Parity Helps to Compute Majority

Igor Carboni Oliveira
Rahul Santhanam
Srikanth Srinivasan

Computational Complexity Conference 2019
Background and Motivation
Bounded-depth boolean circuits

▶ AC$^0$: Bounded-depth circuits with AND, OR, NOT gates.

▶ A model that captures fast parallel computations.

▶ Close connections to logic and finite model theory.
We know a lot about $\mathsf{AC}^0$

- Explicit lower bounds: $2^\Omega(n^{1/(d-1)})$ for $\text{Parity}_n$ and $\text{Majority}_n$.

- Lower bound techniques have led to several advances:
  - Learning Algorithms for $\mathsf{AC}^0$ using random examples.
  - PRGs for $\mathsf{AC}^0$ with poly-log seed length.
  - Exponential lower bounds for $\mathsf{AC}^0$-Frege.
We know a lot about $\text{AC}^0$

- Explicit lower bounds: $2^{\Omega(n^{1/(d-1)})}$ for $\text{Parity}_n$ and $\text{Majority}_n$.

- Lower bound techniques have led to several advances:
  - Learning Algorithms for $\text{AC}^0$ using random examples.
  - PRGs for $\text{AC}^0$ with poly-log seed length.
  - Exponential lower bounds for $\text{AC}^0$-Frege.
This talk: $\text{AC}^0[\oplus]$ circuits

- $\text{AC}^0[\oplus]$: Extension of $\text{AC}^0$ by $\oplus$ (parity) gates.

- Parities can be very helpful: error-correcting codes, hash functions, $\text{GF}(2)$-polynomials, combinatorial designs, ...
This talk: \( \text{AC}^0[\oplus] \) circuits

- \( \text{AC}^0[\oplus] \): Extension of \( \text{AC}^0 \) by \( \oplus \) (parity) gates.

- Parities can be very helpful: error-correcting codes, hash functions, \( \text{GF}(2) \)-polynomials, combinatorial designs, . . .

- Explicit lower bounds: \( 2^{\Omega(n^{1/2(d-1)})} \) for \( \text{Majority}_n \).
This talk: $\text{AC}^0[\oplus]$ circuits

- $\text{AC}^0[\oplus]$: Extension of $\text{AC}^0$ by $\oplus$ (parity) gates.

- Parities can be very helpful: error-correcting codes, hash functions, $\text{GF}(2)$-polynomials, combinatorial designs, ... 

- Explicit lower bounds: $2^{\Omega(n^{1/2(d-1)})}$ for $\text{Majority}_n$.

- $\text{AC}^0$ and $\text{AC}^0[\oplus]$ are significantly different circuit classes: 
  **Example:** depth hierarchy for $\text{AC}^0$, depth collapse for $\text{AC}^0[\oplus]$.
Many fundamental questions remain wide open for $\text{AC}^0[\oplus]$. 

– Can we learn $\text{AC}^0[\oplus]$ using random examples?
– Are there PRGs of seed length $o(n)$?
– Does every tautology admit a short $\text{AC}^0[\oplus]$-Frege proof?
AC$^0$ versus AC$^0[⊕]$

Our primitive understanding of AC$^0[⊕]$ is reflected in part on existing lower bounds:

- Majority is one of the most studied boolean functions.
- Depth-$d$ AC$^0$ complexity of Majority is $2^{\tilde{\Theta}(n^{1/(d-1)})}$ \cite{1980}. (1980’s).
- Best known AC$^0[⊕]$ lower bound is $2^{\Omega(n^{1/2(d-1)})}$ for any $f \in$ NP. \cite{Razborov-Smolensky}

\text{(Razborov-Smolensky approximation method, 1980’s)}

Question. Can ⊕ gates help us computing Majority?
Our primitive understanding of $AC^0[\oplus]$ is reflected in part on existing lower bounds:

- Majority is one of the most studied boolean functions.
- Depth-$d$ $AC^0$ complexity of Majority is $2^{\tilde{\Theta}(n^{1/(d-1)})}$ (1980’s).
- Best known $AC^0[\oplus]$ lower bound is $2^{\Omega(n^{1/2(d-1)})}$ for any $f \in NP.$ (Razborov-Smolensky approximation method, 1980’s)

Question. Can $\oplus$ gates help us computing Majority?
Our primitive understanding of $\mathsf{AC}^0[\oplus]$ is reflected in part on existing lower bounds:

- Majority is one of the most studied boolean functions.
- Depth-$d\mathsf{AC}^0$ complexity of Majority is $2^{\tilde{\Theta}(n^{1/(d-1)})}$ (1980’s).
- Best known $\mathsf{AC}^0[\oplus]$ lower bound is $2^{\Omega(n^{1/2(d-1)})}$ for any $f \in \mathsf{NP}$.

(Razborov-Smolensky approximation method, 1980’s)

Question. Can $\oplus$ gates help us computing Majority?
Our primitive understanding of $\text{AC}^0[\oplus]$ is reflected in part on existing lower bounds:

- Majority is one of the most studied boolean functions.
- Depth-$d$ $\text{AC}^0$ complexity of Majority is $2^{\tilde{\Theta}(n^{1/(d-1)})}$ (1980’s).
- Best known $\text{AC}^0[\oplus]$ lower bound is $2^{\Omega(n^{1/2(d-1)})}$ for any $f \in \text{NP}$.

(Razborov-Smolensky approximation method, 1980’s)

**Question.** Can $\oplus$ gates help us computing Majority?
Why should we care?

1. Combinatorics: huge gap between $2^{n^{1/(d-1)}}$ and $2^{n^{1/2(d-1)}}$.

2. Can we beat the "obviously" optimal algorithm?

3. Parity gates play crucial role in hardness magnification. Example: "a layer of parities away from $\text{NC}^1$ lower bounds".

4. Better understanding of circuit complexity of a class $C$ often leads to progress w.r.t. related questions.
1. Combinatorics: huge gap between $2^{n^{1/(d-1)}}$ and $2^{n^{1/2(d-1)}}$.

2. Can we beat the “obviously” optimal algorithm?
1. Combinatorics: huge gap between $2^{n^{1/(d-1)}}$ and $2^{n^{1/2(d-1)}}$.

2. Can we beat the “obviously” optimal algorithm?

3. Parity gates play crucial role in hardness magnification. Example: “a layer of parities away from $NC^1$ lower bounds”.

Why should we care?
1. Combinatorics: huge gap between $2^{n^{1/(d-1)}}$ and $2^{n^{1/2(d-1)}}$.

2. Can we beat the “obviously” optimal algorithm?

3. Parity gates play crucial role in hardness magnification. 
   **Example**: “a layer of parities away from NC$^1$ lower bounds”.

4. Better understanding of circuit complexity of a class $C$ often leads to progress w.r.t. related questions.
Results
Neither the trivial upper bound of $2^{\tilde{O}(n^{1/(d-1)})}$ gates nor the Razborov-Smolensky lower bound $2^{\Omega(n^{1/2(d-1)})}$ is tight.

Our new upper and lower bounds for $\mathsf{AC^0}[\oplus]$ show that:

- Parity gates can speedup the computation of Majority for each large depth $d \in \mathbb{N}$.
- Indeed, the $\mathsf{AC^0}$ and $\mathsf{AC^0}[\oplus]$ complexities are similar at depth 3, but parity gates significantly help at depth 4.
Informal Summary

Neither the trivial upper bound of $2^{\tilde{O}(n^{1/(d-1)})}$ gates nor the Razborov-Smolensky lower bound $2^{\Omega(n^{1/2(d-1)})}$ is tight.

Our new upper and lower bounds for $\text{AC}_0^0[\oplus]$ show that:

- Parity gates can speedup the computation of Majority for each large depth $d \in \mathbb{N}$.

- Indeed, the $\text{AC}_0^0$ and $\text{AC}_0^0[\oplus]$ complexities are similar at depth 3, but parity gates significantly help at depth 4.
Recall: For \( d \geq 2 \), the depth-\( d \) \( AC^0 \) complexity of Majority \( n \) is \( 2^{\tilde{\Theta}(n^{1/(d-1)})} \).

Theorem 1. Let \( d \geq 5 \) be an integer. Majority on \( n \) bits can be computed by depth-\( d \) \( AC^0[\oplus] \) circuits of size \( 2^{\tilde{O}(n^{2/(d-4)})} \).

A similar upper bound holds for symmetric functions and linear threshold functions.
Recall: For $d \geq 2$, the depth-$d$ $\text{AC}^0$ complexity of $\text{Majority}_n$ is $2^{\tilde{\Theta}\left(n^{1/(d-1)}\right)}$.

**Theorem 1.** Let $d \geq 5$ be an integer. $\text{Majority}$ on $n$ bits can be computed by depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\widetilde{O}\left(n^{2/3 \cdot 1/(d-4)}\right)}$.

▶ A similar upper bound holds for **symmetric functions** and linear threshold functions.
Razborov-Smolensky

The depth-$d$ $\text{AC}^0[\oplus]$ complexity of Majority$_n$ is $2^{\Omega\left(n^{1/(2d-2)}\right)}$.

**Theorem 2.** Let $d \geq 3$ be an integer. Majority on $n$ bits requires depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\Omega\left(n^{1/(2d-4)}\right)}$.

- A small improvement of explicit lower bounds for $f \in \text{NP}$.
- This improvement is significant for very small $d$. 
Razborov-Smolensky

The depth-$d$ $\text{AC}^0[\oplus]$ complexity of $\text{Majority}_n$ is $2^{\Omega\left(n^{1/(2d-2)}\right)}$.

**Theorem 2.** Let $d \geq 3$ be an integer. Majority on $n$ bits requires depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\Omega\left(n^{1/(2d-4)}\right)}$.

- A small improvement of explicit lower bounds for $f \in \text{NP}$.
- This improvement is significant for very small $d$. 
The small depth regime

New lower bound + extension of upper bound techniques yield:

Corollary 1.

The depth-3 $\text{AC}^0[\oplus]$ circuit size complexity of Majority is $2^{\tilde{\Theta}(n^{1/2})}$.

The depth-4 $\text{AC}^0[\oplus]$ circuit size complexity of Majority is $2^{\tilde{\Theta}(n^{1/4})}$.

- Parity gates significantly help at depth 4 but not at depth 3.
Techniques: $AC^0[⊕]$ Upper Bounds
Theorem 1. Let $d \geq 5$ be an integer. Majority on $n$ bits can be computed by depth-$d$ $\text{AC}_0[\oplus]$ circuits of size $2^{\tilde{O}\left(n^{\frac{2}{3}} \cdot \frac{1}{(d-4)}\right)}$.

$$E_i(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{otherwise.} \end{cases}$$

$$D_{i,j}(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{if } |y|_1 = j. \end{cases}$$

Goal: $\text{AC}_0[\oplus]$ circuits of size $\approx 2^{n^{2/3d}}$ for all $D_{i,j}$, $0 \leq i \neq j \leq n$. 
Theorem 1. Let $d \geq 5$ be an integer. Majority on $n$ bits can be computed by depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\tilde{O}(n^{2/3}(d-4))}$.

$$E_i(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{otherwise.} \end{cases}$$

$$D_{i,j}(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{if } |y|_1 = j. \end{cases}$$

Goal: $\text{AC}^0[\oplus]$ circuits of size $\approx 2^{n^{2/3d}}$ for all $D_{i,j}$, $0 \leq i \neq j \leq n$. 
Theorem 1. Let $d \geq 5$ be an integer. Majority on $n$ bits can be computed by depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\tilde{O}\left(n^{2/3}\cdot \frac{1}{(d-4)}\right)}$.

$$E_i(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{otherwise.} \end{cases} \quad \text{and} \quad D_{i,j}(y) = \begin{cases} 1 & \text{if } |y|_1 = i, \\ 0 & \text{if } |y|_1 = j. \end{cases}$$

Goal: $\text{AC}^0[\oplus]$ circuits of size $\approx 2^{n^{2/3d}}$ for all $D_{i,j}$, $0 \leq i \neq j \leq n$. 
The $D_{i,j}$ partial boolean function

$D_{i,j} : \{0, 1\}^n \rightarrow \{0, 1\}$

$i, j \in [n]$

$D_{i,j}(x) = 1$

$D_{i,j}(x) = 0$

$D_{i,j}(x) \in \{0, 1\}$
We consider the value $|i - j|$

- **Small regime:** $|i - j| \leq n^{1/3}$.

We use an "algebraic" construction. This circuit relies on a $\mathbb{F}_2$ polynomial, divide-and-conquer, and needs $\oplus$ gates.

- **Large regime:** $|i - j| > n^{1/3}$.

We use a "combinatorial" construction. This circuit relies on a probabilistic construction of $\text{AC}^0$ circuits for the Coin Problem.
We consider the value $|i - j|:

- **Small regime:** $|i - j| \leq n^{1/3}$.

  We use an "algebraic" construction. This circuit relies on a $\mathbb{F}_2$ polynomial, divide-and-conquer, and needs $\oplus$ gates.

- **Large regime:** $|i - j| > n^{1/3}$.

  We use a "combinatorial" construction. This circuit relies on a probabilistic construction of $\text{AC}^0$ circuits for the *Coin Problem*. 
$|i - j| \leq n^{1/3}$: The algebraic construction I

Lemma [AW15]:

$c_1, c_2, \ldots, c_\ell \in \mathbb{Z}$

There is a polynomial $Q: \{0, 1\}^n \to \mathbb{Z}$ such that:

$Q(x) = c_i$ when $|x|_1$ agrees with corresponding layer.

Moreover,

$\deg(Q) \leq \ell - 1$ and $Q(x) = \sum_{t=0}^{\ell-1} a_t \cdot Q_t(x)$

$Q_t(x) = \sum_{S \in \binom{[n]}{t}} \prod_{j \in S} x_j$

$t$-th symmetric elementary polynomial
$|i - j| \leq n^{1/3}$: The algebraic construction II

- $Q(x_1, \ldots, x_n)$ is defined over $\mathbb{Z}$. We take a homomorphism $\psi: \mathbb{Z} \rightarrow \mathbb{F}_2$.

\[
P(x) = \sum_{t=0}^{\ell-1} b_t \cdot P_t(x) \text{ over } \mathbb{F}_2, \text{ where } \ell = (i - j) + 1.
\]

- $P(x)$ computes $D_{i,j}(x)$ and has degree at most $\ell \leq n^{1/3}$.

- We would like to compute $P(x)$ in depth-$d \text{ AC}^0[\oplus]$.

- Goal: elementary symmetric polynomials $Q_1, \ldots, Q_\ell$, where $\ell \leq n^{1/3}$.
\( |i - j| \leq n^{1/3} \): The algebraic construction II

- \( Q(x_1, \ldots, x_n) \) is defined over \( \mathbb{Z} \). We take a homomorphism \( \psi: \mathbb{Z} \rightarrow \mathbb{F}_2 \).

\[
P(x) = \sum_{t=0}^{\ell-1} b_t \cdot P_t(x) \text{ over } \mathbb{F}_2, \text{ where } \ell = (i - j) + 1.
\]

- \( P(x) \) computes \( D_{i,j}(x) \) and has degree at most \( \ell \leq n^{1/3} \).

- We would like to compute \( P(x) \) in depth-\( d \) \( \text{AC}^0[\oplus] \).

- Goal: elementary symmetric polynomials \( Q_1, \ldots, Q_{\ell} \) where \( \ell \leq n^{1/3} \).
$|i - j| \leq n^{1/3}$: The algebraic construction II

- $Q(x_1, \ldots, x_n)$ is defined over $\mathbb{Z}$. We take a homomorphism $\psi: \mathbb{Z} \to \mathbb{F}_2$.

  $$P(x) = \sum_{t=0}^{\ell-1} b_t \cdot P_t(x) \quad \text{over } \mathbb{F}_2, \text{ where } \ell = (i - j) + 1.$$ 

- $P(x)$ computes $D_{i,j}(x)$ and has degree at most $\ell \leq n^{1/3}$.

- We would like to compute $P(x)$ in depth-$d$ $\text{AC}^0[\oplus]$.

- **Goal**: elementary symmetric polynomials $Q_1, \ldots, Q_\ell$, where $\ell \leq n^{1/3}$. 
\[ P_\ell(x_1, \ldots, x_n) = \sum_{S \in \binom{[n]}{\ell}} \prod_{j \in S} x_j \]

We simulate \( P_\ell \) using an algebraic branching program:

Divide-and-conquer approach similar to depth-\( d \) circuit for STCONN:

We can compute \( P_\ell \) using \( \land \) and \( \lor \) in depth \( d \) and size \( n^{O(\ell^2/d)} \).

For \( \ell \leq n^{1/3} \), this gives \( \text{AC}^0[\lor] \) circuit size \( 2^{\tilde{O}(n^{2/3}d)} \).
$|i - j| \leq n^{1/3}$: The algebraic construction III

\[ P_\ell(x_1, \ldots, x_n) = \sum_{S \in \binom{[n]}{\ell}} \prod_{j \in S} x_j \]

We simulate $P_\ell$ using an algebraic branching program:

\[
\sum \prod \text{width } n \\
\text{length } \ell + 1
\]

**Divide-and-conquer** approach similar to depth-$d$ circuit for STCONN:

We can compute $P_\ell$ using $\wedge$ and $\oplus$ in depth $d$ and size $n^{O(\ell^2/d)}$.

For $\ell \leq n^{1/3}$, this gives $\text{AC}^0[\oplus]$ circuit size $2^{O(n^{3/4d})}$. 
$i - j \leq n^{1/3}$: The algebraic construction III

$$P_\ell(x_1, \ldots, x_n) = \sum_{S \in ([n] \atop \ell)} \prod_{j \in S} x_j$$

We simulate $P_\ell$ using an algebraic branching program:

Divide-and-conquer approach similar to depth-$d$ circuit for STCONN:

We can compute $P_\ell$ using $\land$ and $\lor$ in depth $d$ and size $n^{O(\ell^2/d)}$.

For $\ell \leq n^{1/3}$, this gives $\text{AC}^0[\lor]$ circuit size $2^{\tilde{O}(n^{2/3d})}$. 
By moving from $n$ to $\Theta(n)$ input bits, we can assume $i$ and $j$ are equally spaced from middle layer.

Let $i = n/2 + t$ and $j = n/2 - t$. Enough to compute Approximate Majority / Coin Problem.

Elegant construction [OW07], [Ama09], [RS17]: Can be done by depth-$d$ $\text{AC}^0$ circuits of size roughly $2^{(n/t)^{1/d}}$.

For $t = \Theta(|i - j|) > n^{1/3}$, this size bound is $2^{O(n^{2/3d})}$. 
\[ i - j > n^{1/3} \]: The combinatorial construction

By moving from \( n \) to \( \Theta(n) \) input bits, we can assume \( i \) and \( j \) are equally spaced from middle layer.

Let \( i = n/2 + t \) and \( j = n/2 - t \). Enough to compute Approximate Majority / Coin Problem.

Elegant construction [OW07], [Ama09], [RS17]:
Can be done by depth-\( d \) AC\(^0\) circuits of size roughly \( 2^{(n/t)^{1/d}} \).

For \( t = \Theta(|i - j|) > n^{1/3} \), this size bound is \( 2^{O(n^{2/3d})} \).
| $i - j| > n^{1/3}$: The combinatorial construction

By moving from $n$ to $\Theta(n)$ input bits, we can assume $i$ and $j$ are equally spaced from middle layer.

Let $i = n/2 + t$ and $j = n/2 - t$. Enough to compute \textit{Approximate Majority / Coin Problem}.

\textbf{Elegant construction} [OW07], [Ama09], [RS17]:
Can be done by depth-$d$ $\text{AC}^0$ circuits of size roughly $2^{(n/t)^{1/d}}$.

For $t = \Theta(|i - j|) > n^{1/3}$, this size bound is $2^{O(n^{2/3d})}$. 
By moving from $n$ to $\Theta(n)$ input bits, we can assume $i$ and $j$ are equally spaced from middle layer.

Let $i = n/2 + t$ and $j = n/2 - t$. Enough to compute Approximate Majority / Coin Problem.

Elegant construction [OW07], [Ama09], [RS17]:
Can be done by depth-$d$ $AC^0$ circuits of size roughly $2^{(n/t)^{1/d}}$.

For $t = \Theta(|i - j|) > n^{1/3}$, this size bound is $2^{O(n^{2/3d})}$. 
Extensions of the Upper Bound

- Previous argument works for all **symmetric functions**.

- In depth $d = 4$, careful depth control + new ingredient: *randomly splitting variables into buckets*.

- Linear Threshold Functions (LTFs) and Polytopes: $\text{AC}^0$ reduction to Exact Threshold Functions (ETH) via [HP10], then reduction to symmetric functions (Chinese remaindering).
Extensions of the Upper Bound

- Previous argument works for all **symmetric functions**.

- In depth $d = 4$, careful depth control + new ingredient: *randomly splitting variables into buckets.*

- Linear Threshold Functions (LTFs) and Polytopes: $AC^0$ reduction to Exact Threshold Functions (ETH) via [HP10], then reduction to symmetric functions (Chinese remaindering).
Extensions of the Upper Bound

► Previous argument works for all **symmetric functions**.

► In depth $d = 4$, careful depth control + new ingredient: *randomly splitting variables into buckets*.

► **Linear Threshold Functions** (LTFs) and **Polytopes**:
  AC$^0$ reduction to Exact Threshold Functions (ETH) via [HP10], then reduction to symmetric functions (Chinese remaindering).
Techniques: $AC^0[⊕]$ Lower Bounds
Theorem 2. Let $d \geq 3$ be an integer. Majority on $n$ bits requires depth-$d$ $\text{AC}^0[\oplus]$ circuits of size $2^{\Omega(n^{1/(2d-4)})}$.

Recall: Razborov-Smolensky shows a $2^{\Omega(n^{1/(2d-2)})}$ lower bound.

Intuition: How to save two layers of gates in the polynomial approximation method?
Degree Upper Bound:

Probabilistic polynomial $P$ over $\mathbb{F}_2$ correct on each input w.h.p. AND, OR, NOT, PARITY: error $\varepsilon$ and degree $\log(1/\varepsilon)$

Size-$s$ depth-$d$ $\text{AC}^0[\oplus]$: $\deg(P) \approx (\log s)^{d-1}$ and error $\varepsilon \leq 1/50$.

Degree Lower Bound:

For Majority$_n$, $\deg(P)$ must be $\geq \sqrt{n \cdot \log(1/\varepsilon)}$. 
Degree Upper Bound:

Probabilistic polynomial $P$ over $\mathbb{F}_2$ correct on each input w.h.p. AND, OR, NOT, PARITY: error $\varepsilon$ and degree $\log(1/\varepsilon)$

Size-$s$ depth-$d$ $\text{AC}^0[\oplus]$: $\deg(P) \approx (\log s)^{d-1}$ and error $\varepsilon \leq 1/50$.

Degree Lower Bound:

For Majority$_n$, $\deg(P)$ must be $\geq \sqrt{n \cdot \log(1/\varepsilon)}$. 
Putting together the approximate degree bounds:

\[(\log s)^{d-1} \geq \sqrt{n \cdot \log(1/\varepsilon)}, \quad \varepsilon = 1/50.\]

This implies that \( s \geq 2^{\Omega(n^{1/(2d-2)})}. \)

(The RS lower bound is maximized when \( \varepsilon = \text{constant}. \))
Our approach

We follow Razborov-Smolensky, with two new ideas.

Idea 1. Exploit error $\varepsilon = 1/50$ of polynomial approximator:
– Error is one-sided and $\leq 1/\log s$ on say $C^{-1}(1)$.
– Hope to exploit stronger degree lower bound of $\sqrt{n \cdot \log(1/\varepsilon)}$.

Idea 2. Random restrictions for $\text{AC}^0[\oplus]$ circuits:
– Prove that w.h.p. a random restriction leads to depth-2 subcircuits of smaller approximate degree. Can do better than $(\log s)^2$ on bottom layers.
Our approach

We follow Razborov-Smolensky, with two new ideas.

**Idea 1.** Exploit error $\varepsilon = 1/50$ of polynomial approximator:

- Error is **one-sided** and $\leq 1/\log s$ on say $C^{-1}(1)$.
- Hope to exploit stronger degree lower bound of $\sqrt{n \cdot \log(1/\varepsilon)}$.

**Idea 2.** Random restrictions for $\text{AC}^0[\oplus]$ circuits:

- Prove that w.h.p. a random restriction leads to depth-2 subcircuits of smaller approximate degree. Can do better than $(\log s)^2$ on bottom layers.
First idea: One-sided approximations

- We approximate every non-output gate to error $\leq \frac{1}{s^2}$.

- By union bound, every input wire of output gate is correct (except with prob. $\leq \frac{1}{s}$).

- Approximation method over OR gate is one-sided ("random parities"): zero inputs to OR gate always produce zero.
First idea: Stronger degree lower bound

▶ Smolensky’s approximate degree lower bound:

\[ \deg_\varepsilon(Maj_n) = \Omega(\sqrt{n \cdot \log(1/\varepsilon)}) \).

Can we maintain this lower bound when error on \( Maj_n^{-1}(0) \) is \( \leq \varepsilon \) but error on \( Maj_n^{-1}(1) \) is as large as \( 1/50 \)?

▶ We extend the technique of certifying polynomials [KS12] to show this is the case.
First idea: Stronger degree lower bound

▶ Smolensky’s approximate degree lower bound:

\[
\deg_\varepsilon(\text{Majority}_n) = \Omega(\sqrt{n \cdot \log(1/\varepsilon)}).
\]

Can we maintain this lower bound when error on \(\text{Majority}^{-1}_n(0)\) is \(\leq \varepsilon\) but error on \(\text{Majority}^{-1}_n(1)\) is as large as \(1/50\)?

▶ We extend the technique of certifying polynomials [KS12] to show this is the case.
Second idea: random restrictions for $\text{AC}^0[\oplus]$

- We prove the following lemma:

**Random Restriction Lemma.** Let $C$ be a depth-2 $\text{AC}^0[\oplus]$ circuit on $n$ vars and of size $s \geq n^2$. Let $p_* \leq 1/(500 \log s)$. Then,

$$\Pr_{\rho \sim \mathcal{R}_{p*}^n} \left[ \deg_{\epsilon=1/s^2}(C|\rho) > 10 \log s \mid \rho \text{ is balanced} \right] < \frac{1}{10s}.$$ 

- Case analysis based on gates of $C$ (OR, AND, PARITY).
We prove the following lemma:

**Random Restriction Lemma.** Let $C$ be a depth-2 $\text{AC}^0[\oplus]$ circuit on $n$ vars and of size $s \geq n^2$. Let $p_* \leq 1/(500 \log s)$. Then,

$$\mathbb{P}_{\rho \sim \mathcal{R}_{p_*}^n} [\deg_{\varepsilon=1/s^2} (C|\rho) > 10 \log s \mid \rho \text{ is balanced}] < \frac{1}{10s}.$$ 

Case analysis based on gates of $C$ (OR, AND, PARITY).
Concluding Remarks
Challenge: What is the $\text{AC}^0[\oplus]$ complexity of Majority?

- Close the gap between the $2^{\tilde{O}(n^{2/3(d-4)})}$ upper bound and the $2^\Omega(n^{1/(2d-4)})$ lower bound.

- Find more examples where the “optimal” algorithm or circuit can be improved.
Open Problems

Challenge: What is the $\text{AC}^0[\oplus]$ complexity of Majority?

- Close the gap between the $2^{\tilde{O}\left(n^{2/(d-4)}\right)}$ upper bound and the $2^{\Omega\left(n^{1/(2d-4)}\right)}$ lower bound.

- Find more examples where the “optimal” algorithm or circuit can be improved.