Lower Bounds and Separations for Torus Polynomials

V. Krishan [1] Sundar Vishwanathan [2]

[1]TCS, IMSc Chennai

[2]CSE, IIT Bombay

Main Goal

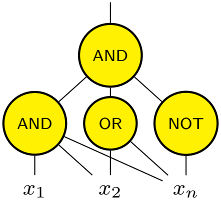
Goal

Resolve Barrington's conjecture on constant-depth circuits.

Main Goal

Goal

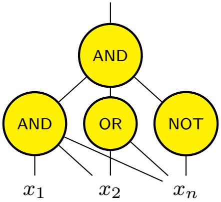
Resolve Barrington's conjecture on constant-depth circuits.



Main Goal

Goal

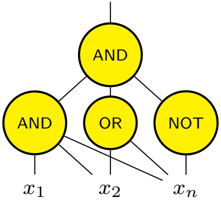
Resolve Barrington's conjecture on constant-depth circuits.



Computational resources:

Goal

Resolve Barrington's conjecture on constant-depth circuits.



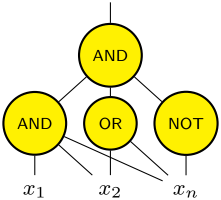
Computational resources:

- Size: number of gates.

Main Goal

Goal

Resolve Barrington's conjecture on constant-depth circuits.



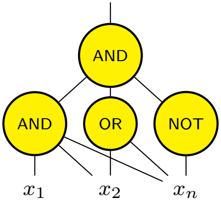
Computational resources:

- Size: number of gates.
- Depth: number of layers.

Main Goal

Goal

Resolve Barrington's conjecture on constant-depth circuits.



Computational resources:

- Size: number of gates.
- Depth: number of layers.

AC^0: Constant-depth polynomial size with AND, OR, NOT gates.

The Landscape of Circuit Complexity

$$\begin{tikzcd} \mathsf{NP} \\ \mathsf{P} \arrow[u] \end{tikzcd}$$

NP

P

AC^0

The Landscape of Circuit Complexity

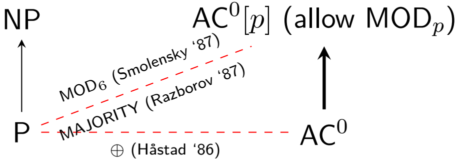
- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$

The Landscape of Circuit Complexity

$$\begin{tikzcd} \mathsf{NP} & \mathsf{AC}^0[p] \text{ (allow } \mathsf{MOD}_p\text{)} \\ \mathsf{P} \arrow[u] \arrow[r, dashed, no head, "\oplus \text{ (Håstad `86)}"'] & \mathsf{AC}^0 \arrow[u] \end{tikzcd}$$

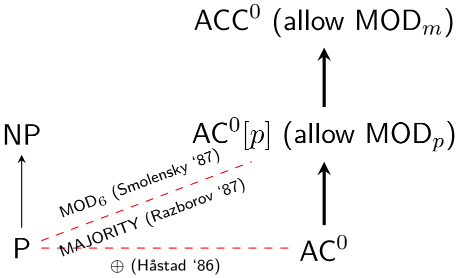
- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$
- $\mathsf{MOD}_p(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i$ is divisible by p.

The Landscape of Circuit Complexity



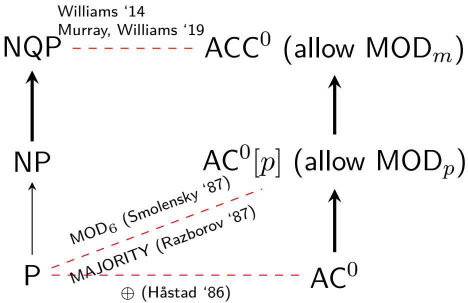
- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$
- $\mathrm{MOD}_p(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i$ is divisible by p.
- $\mathrm{MAJORITY}(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i \geq \frac{n}{2}$.

The Landscape of Circuit Complexity



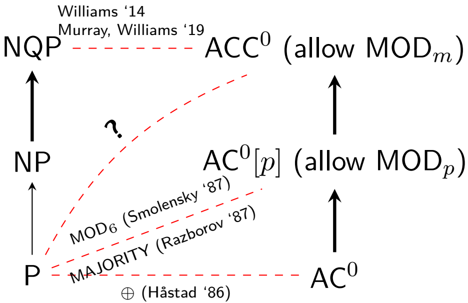
- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$
- MOD$_p(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i$ is divisible by p.
- MAJORITY$(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i \geq \frac{n}{2}$.

The Landscape of Circuit Complexity



- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$
- MOD$_p(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i$ is divisible by p.
- MAJORITY$(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i \geq \frac{n}{2}$.

The Landscape of Circuit Complexity



- $\oplus(x_1, \ldots, x_n) = \sum x_i \mod 2$
- MOD$_p(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i$ is divisible by p.
- MAJORITY$(x_1, \ldots, x_n) = 1$ if and only if $\sum x_i \geq \frac{n}{2}$.

Barrington's Conjecture

Definition (ACC^0)

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.
- Constant depth.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.
- Constant depth.

Conjecture (Barrington '89)

MAJORITY $\notin$ ACC^0.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.
- Constant depth.

Conjecture (Barrington '89)

MAJORITY $\notin \mathsf{ACC}^0$.

Previous techniques seemingly do not apply:

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD$_m$ gates.
- Polynomial size.
- Constant depth.

Conjecture (Barrington '89)

MAJORITY $\notin$ ACC^0.

Previous techniques seemingly do not apply:

- Håstad's technique.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.
- Constant depth.

Conjecture (Barrington '89)

MAJORITY $\notin$ ACC^0.

Previous techniques seemingly do not apply:

- Håstad's technique.
- Razborov-Smolensky method.

Barrington's Conjecture

Definition (ACC^0)

- AND, OR, NOT and MOD_m gates.
- Polynomial size.
- Constant depth.

Conjecture (Barrington '89)

MAJORITY $\notin$ ACC^0.

Previous techniques seemingly do not apply:

- Håstad's technique.
- Razborov-Smolensky method.
- Murray and Williams' framework.

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if,

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if, for each a,

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if, for each a, there is some $Z(a) \in \mathbb{Z}$,

Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if, for each a, there is some $Z(a) \in \mathbb{Z}$, $P(a)$ is within ε of $Z(a) + \frac{f(a)}{2}$.

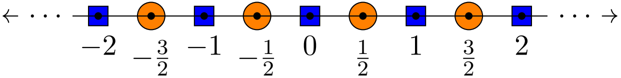
Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if, for each a, there is some $Z(a) \in \mathbb{Z}$, $P(a)$ is within ε of $Z(a) + \frac{f(a)}{2}$.



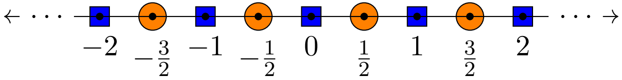
Distinguisher: Torus Polynomials

General Approach: Find a distinguisher.

- Define $\mu : f \to \mathbb{R}$.
- Calculate $\mu(\mathcal{C}) = \{\mu(C) : C \in \mathcal{C}\}$.
- Prove $\mu(g) \notin \mu(\mathcal{C})$, hence $g \notin \mathcal{C}$.

Definition (Torus Polynomial Approximation (BHLR '19))

P is a *torus polynomial* ε-approximating f if, for each a, there is some $Z(a) \in \mathbb{Z}$, $P(a)$ is within ε of $Z(a) + \frac{f(a)}{2}$.



Trivial upper bound: degree n for any f.

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, *and* $f \in \mathsf{ACC}^0$,

Why Torus Polynomials

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$*, and* $f \in \mathsf{ACC}^0$*, there exists a polylog-degree torus polynomial* ε*-approximating* f*.*

Why Torus Polynomials

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, and $f \in \mathsf{ACC}^0$, there exists a polylog-degree torus polynomial ε-approximating f.

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, any symmetric torus polynomial ε-approximating MAJORITY *must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.*

Why Torus Polynomials

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, and $f \in \mathsf{ACC}^0$, there exists a polylog-degree torus polynomial ε-approximating f.

Theorem (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, any symmetric torus polynomial ε-approximating MAJORITY *must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.*

Conjecture (BHLR '19)

For $\varepsilon = \frac{1}{20n}$, any torus polynomial ε-approximating MAJORITY must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.

Linear Programming Approach

- Choose n, d, ε and f.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.
- For any $a \in \{0,1\}^n$, $P(a)$ is linear combination of coefficients.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.
- For any $a \in \{0,1\}^n$, $P(a)$ is linear combination of coefficients.
- We want for some $Z(a) \in \mathbb{Z}$:

$$Z(a) + \frac{f(a)}{2} - \varepsilon \leq P(a) \leq Z(a) + \frac{f(a)}{2} + \varepsilon$$

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.
- For any $a \in \{0,1\}^n$, $P(a)$ is linear combination of coefficients.
- We want for some $Z(a) \in \mathbb{Z}$:

$$Z(a) + \frac{f(a)}{2} - \varepsilon \leq P(a) \leq Z(a) + \frac{f(a)}{2} + \varepsilon$$

- For each $Z : \{0,1\}^n \to \mathbb{Z}$, obtain a linear program.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.
- For any $a \in \{0,1\}^n$, $P(a)$ is linear combination of coefficients.
- We want for some $Z(a) \in \mathbb{Z}$:

$$Z(a) + \frac{f(a)}{2} - \varepsilon \leq P(a) \leq Z(a) + \frac{f(a)}{2} + \varepsilon$$

- For each $Z : \{0,1\}^n \to \mathbb{Z}$, obtain a linear program.
- A torus polynomial exists iff some linear program is feasible.

Linear Programming Approach

- Choose n, d, ε and f.
- Goal: Any P that ε-approximates f has degree more than d.
- Treat coefficients as variables.
- For any $a \in \{0,1\}^n$, $P(a)$ is linear combination of coefficients.
- We want for some $Z(a) \in \mathbb{Z}$:

$$Z(a) + \frac{f(a)}{2} - \varepsilon \leq P(a) \leq Z(a) + \frac{f(a)}{2} + \varepsilon$$

- For each $Z : \{0,1\}^n \to \mathbb{Z}$, obtain a linear program.
- A torus polynomial exists iff some linear program is feasible.
- Lower bound iff programs are infeasible iff duals are feasible.

The Family of Duals

- For each Z, find $\gamma \in nullspace(M(n,d))$, such that :

$$\left|\left\langle Z + \frac{f}{2}, \gamma \right\rangle\right| > \varepsilon \|\gamma\|_1$$

The Family of Duals

- For each Z, find $\gamma \in nullspace(M(n,d))$, such that :

$$\left|\left\langle Z + \frac{f}{2}, \gamma \right\rangle\right| > \varepsilon \|\gamma\|_1$$

- $M(n,d)$ has evaluations of monomials with degree at most d.

The Family of Duals

- For each Z, find $\gamma \in nullspace(M(n,d))$, such that :

$$\left|\left\langle Z + \frac{f}{2}, \gamma \right\rangle\right| > \varepsilon \|\gamma\|_1$$

- $M(n,d)$ has evaluations of monomials with degree at most d.
- Extends the *method of dual polynomials* to torus polynomials.

The Family of Duals

- For each Z, find $\gamma \in nullspace(M(n,d))$, such that :

$$\left|\left\langle Z + \frac{f}{2}, \gamma \right\rangle\right| > \varepsilon \|\gamma\|_1$$

- $M(n,d)$ has evaluations of monomials with degree at most d.
- Extends the *method of dual polynomials* to torus polynomials.
- Allows for incremental progress.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.
  - Resolves all but a finite subfamily.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.
  - Resolves all but a finite subfamily.
- Fix $Z(a)$ up to $|a| \lesssim d^2$, other $Z(a)$ are uniquely determined.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.
  - Resolves all but a finite subfamily.
- Fix $Z(a)$ up to $|a| \lesssim d^2$, other $Z(a)$ are uniquely determined.
  - Reduces the degrees of freedom if $d \ll \sqrt{n}$.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.
  - Resolves all but a finite subfamily.
- Fix $Z(a)$ up to $|a| \lesssim d^2$, other $Z(a)$ are uniquely determined.
  - Reduces the degrees of freedom if $d \ll \sqrt{n}$.
- New nullspace vectors supported on a single Hamming layer.

Our Contribution

- $|Z(a)| \leq \varepsilon \cdot 2^{d+1} \cdot \binom{|a|}{d+1}$.
  - Resolves all but a finite subfamily.
- Fix $Z(a)$ up to $|a| \lesssim d^2$, other $Z(a)$ are uniquely determined.
  - Reduces the degrees of freedom if $d \ll \sqrt{n}$.
- New nullspace vectors supported on a single Hamming layer.
  - Asymmetric construction, unlike previously known.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.
- For $\varepsilon = \frac{1}{20n}$, any *symmetric* torus polynomial ε-approximating AND must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.
- For $\varepsilon = \frac{1}{20n}$, any *symmetric* torus polynomial ε-approximating AND must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.
  - Symmetric torus polynomials are weaker.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.
- For $\varepsilon = \frac{1}{20n}$, any *symmetric* torus polynomial ε-approximating AND must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.
  - Symmetric torus polynomials are weaker.
  - Extends MAJORITY lower bound from [BHLR '19].

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.
- For $\varepsilon = \frac{1}{20n}$, any *symmetric* torus polynomial ε-approximating AND must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.
  - Symmetric torus polynomials are weaker.
  - Extends MAJORITY lower bound from [BHLR '19].
- Error-degree trade-off for symmetric torus polynomials approximating MAJORITY.

Our Results

- $\Omega\left(\log\left(\frac{1}{\varepsilon}\right)\right)$-degree lower bound for torus polynomials ε-approximating AND.
  - No error-reduction if MAJORITY requires large degree.
- For $\varepsilon = \frac{1}{20n}$, any *symmetric* torus polynomial ε-approximating AND must have degree $\widetilde{\Omega}\left(\sqrt{n}\right)$.
  - Symmetric torus polynomials are weaker.
  - Extends MAJORITY lower bound from [BHLR '19].
- Error-degree trade-off for symmetric torus polynomials approximating MAJORITY.
  - Strengthens corresponding result from [BHLR '19].

Future Directions

- Continue the program to find feasible solutions for more Zs.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize "solved" Zs using known solutions.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize "solved" Zs using known solutions.
  - Find more possible solutions.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize "solved" Zs using known solutions.
  - Find more possible solutions.
- Bridge the lower-upper bound gap for AND.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize “solved” Zs using known solutions.
  - Find more possible solutions.
- Bridge the lower-upper bound gap for AND.
  - Current proof uses only one solution.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize “solved” Zs using known solutions.
  - Find more possible solutions.
- Bridge the lower-upper bound gap for AND.
  - Current proof uses only one solution.
  - Use multiple solutions for stronger lower bound.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize "solved" Zs using known solutions.
  - Find more possible solutions.
- Bridge the lower-upper bound gap for AND.
  - Current proof uses only one solution.
  - Use multiple solutions for stronger lower bound.
- Error-degree trade-off for symmetric torus polynomials approximating AND.

Future Directions

- Continue the program to find feasible solutions for more Zs.
  - Characterize "solved" Zs using known solutions.
  - Find more possible solutions.
- Bridge the lower-upper bound gap for AND.
  - Current proof uses only one solution.
  - Use multiple solutions for stronger lower bound.
- Error-degree trade-off for symmetric torus polynomials approximating AND.
  - Use lattice theory.

Thank you!