_{n,d} = \text{Tr}(X_1 \cdot X_2 \cdot \ldots \cdot X_d)
  \]

- There is a circuit of size $s = d \cdot n^\omega$

**Question:** What’s the smallest circuit of depth $\Delta$ computing $\text{IMM}_{n,d}$?
Example: Iterated Matrix Multiplication

\[ \sum \text{IMM}_{n,d} (X_1 \cdot X_2 \cdots X_d) \]
Example: Iterated Matrix Multiplication

\[ \text{IMM}_{n,d} \]

\[ \sum \]

\[ (X_1 \cdot X_2 \cdot \ldots \cdot X_d) \]

\[ \ast \]

\[ \Delta\text{-depth circuit} \]
Example: Iterated Matrix Multiplication

\[ \text{IMM}_{n,d} \]

\[ \sum \]

\[(X_1 \cdot X_2 \cdots \cdot X_d)\]

\[ \ast \]

\[ X_1 \cdot X_2 \cdots \cdot X_t \]

\[ \ast \]

\[ X_1 \quad X_2 \quad \ldots \quad X_t \]

\[ \ast \]

\[ X_{d-t+1} \cdot \cdots \cdot X_d \]

\[ \ast \]

\[ X_{d-t+1} \quad \ldots \quad X_d \]

\[ \Delta\text{-depth circuit} \]

\[ t = d^{2/\Delta} \]

\[ \text{size} = n^{O(d^{2/\Delta})} \]
Can computation be efficiently parallelized?

**Question:** How efficiently can we simulate circuits of size $s$ by circuits of depth $\Delta$?

**Theorem (Hya79, VSBR83, AV08, Koi12, Tav13):** Any circuit of size $s$ and degree $d$ can be simulated by a homogeneous and regular $\Delta$-depth circuit of size $s^{O(d^2/\Delta)}$. 
A potential approach
Proof Strategy?

- $f_n(x)$ is an explicit polynomial and we want to show that $f$ has no poly(n) sized circuit.
Proof Strategy?

- Suppose $\#\text{Steps}(f_n) = n^{o(1)}$. 
Proof Strategy?

- Suppose \(#\text{Steps}(f_n) = n^O(1)\). Then:
  \[ f_n = T_1 + T_2 + \ldots + T_s, \]
  where each \(T_i\) is a product of low-degree polynomials and \(s\) is not too large.
Proof Strategy?

- Suppose \( \text{Steps}(f_n) = n^{O(1)} \). Then:
  \[
  f_n = T_1 + T_2 + ... + T_s,
  \]
  where each \( T_i \) is a product of \( O(\sqrt{d}) \)-degree polynomials and \( s \) is \( n^{O(\sqrt{d})} \).

- Show that this is impossible.
Proof Strategy?

- Suppose \( \#\text{Steps}(f_n) = n^{O(1)} \). Then:
  \[
  f_n = T_1 + T_2 + \ldots + T_s,
  \]
  where each \( T_i \) is a product of \( O(\sqrt{d}) \)-degree polynomials and \( s \) is \( n^{O(\sqrt{d})} \).

**Theorem (K-Saha-Saptharishi 14):** There exists an explicit \( \{f_n : n \geq 1\} \) with \( n = d^2 \) such that
\[
s \geq n^{\Omega(\sqrt{d})}.
\]
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of low-degree polynomials.

- Find a geometric property GP of $V(T_i)$.

- Associate a matrix $M(g)$ to every polynomial $g$ such that:
  1. rank($M(T_i)$) is small, and
  2. Linearity: $M(\alpha \cdot g + \beta \cdot h) = \alpha \cdot M(g) + \beta \cdot M(h)$,
  3. rank($M(f_n)$) is large.
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of low-degree polynomials.

- Find a geometric property GP of $V(T_i)$.

- Express the property GP in terms of rank of a big matrix $M$:
  1. if $T$ has the property than rank($M(T)$) is small, and
  2. Linearity: $M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h)$,

- Show that rank($M(f_n)$) is large.
Define: \( V(f) = \{ a \in \mathbb{C}^n : f(a) = 0 \} \)
Proof Strategy

- Suppose \( f_n = T_1 + T_2 + \ldots + T_s \), where each \( T_i \) is a product of low-degree polynomials.

- Find a geometric property GP of \( V(T_i) \).

- Express the property GP in terms of rank of a big matrix \( M \):
  (1) if \( T \) has the property than rank(\( M(T) \)) is small, and
  (2) **Linearity**: \( M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h) \),

- Show that rank(\( M(f_n) \)) is large.
Lower Bounding rank of large matrices

- If a matrix $M(f_n)$ has a large upper triangular submatrix, then it has large rank.

- (Alon): If the columns of $M(f_n)$ are almost orthogonal then $M(f_n)$ has large rank.
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of low-degree polynomials.

- Find a geometric property GP of $V(T_i)$.

- Express the property GP in terms of rank of a big matrix $M$:
  1. if $T$ has the property than rank($M(T)$) is small, and
  2. Linearity: $M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h)$,

- Show that rank($M(f_n)$) is large.
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of low-degree polynomials.

- Find a geometric property GP of $V(T_i)$.

- Express the property GP in terms of rank of a big matrix $M$:
  1. If $T$ has the property than rank($M(T)$) is small, and
  2. Linearity: $M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h)$,

- Show that rank($M(f_n)$) is large.
Finding a geometric property $GP$ of $T$

$T$ is a product of low degree polynomials

$V(T)$ is a union of low-degree hypersurfaces
Finding a geometric property GP of T

$T$ is a product of low degree polynomials

$V(T)$ is a union of low-degree hypersurfaces

$V(T)$ has lots of high-order singularities

$V(\partial = \kappa T)$ has lots of points
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of low-degree polynomials.

- Find a geometric property GP of $V(T_i)$.

- Express the property GP in terms of rank of a big matrix $M$:
  1. if $T$ has the property than rank($M(T)$) is small, and
  2. Linearity: $M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h)$,

- Show that rank($M(f_n)$) is large.
Proof Strategy

- Suppose $f_n = T_1 + T_2 + \ldots + T_s$, where each $T_i$ is a product of *low-degree* polynomials.

- Find a geometric property GP of $V(T_i)$.

- Express the property GP in terms of rank of a *big* matrix $M$:
  - (1) if $T$ has the property than rank($M(T)$) is *small*, and
  - (2) *Linearity*: $M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h)$,

- Show that rank($M(f_n)$) is *large*. 
Expressing largeness of a variety in terms of rank

\[ V = V(f_1, f_2, \ldots, f_m) \text{ is a variety.} \]

Let \( \mathcal{G}_\ell \) = set of degree-\( \ell \) polynomials.
Expressing largeness of a variety in terms of rank

\[ V = V(f_1, f_2, ..., f_m) \] is a variety.

Let \( G_\ell = \) set of degree-\( \ell \) polynomials. Let \( G_\ell(V) = \) set of degree-\( \ell \) polynomials which vanish at every point of \( V \).

**Hilbert’s Theorem (Informal):** If \( V \) is “large” then \( G_\ell(V) \) has small dimension.
Expressing largeness of a variety in terms of rank

\[ V = V(f_1, f_2, \ldots, f_m) \] is a variety.

Let \( \mathcal{G}_\ell = \) set of degree-\( \ell \) polynomials. Let \( \mathcal{G}_\ell(V) = \) set of degree-\( \ell \) polynomials which vanish at every point of \( V \).

**Hilbert’s Theorem (Informal):** If \( V \) is “large” then \( \mathcal{G}_\ell(V) \) has small dimension.

Let \( \mathcal{I}_\ell(V) = \{ (a_1 \cdot f_1 + a_2 \cdot f_2 + \ldots + a_m \cdot f_m) \) of \( \deg \leq \ell \} \subseteq \mathcal{G}_\ell(V) \).

**Hilbert’s Theorem (Formal):** If \( V \) has dimension \( r \) then \( \mathcal{I}_\ell(V) \) has asymptotic dimension \( \binom{n + \ell}{n} - \Theta(\ell^r) \).
Proof Strategy

- Suppose \( f_n = T_1 + T_2 + \ldots + T_s \), where each \( T_i \) is a product of \textit{low-degree} polynomials.

- Find a geometric property \( GP \) of \( V(T_i) \).

- Express the property \( GP \) in terms of rank of a \textit{big} matrix \( M \):
  1. if \( T \) has the property than \( \text{rank}(M(T)) \) is \textit{small}, and
  2. Linearity: \( M(\alpha \cdot g + \beta \cdot h) = \alpha M(g) + \beta M(h) \),

- Show that \( \text{rank}(M(f_n)) \) is \textit{large}. 
Summary

\[ f_n = T_1 + T_2 + \ldots + T_s, \] where each \( T_i \) is a product of \( O(\sqrt{d}) \)-degree homogeneous polynomials

**Theorem (K-Saha-Saptharishi 14):** There exists an explicit \( \{f_n : n \geq 1\} \) with \( n = d^2 \) such that
\[ s \geq n^{\Omega(\sqrt{d})}. \]

**Theorem (FLMS14, KS15):** For \( \text{IMM}_{n,d} \) with \( n = d^{10}, \)
\[ s \geq n^{\Omega(\sqrt{d})}. \]
**Conclusion/Open Questions**

- Most lower bounds we have are instantiations of this strategy. Can we solve VP vs VNP using it?

- Any circuit of size $s$ and degree $d$ can be simulated by a homogeneous and regular $\Delta$-depth circuit of size $s^{O(d^{2/\Delta})}$. **Is this optimal?**
**Conclusion/Open Questions**

- Most lower bounds we have are instantiations of this strategy. Can we solve VP vs VNP using it?

- Any circuit of size $s$ and degree $d$ can be simulated by a homogeneous and regular $\Delta$-depth circuit of size $s^{O(d^{2/\Delta})}$. **Is this optimal?**

- Any circuit of size $s$ and degree $d$ can be simulated by a $\Delta$-depth circuit of size $s^{O(d^{1/\Delta})}$. 

**Conclusion/Open Questions**

- Most lower bounds we have are instantiations of this strategy. Can we solve VP vs VNP using it?

- Any circuit of size $s$ and degree $d$ can be simulated by a homogeneous and regular $\Delta$-depth circuit of size $s^{O(d^{2/\Delta})}$. Is this optimal?

- Any circuit of size $s$ and degree $d$ can be simulated by a $\Delta$-depth circuit of size $s^{O(d^{1/\Delta})}$. Is this optimal?
Conclusion/Open Questions

- Most lower bounds we have are instantiations of this strategy. Can we solve VP vs VNP using it?

- Any circuit of size $s$ and degree $d$ can be simulated by a $\Delta$-depth circuit of size $s^{O(d^{1/\Delta})}$. Is this optimal?

- Exploit other Geometric Properties? number of irreducible components, presence of high-dimensional affine subspaces, ...