Super-quadratic Lower Bounds for Depth-2 Linear Threshold Circuits

Lijie Chen
(UC Berkeley & OpenAI)

Avishay Tal
(UC Berkeley)

Yichuan Wang
(UC Berkeley)

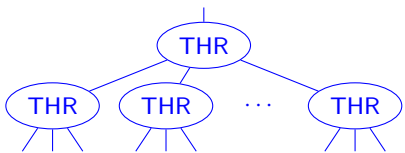
Apr 16, 2026

Today's Plan

- **1. Backgrounds in TC^0 circuit lower bounds**
- **2. Two main ideas behind our new lower bound:**
  - Random Restrictions in the Algorithmic Method
  - List-Approximation Polynomials

The Problem Setting - Threshold Circuits (TC^0)

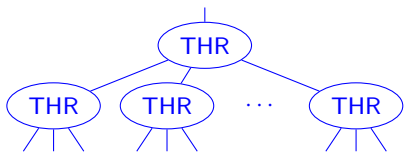
TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates



The Problem Setting - Threshold Circuits (TC^0)

TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates

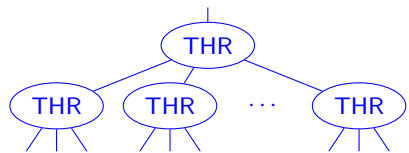
- MAJ (Majority Gate):
  - Outputs 1 iff there are more 1-s than 0-s in the input bits;
- THR (Real-Weighted Threshold Gate):
  - Parameters: weights $w_1, \cdots, w_n \in \mathbb{R}$ and threshold value $t \in \mathbb{R}$;
  - Input: $(x_1, \cdots, x_n) \in \{0,1\}^n$;
  - Output: $\mathbb{I}[w_1 x_1 + \cdots + w_n x_n \geq t]$.



The Problem Setting - Threshold Circuits (TC^0)

TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates

- MAJ (Majority Gate):
  - Outputs 1 iff there are more 1-s than 0-s in the input bits;
- THR (Real-Weighted Threshold Gate):
  - Parameters: weights $w_1, \cdots, w_n \in \mathbb{R}$ and threshold value $t \in \mathbb{R}$;
  - Input: $(x_1, \cdots, x_n) \in \{0,1\}^n$;
  - Output: $\mathbb{I}[w_1 x_1 + \cdots + w_n x_n \geq t]$.

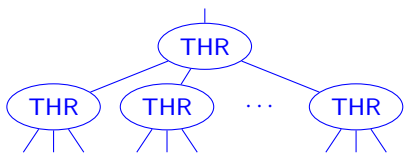


Fact: $\mathsf{THR} \subseteq \mathsf{MAJ} \circ \mathsf{MAJ}$

The Problem Setting - Threshold Circuits (TC^0)

TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates

- MAJ (Majority Gate):
  - Outputs 1 iff there are more 1-s than 0-s in the input bits;
- THR (Real-Weighted Threshold Gate):
  - Parameters: weights $w_1, \cdots, w_n \in \mathbb{R}$ and threshold value $t \in \mathbb{R}$;
  - Input: $(x_1, \cdots, x_n) \in \{0,1\}^n$;
  - Output: $\mathbb{I}[w_1 x_1 + \cdots + w_n x_n \geq t]$.

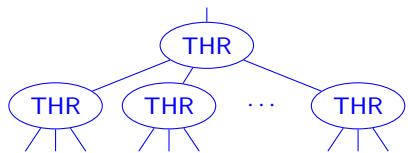


Fact: $\mathsf{THR} \subseteq \mathsf{MAJ} \circ \mathsf{MAJ}$, $\underbrace{\mathsf{THR} \circ \cdots \circ \mathsf{THR}}_{k \text{ Layers}} \subseteq \underbrace{\mathsf{MAJ} \circ \cdots \circ \mathsf{MAJ}}_{k+1 \text{ Layers}}$

The Problem Setting - Threshold Circuits (TC^0)

TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates

- MAJ (Majority Gate):
  - Outputs 1 iff there are more 1-s than 0-s in the input bits;
- THR (Real-Weighted Threshold Gate):
  - Parameters: weights $w_1, \cdots, w_n \in \mathbb{R}$ and threshold value $t \in \mathbb{R}$;
  - Input: $(x_1, \cdots, x_n) \in \{0,1\}^n$;
  - Output: $\mathbb{I}[w_1 x_1 + \cdots + w_n x_n \geq t]$.



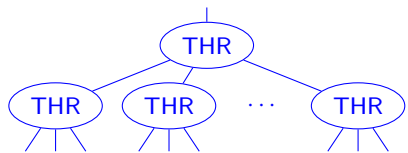
Fact: $\mathsf{THR} \subseteq \mathsf{MAJ} \circ \mathsf{MAJ}$, $\underbrace{\mathsf{THR} \circ \cdots \circ \mathsf{THR}}_{k \text{ Layers}} \subseteq \underbrace{\mathsf{MAJ} \circ \cdots \circ \mathsf{MAJ}}_{k+1 \text{ Layers}}$

$\mathsf{ACC}^0 \subseteq \mathsf{TC}^0 \subseteq \mathsf{NC}^1$

The Problem Setting - Threshold Circuits (TC^0)

TC^0: polynomial-size, **constant-depth** circuits with **Linear Threshold Gates** and free NOT gates

- MAJ (Majority Gate):
  - Outputs 1 iff there are more 1-s than 0-s in the input bits;
- THR (Real-Weighted Threshold Gate):
  - Parameters: weights $w_1, \cdots, w_n \in \mathbb{R}$ and threshold value $t \in \mathbb{R}$;
  - Input: $(x_1, \cdots, x_n) \in \{0,1\}^n$;
  - Output: $\mathbb{I}[w_1 x_1 + \cdots + w_n x_n \geq t]$.



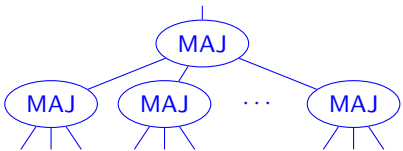
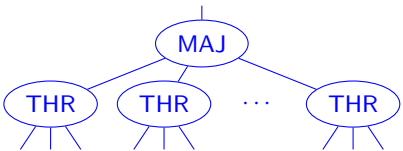
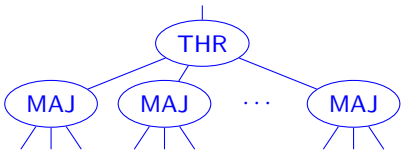
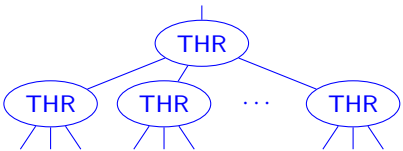
Fact: $\mathsf{THR} \subseteq \mathsf{MAJ} \circ \mathsf{MAJ}$, $\underbrace{\mathsf{THR} \circ \cdots \circ \mathsf{THR}}_{k \text{ Layers}} \subseteq \underbrace{\mathsf{MAJ} \circ \cdots \circ \mathsf{MAJ}}_{k+1 \text{ Layers}}$

$\mathsf{ACC}^0 \subseteq \mathsf{TC}^0 \subseteq \mathsf{NC}^1$

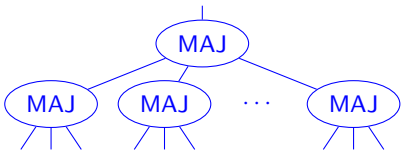
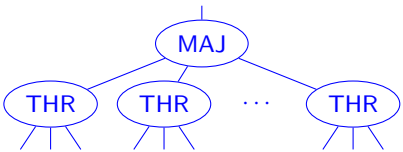
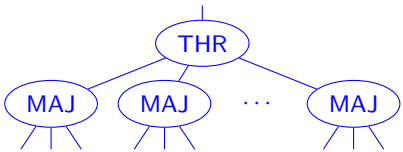
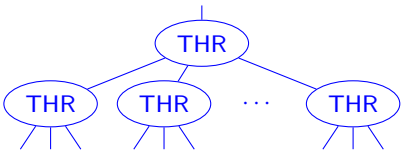
Big Open Problem: $\mathsf{NEXP} \overset{?}{\not\subseteq} \mathsf{TC}^0$, or even $\mathsf{E}^{\mathsf{NP}} \overset{?}{\not\subseteq} \mathsf{TC}^0$ (Although we believe $\mathsf{P} \not\subseteq \mathsf{TC}^0$)

Depth-2 Threshold Circuit Lower Bounds

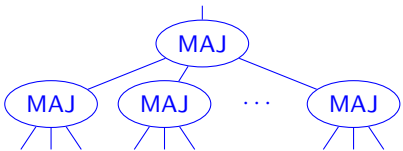
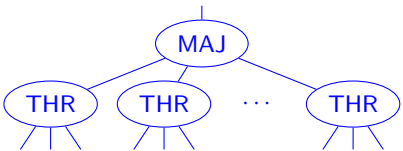
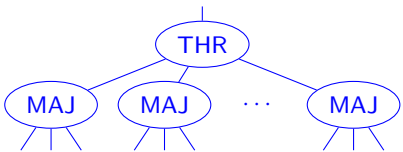
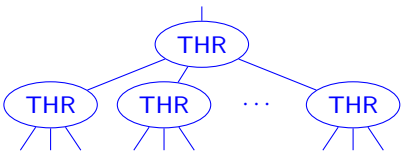
Depth-2 Threshold Circuit Lower Bounds

MAJ ∘ MAJ	MAJ ∘ THR
	
THR ∘ MAJ	**THR ∘ THR**
	

Depth-2 Threshold Circuit Lower Bounds

MAJ ∘ MAJ	MAJ ∘ THR
	
InnerProduct $\notin$ MAJ ∘ MAJ [HMPST' 93]	InnerProduct $\notin$ MAJ ∘ THR [Nisan' 94]
THR ∘ MAJ	**THR ∘ THR**
	
InnerProduct $\notin$ THR ∘ MAJ [Forster' 02]	

Depth-2 Threshold Circuit Lower Bounds

MAJ ∘ MAJ	MAJ ∘ THR
 InnerProduct $\notin$ MAJ ∘ MAJ [HMPST' 93]	 InnerProduct $\notin$ MAJ ∘ THR [Nisan' 94]
THR ∘ MAJ	**THR ∘ THR**
 InnerProduct $\notin$ THR ∘ MAJ [Forster' 02]	 Open: $\mathsf{E}^{\mathsf{NP}} \overset{?}{\not\subseteq}$ THR ∘ THR

Approaches Towards Circuit Lower Bounds

The Algorithmic Method: [Williams' 11]
Non-trivial Circuit Analysis algorithms
$\implies$ Circuit Lower Bounds

Approaches Towards Circuit Lower Bounds

The Algorithmic Method: [Williams' 11]
Non-trivial Circuit Analysis algorithms
$\implies$ Circuit Lower Bounds

- CAPP (Circuit Acceptance Probability Problem)
  - Input: A Circuit
  - Output: Estimate the fraction of inputs in $\{0,1\}^n$ that outputs 1
  - Error: $\pm o(1)$ (arbitrarily small constant)
- Non-trivial: runs in deterministic $2^n/n^{\omega(1)}$ time for n-input poly(n)-size circuits. (Trivial is $2^n \cdot \text{poly}(n)$.)

$$\mathsf{ACC}^0 - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$$

Approaches Towards Circuit Lower Bounds

The Algorithmic Method: [Williams' 11]
Non-trivial Circuit Analysis algorithms
$\implies$ Circuit Lower Bounds

- CAPP (Circuit Acceptance Probability Problem)
  - Input: A Circuit
  - Output: Estimate the fraction of inputs in $\{0,1\}^n$ that outputs 1
  - Error: $\pm o(1)$ (arbitrarily small constant)
- Non-trivial: runs in deterministic $2^n/n^{\omega(1)}$ time for n-input poly(n)-size circuits. (Trivial is $2^n \cdot \text{poly}(n)$.)

$$\mathsf{ACC}^0 - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$$
$$\oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{THR} \circ \mathsf{THR}$$

Approaches Towards Circuit Lower Bounds

The Algorithmic Method: [Williams' 11]
Non-trivial Circuit Analysis algorithms
$\implies$ Circuit Lower Bounds

- CAPP (Circuit Acceptance Probability Problem)
  - Input: A Circuit
  - Output: Estimate the fraction of inputs in $\{0,1\}^n$ that outputs 1
  - Error: $\pm o(1)$ (arbitrarily small constant)
- Non-trivial: runs in deterministic $2^n/n^{\omega(1)}$ time for n-input poly(n)-size circuits. (Trivial is $2^n \cdot \text{poly}(n)$.)

$$\mathsf{ACC}^0 - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$$
$$\oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{THR} \circ \mathsf{THR}$$

" $\implies$ "-s are already known, but we lack Circuit Analysis Algorithms !

Approaches Towards Circuit Lower Bounds

The Algorithmic Method: [Williams' 11]
Non-trivial Circuit Analysis algorithms
$\implies$ Circuit Lower Bounds

- CAPP (Circuit Acceptance Probability Problem)
  - Input: A Circuit
  - Output: Estimate the fraction of inputs in $\{0,1\}^n$ that outputs 1
  - Error: $\pm o(1)$ (arbitrarily small constant)
- Non-trivial: runs in deterministic $2^n/n^{\omega(1)}$ time for n-input poly(n)-size circuits. (Trivial is $2^n \cdot \text{poly}(n)$.)

$$\mathsf{ACC}^0 - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$$
$$\oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP} \implies \mathsf{NEXP} \not\subseteq \mathsf{THR} \circ \mathsf{THR}$$
$$n^\alpha\text{-size-}\oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP} \implies \mathsf{E}^{\mathsf{NP}} \not\subseteq n^\alpha\text{-size-}\mathsf{THR} \circ \mathsf{THR}$$

" $\implies$ "-s are already known, but we lack Circuit Analysis Algorithms !

Our Result

- Previous Results:

$\mathsf{P} \not\subseteq n^{1.5-\varepsilon}$-size-$\mathsf{THR} \circ \mathsf{THR}$ [Kane-Williams' 16]

$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2-\varepsilon}$-size-$\mathsf{THR} \circ \mathsf{THR}$ [Alman-Chan-Williams' 16] [Tamaki' 16]

Our Result

- Previous Results:

$$\mathsf{P} \not\subseteq n^{1.5-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR} \qquad \text{[Kane-Williams' 16]}$$
$$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR} \qquad \text{[Alman-Chan-Williams' 16] [Tamaki' 16]}$$

- **Our Result:**

$$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2.5-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR}$$

via designing an algorithm for

$$n^{2.5-\varepsilon}\text{-size-}\oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP}$$

Today's Plan

- 1. Backgrounds in TC^0 circuit lower bounds

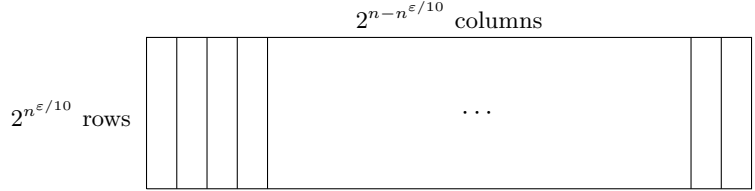
- **2. Two main ideas behind our new lower bound:**
  - Random Restrictions in the Algorithmic Method
  - List-Approximation Polynomials

Idea 1: Random Restrictions in the Algorithmic Method

- A standard step in designing circuit analysis algorithms: partition the input into two parts:

$$x \in \{0,1\}^n \rightarrow \boxed{|y| = n^{\varepsilon/10}} \boxed{|z| = n - n^{\varepsilon/10}}$$

- Arrange the circuit's truth table (2^n entries) into a $2^{n^{\varepsilon/10}} \times 2^{n-n^{\varepsilon/10}}$ table:



- The algorithm actually computes CAPP for all columns (and then add them together).

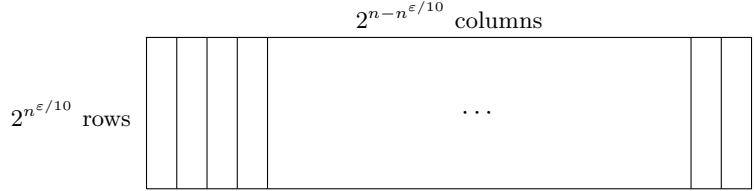
Idea 1: Random Restrictions in the Algorithmic Method

- A standard step in designing circuit analysis algorithms: partition the input into two parts:

$$x \in \{0,1\}^n \rightarrow \boxed{|y| = n^{\varepsilon/10}} \boxed{|z| = n - n^{\varepsilon/10}}$$

free variables ⋆ fixed variables 0/1

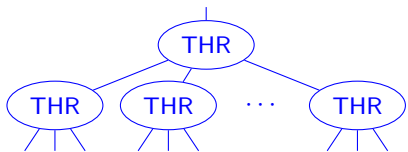
- Arrange the circuit's truth table (2^n entries) into a $2^{n^{\varepsilon/10}} \times 2^{n-n^{\varepsilon/10}}$ table:



- The algorithm actually computes CAPP for all columns (and then add them together).
- **Observation:** Each column can be viewed as a restriction!

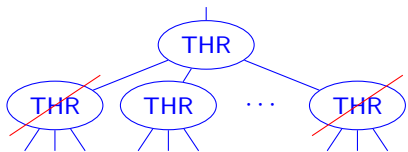
Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.



Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.

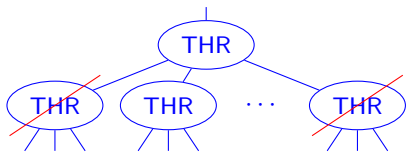


Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.
- **Proof:** A Toy example: n-bit MAJ gate
  - If non-constant, then

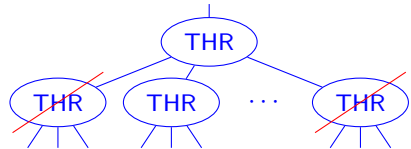
$$|\#\text{fixed 1-s} - \#\text{fixed 0-s}| \leq k$$

  - $\Pr\big[|\#\text{fixed 1-s} - \#\text{fixed 0-s}| \leq k\big] \leq k/\sqrt{n}$



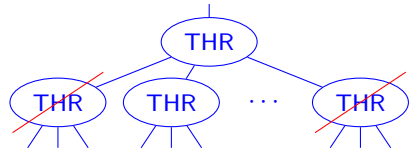
Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.

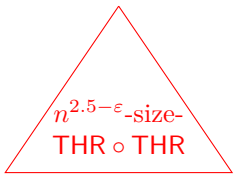


Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.

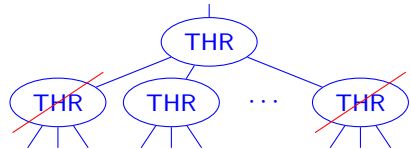


For $n^{2.5-\varepsilon}$-size $\mathsf{THR} \circ \mathsf{THR}$ circuits:

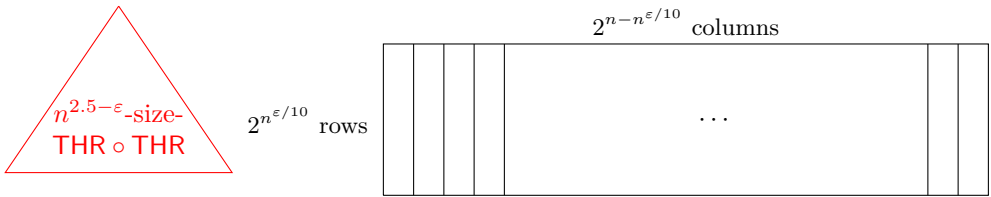


Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.

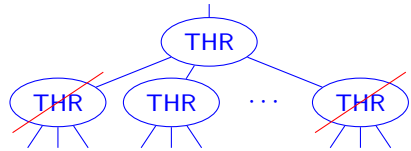


For $n^{2.5-\varepsilon}$-size THR $\circ$ THR circuits:

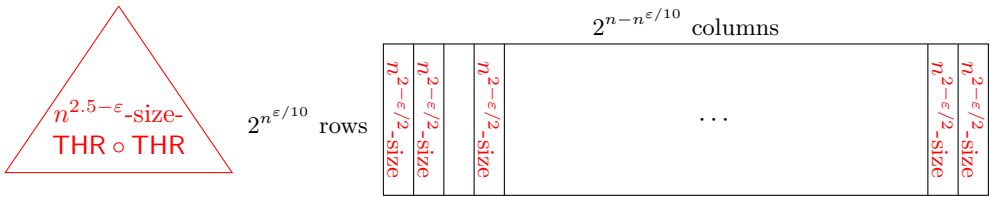


Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.



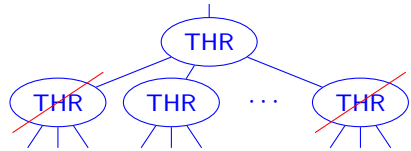
For $n^{2.5-\varepsilon}$-size THR $\circ$ THR circuits:



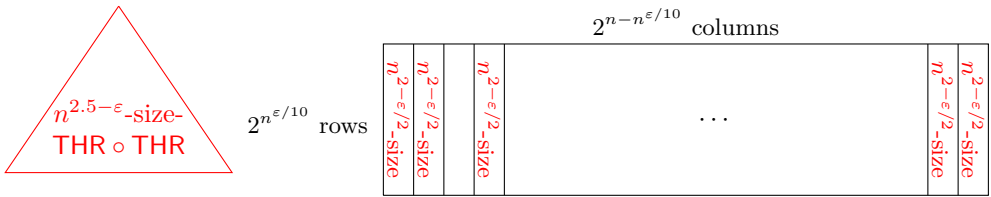
- Most columns become $n^{2-\varepsilon/2}$-size THR $\circ$ THR circuits!

Simplification of Threshold Gates under Random Restrictions

- **Random Restriction:** randomly set $k := n^{\varepsilon/10}$ input bits to be $\star$ (free variables), set each remaining bit to be fixed 0 or 1 randomly.
- **Theorem:** A THR gate remains non-constant w.p. $\approx k/\sqrt{n}$.



For $n^{2.5-\varepsilon}$-size THR $\circ$ THR circuits:



- Most columns become $n^{2-\varepsilon/2}$-size THR $\circ$ THR circuits!
- Issue: the $n^{2-\varepsilon/2}$-size circuits of each column are different.

Open Up Previous Proofs

- Why $n^{2-\varepsilon}$ in previous results?

Open Up Previous Proofs

- Why $n^{2-\varepsilon}$ in previous results?
  (Some details skipped: CAPP of $\mathsf{THR} \circ \mathsf{THR}$ can be reduced to CAPP of $\mathsf{MAJ} \circ \mathsf{THR}$)

Open Up Previous Proofs

- Why $n^{2-\varepsilon}$ in previous results?
  (Some details skipped: CAPP of $\mathsf{THR} \circ \mathsf{THR}$ can be reduced to CAPP of $\mathsf{MAJ} \circ \mathsf{THR}$)

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- Pointwise-Approximation: there is a distribution $\mathcal{D}$ over degree-$\widetilde{O}(\sqrt{m})$ polynomials such that:

$$\forall x \in \{0,1\}^m, \ \Pr_{P \sim \mathcal{D}}[P(x) = \mathsf{MAJ}(x)] \geq 1 - 1/m^{\omega(1)}.$$

Open Up Previous Proofs

- Why $n^{2-\varepsilon}$ in previous results?
  (Some details skipped: CAPP of $\mathsf{THR} \circ \mathsf{THR}$ can be reduced to CAPP of $\mathsf{MAJ} \circ \mathsf{THR}$)

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- Pointwise-Approximation: there is a distribution $\mathcal{D}$ over degree-$\widetilde{O}(\sqrt{m})$ polynomials such that:
$$\forall x \in \{0,1\}^m, \ \Pr_{P \sim \mathcal{D}}[P(x) = \mathsf{MAJ}(x)] \geq 1 - 1/m^{\omega(1)}.$$

- A CAPP algorithm for $\mathsf{POLY}[n^{1-\Omega(\varepsilon)}] \circ \mathsf{THR}$.

Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

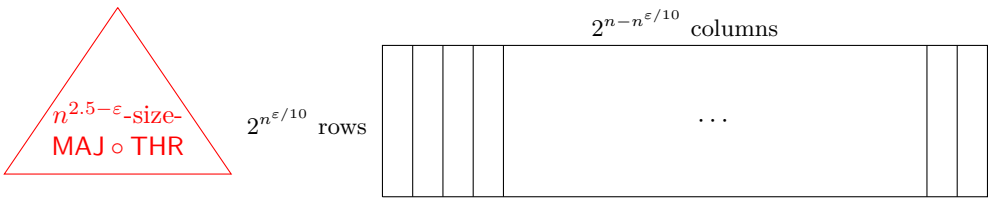
- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:

Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:

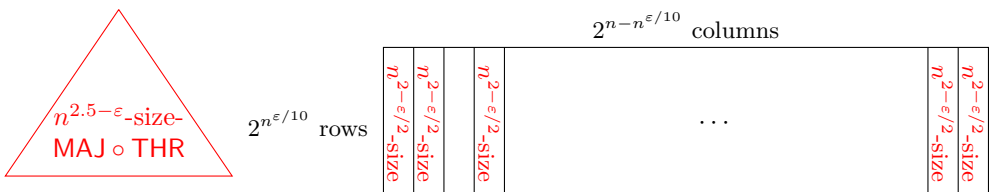


Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:

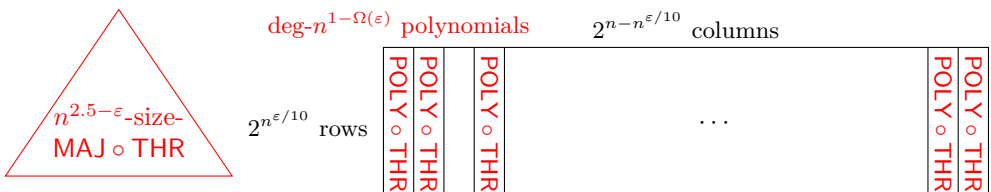


Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:

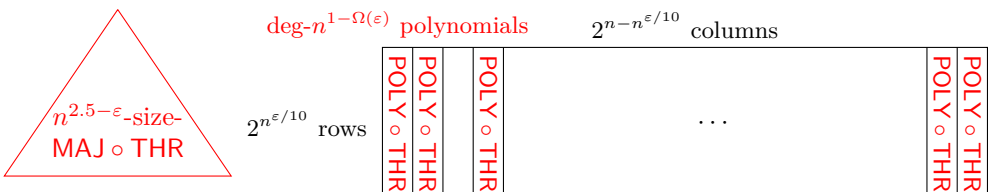


Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:



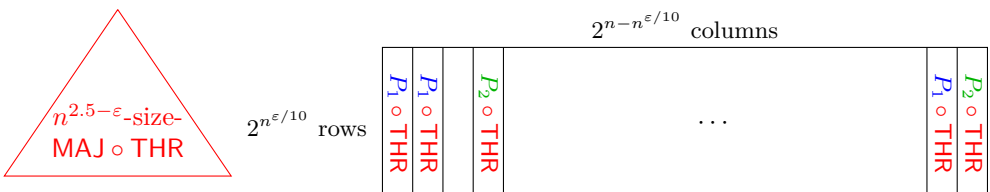
- **Our observation:** There are only $n^{10 \log n}$ many different polynomials!

Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:



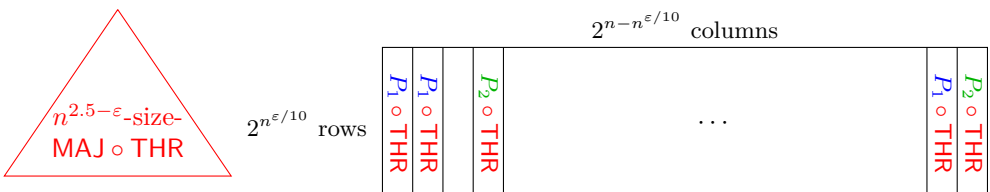
- **Our observation:** There are only $n^{10 \log n}$ many different polynomials!

Idea 2: List-Approximation Polynomials

Key Lemma (Alman-Williams' 15)

MAJ *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- For $n^{2.5-\varepsilon}$-size MAJ $\circ$ THR circuits:



- **Our observation:** There are only $n^{10 \log n}$ many different polynomials!
- For each $i \in \left[n^{10 \log n}\right]$, compute $P_i \circ$ THR $-$ CAPP on all columns;
- Then for each column $z \in \{0,1\}^{n-n^{\varepsilon/10}}$, look up the corresponding $P_i \circ$ THR $-$ CAPP.

[Alman-Williams' 15]'s Proof

- SUM with 1-hot encoding of output (1hotSUM):
  - Input: $(x_1, \cdots, x_m) \in \{0,1\}^m$;
  - Output: $(y_0, y_1, \cdots, y_m) \in \{0,1\}^{m+1}$:
    If $\sum_{i \in [m]} x_i = s$, then only the s-th bit is 1, the others are 0.

Theorem

1hotSUM *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- Pointwise-Approximation: there is a distribution $\mathcal{D}$ over $(m+1)$-tuple of degree-$\widetilde{O}(\sqrt{m})$ polynomials such that:

$$\forall x \in \{0,1\}^m, \Pr_{(P_0,P_1,\cdots,P_m)\sim\mathcal{D}}\left[(P_0(x), P_1(x), \cdots, P_m(x)) = \mathsf{1hotSUM}(x)\right] \geq 1 - 1/m^{\omega(1)}.$$

[Alman-Williams' 15]'s Proof

Theorem

1hotSUM *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- **Proof:** Use Induction.

[Alman-Williams' 15]'s Proof

Theorem

1hotSUM *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- **Proof:** Use Induction.
  - Randomly sample an $(m/2)$-subset $A \subseteq [m]$;
  - By induction, there is a distribution of polynomials $(Q_0, \cdots, Q_{m/2})$ that approximates $\mathsf{1hotSUM}(x_A)$;
  - By concentration inequalities, $s := \sum_{i \in A} x_i$ is a $\pm\widetilde{O}(\sqrt{m})$ approximation of $\frac{1}{2} \cdot \sum_{i \in [m]} x_i$ (w.h.p. over the choice of A);

[Alman-Williams' 15]'s Proof

Theorem

1hotSUM *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- **Proof:** Use Induction.
  - Randomly sample an $(m/2)$-subset $A \subseteq [m]$;
  - By induction, there is a distribution of polynomials $(Q_0, \cdots, Q_{m/2})$ that approximates $\mathsf{1hotSUM}(x_A)$;
  - By concentration inequalities, $s := \sum_{i \in A} x_i$ is a $\pm\widetilde{O}(\sqrt{m})$ approximation of $\frac{1}{2} \cdot \sum_{i \in [m]} x_i$ (w.h.p. over the choice of A);
  - Intuitively, Q gives a $\pm\widetilde{O}(\sqrt{m})$ estimate of $\sum_{i \in [m]} x_i$, then apply a degree-$\widetilde{O}(\sqrt{m})$ interpolation polynomial to compute the exact value;
  - Formally, let

$$P_k(x) := \sum_{s=k/2-\widetilde{O}(\sqrt{m})}^{k/2+\widetilde{O}(\sqrt{m})} Q_s(x_A) \cdot \underbrace{\left(\begin{cases} 1 & \text{if } \sum_{i \in [m]} x_i = k \\ 0 & \text{if } \sum_{i \in [m]} x_i \in [2s \pm \widetilde{O}(\sqrt{m})] \setminus \{k\} \\ \text{any} & \text{otherwise} \end{cases} \right)}_{\text{interpolation polynomial}}.$$

[Alman-Williams' 15]'s Proof

Theorem

1hotSUM *of m bits can be approximated by degree-$\widetilde{O}(\sqrt{m})$ polynomials over $\mathbb{F}_2$.*

- **Proof:** Use Induction.
  - Randomly sample an $(m/2)$-subset $A \subseteq [m]$;
  - By induction, there is a distribution of polynomials $(Q_0, \cdots, Q_{m/2})$ that approximates $\mathsf{1hotSUM}(x_A)$;
  - By concentration inequalities, $s := \sum_{i \in A} x_i$ is a $\pm\widetilde{O}(\sqrt{m})$ approximation of $\frac{1}{2} \cdot \sum_{i \in [m]} x_i$ (w.h.p. over the choice of A);
  - Intuitively, Q gives a $\pm\widetilde{O}(\sqrt{m})$ estimate of $\sum_{i \in [m]} x_i$, then apply a degree-$\widetilde{O}(\sqrt{m})$ interpolation polynomial to compute the exact value;
  - Formally, let

$$P_k(x) := \sum_{s=k/2-\widetilde{O}(\sqrt{m})}^{k/2+\widetilde{O}(\sqrt{m})} Q_s(x_A) \cdot \underbrace{\left(\begin{cases} 1 & \text{if } \sum_{i \in [m]} x_i = k \\ 0 & \text{if } \sum_{i \in [m]} x_i \in [2s \pm \widetilde{O}(\sqrt{m})] \setminus \{k\} \\ \text{any} & \text{otherwise} \end{cases} \right)}_{\text{interpolation polynomial}}.$$

  - In total: #rounds $= \log m$, degree: $\widetilde{O}(\sqrt{m}) \cdot \log m = \widetilde{O}(\sqrt{m})$.

Our Changes

- Recall: what do we need?

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;
  - **List-Approximation:** For different subcubes X's, there are at most $n^{10\log n}$ different polynomials.

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;
  - **List-Approximation:** For different subcubes X's, there are at most $n^{10\log n}$ different polynomials.
- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2\cdot\sum_{i\in A} x_i$:

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;
  - **List-Approximation:** For different subcubes X's, there are at most $n^{10\log n}$ different polynomials.
- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2\cdot\sum_{i\in A} x_i$:
  - **Error from free indices ($n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{free}} x_i \overset{\widetilde{O}(n^{1-\varepsilon/4})}{\approx\!\approx} 2\cdot\sum_{i\in A\cap\text{free}} x_i$$

  - **Error from fixed indices ($n^{2.5-\varepsilon}-n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i \overset{n^{1.25}}{\approx\!\approx} 2\cdot\sum_{i\in A\cap\text{fixed}} x_i \qquad \text{(bad)}$$

    However, the error is a fixed value for a given X !

Our Changes

- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2 \cdot \sum_{i\in A} x_i$:
  - **Error from free indices ($n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{free}} x_i \overset{\tilde{O}(n^{1-\varepsilon/4})}{\approx} 2 \cdot \sum_{i\in A\cap\text{free}} x_i$$

  - **Error from fixed indices ($n^{2.5-\varepsilon} - n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i \overset{n^{1.25}}{\approx} 2 \cdot \sum_{i\in A\cap\text{fixed}} x_i \qquad \text{(bad)}$$

    However, the error is a fixed value for a given X !

Our Changes

- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2\cdot\sum_{i\in A} x_i$:
  - **Error from free indices ($n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{free}} x_i \overset{\widetilde{O}(n^{1-\varepsilon/4})}{\approx\!\approx} 2\cdot\sum_{i\in A\cap\text{free}} x_i$$

  - **Error from fixed indices ($n^{2.5-\varepsilon}-n^{2-\varepsilon/2}$ bits)**:

$$\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i \overset{n^{1.25}}{\approx\!\approx} 2\cdot\sum_{i\in A\cap\text{fixed}} x_i \qquad \text{(bad)}$$

    However, the error is a fixed value for a given X !

- Hardcode $a := \left(\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i - 2\cdot\sum_{i\in A\cap\text{fixed}} x_i\right)$ as an advice, then

$$\sum_{i\in[n^{2.5-\varepsilon}]} x_i \overset{\pm\widetilde{O}(n^{1-\varepsilon/4})}{\approx\!\approx} 2\cdot\left(\sum_{i\in A} x_i\right) + a \qquad (\text{for } x\in X)$$

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;
  - **List-Approximation:** For different subcubes X's, there are at most $n^{10\log n}$ different polynomials.
- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2\cdot\sum_{i\in A} x_i$:
- Hardcode $a := \left(\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i - 2\cdot\sum_{i\in A\cap\text{fixed}} x_i\right)$ as an advice, then

$$\sum_{i\in[n^{2.5-\varepsilon}]} x_i \overset{\pm\widetilde{O}(n^{1-\varepsilon/4})}{\approx\approx} 2\cdot\left(\sum_{i\in A} x_i\right) + a \qquad (\text{for } x\in X)$$

Our Changes

- Recall: what do we need?
  - MAJ (top gate) on $n^{2.5-\varepsilon}$ bits;
  - Input is restricted to an $n^{2-\varepsilon/2}$-dim subcube X of $\{0,1\}^{2.5-\varepsilon}$;
    (Only $n^{2-\varepsilon/2}$ bottom THR gates are still alive. Others are fixed)
  - Construct deg-$n^{1-\Omega(\varepsilon)}$ polynomials that approximate MAJ on X;
  - **List-Approximation:** For different subcubes X's, there are at most $n^{10\log n}$ different polynomials.
- In [Alman-Williams' 15]'s proof: Estimation $\sum_{i\in[n^{2.5-\varepsilon}]} x_i \approx 2\cdot\sum_{i\in A} x_i$:
- Hardcode $a := \left(\sum_{i\in[n^{2.5-\varepsilon}]\cap\text{fixed}} x_i - 2\cdot\sum_{i\in A\cap\text{fixed}} x_i\right)$ as an advice, then

$$\sum_{i\in[n^{2.5-\varepsilon}]} x_i \overset{\pm\widetilde{O}(n^{1-\varepsilon/4})}{\approx\approx} 2\cdot\left(\sum_{i\in A} x_i\right) + a \qquad (\text{for } x\in X)$$

- **a-s of each induction step encodes everything we need to know about X !**
  Encoding length: $O\left((\log n)^2\right)$; #different polynomials: $n^{O(\log n)}$.

Summary

- **Our Result:**

$$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2.5-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR}$$

via designing an algorithm for

$$n^{2.5-\varepsilon}\text{-size-} \oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP}$$

(Previous: $n^{2-\varepsilon}$)

Summary

- **Our Result:**

$$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2.5-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR}$$

  via designing an algorithm for

$$n^{2.5-\varepsilon}\text{-size-} \oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP}$$

  (Previous: $n^{2-\varepsilon}$)

- **New Ideas behind our proof:**
  - Combine Random Restriction with the Algorithmic Method
  - Use a small list of approximation polynomials ($n^{10 \log n}$ many) to cover all cases ($2^{n-n^{\varepsilon/10}}$ columns)

Summary

- **Our Result:**

$$\mathsf{E}^{\mathsf{NP}} \not\subseteq n^{2.5-\varepsilon}\text{-size-}\mathsf{THR} \circ \mathsf{THR}$$

  via designing an algorithm for

$$n^{2.5-\varepsilon}\text{-size-} \oplus_2 \circ \mathsf{THR} \circ \mathsf{THR} - \mathsf{CAPP}$$

  (Previous: $n^{2-\varepsilon}$)

- **New Ideas behind our proof:**
  - Combine Random Restriction with the Algorithmic Method
  - Use a small list of approximation polynomials ($n^{10 \log n}$ many) to cover all cases ($2^{n-n^{\varepsilon/10}}$ columns)

- **Open Problems:**
  - Push forward circuit lower bounds?
  - Other applications of list approximation polynomials?

Thank You

- Thank you for listening.