Size Bounds on Low Depth Circuits for Promise Majority

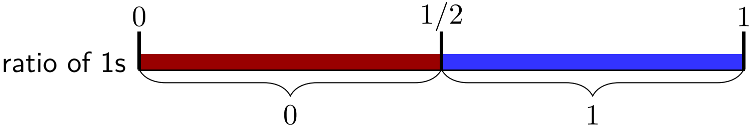
Joshua Cook

The University of Texas at Austin

July 3, 2022

Talk Outline

Result Overview

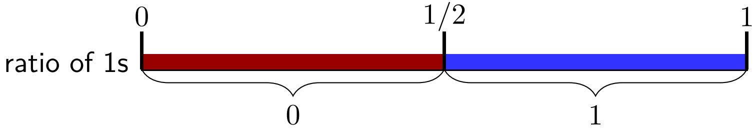


Definition (Majority)

For $n \in \mathbf{N}$, let Maj : $\{0,1\}^n \to \{0,1\}$ be defined by

$$\mathsf{Maj}(x) = \mathbf{1}\left[\sum_i x_i \geq n/2\right].$$

Majority



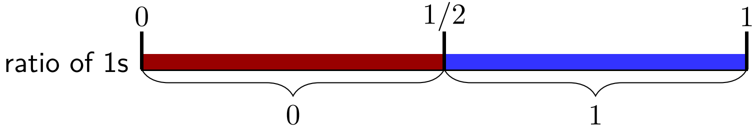
Definition (Majority)

For $n \in \mathbf{N}$, let Maj : $\{0,1\}^n \to \{0,1\}$ be defined by

$$\mathrm{Maj}(x) = \mathbf{1}\left[\sum_i x_i \geq n/2\right].$$

- Component of many results, such as circuit derandomization [1].

Majority



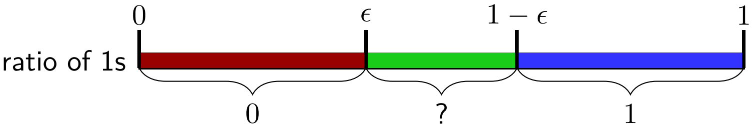
Definition (Majority)

For $n \in \mathbf{N}$, let Maj : $\{0,1\}^n \to \{0,1\}$ be defined by

$$\text{Maj}(x) = \mathbf{1}\left[\sum_i x_i \geq n/2\right].$$

- Component of many results, such as circuit derandomization [1].
- Widely studied, not computable by AC0, simple computation models.

Promise Majority



Approximate majority[2], promise majority[6], approximate selector[4], etc.

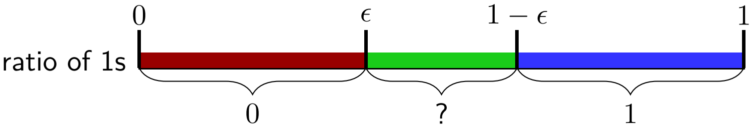
Definition (Promise Majority)

For $n \in \mathbf{N}$, $\epsilon \in (0, 1/2)$, and function $f : \{0,1\}^n \to \{0,1\}$, we say f solves ϵ-promise majority if for all $x \in \{0,1\}^n$ with $\sum_{i \in [n]} x_i < \epsilon n$ and for all $y \in \{0,1\}^n$ with $\sum_{i \in [n]} 1 - y_i < \epsilon n$

$$f(x) = 0, f(y) = 1.$$

- Often usable in place of majority, in circuit derandomization.

Promise Majority



Approximate majority[2], promise majority[6], approximate selector[4], etc.

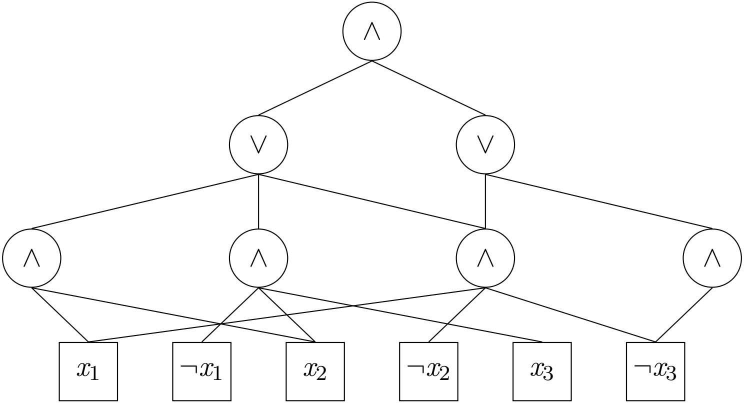
Definition (Promise Majority)

For $n \in \mathbf{N}$, $\epsilon \in (0, 1/2)$, and function $f : \{0,1\}^n \to \{0,1\}$, we say f solves ϵ-promise majority if for all $x \in \{0,1\}^n$ with $\sum_{i\in[n]} x_i < \epsilon n$ and for all $y \in \{0,1\}^n$ with $\sum_{i\in[n]} 1 - y_i < \epsilon n$

$$f(x) = 0, f(y) = 1.$$

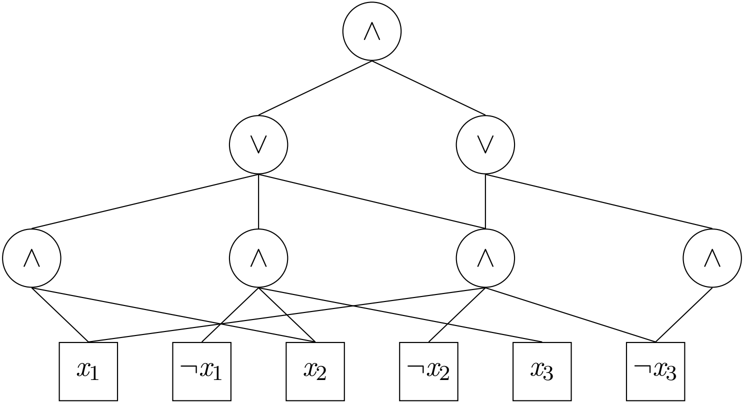
- Often usable in place of majority, in circuit derandomization.
- Widely studied, computable by AC0.

AC0



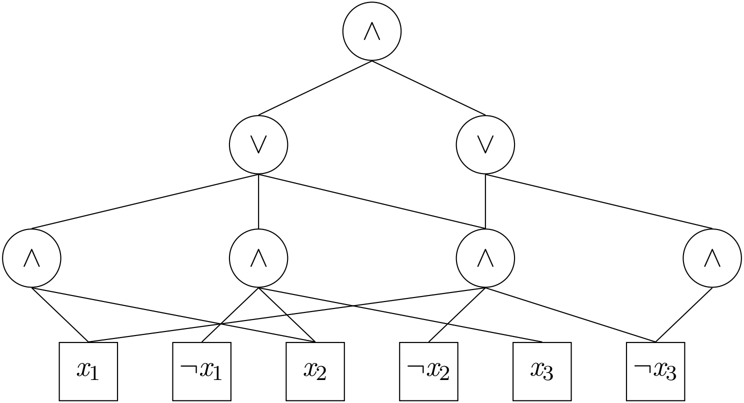
- Alternating circuit: unbounded fan in "AND" and "OR" gates.

AC0



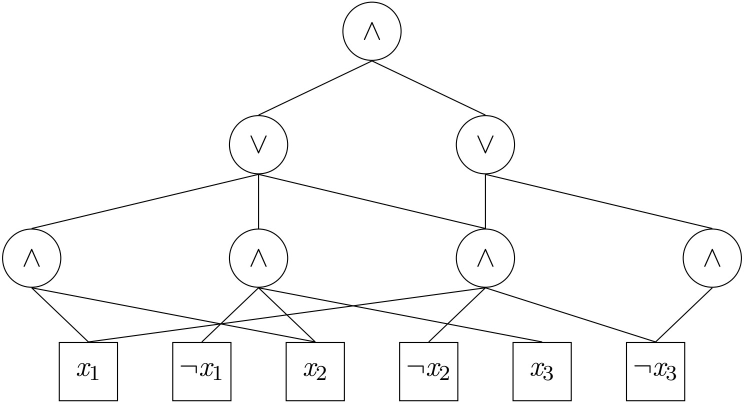
- Alternating circuit: unbounded fan in "AND" and "OR" gates.
- Layers "Alternate" between "AND" and "OR" gates.

AC0



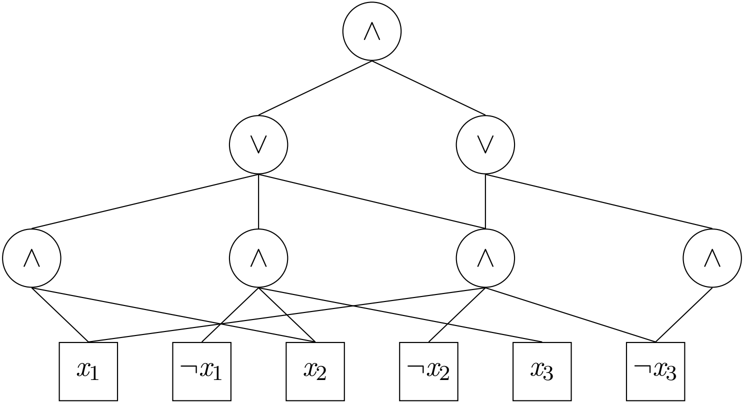
- Alternating circuit: unbounded fan in “AND” and “OR” gates.
- Layers “Alternate” between “AND” and “OR” gates.
- Bottom layer includes negated inputs.

AC0



- Alternating circuit: unbounded fan in "AND" and "OR" gates.
- Layers "Alternate" between "AND" and "OR" gates.
- Bottom layer includes negated inputs.
- Size is number of gates (same results for wires).

AC0



- Alternating circuit: unbounded fan in "AND" and "OR" gates.
- Layers "Alternate" between "AND" and "OR" gates.
- Bottom layer includes negated inputs.
- Size is number of gates (same results for wires).
- AC0 constant depth, polynomial size.

Depth-3 ϵ-Promise Circuit Upper Bounds

Depth-3 Upper Bounds:

Author	ϵ	Size	Uniformity
Ajtai 1983 [2]	$(0, 1/2)$	$(\epsilon \ln(\epsilon) n)^{2 + \frac{\ln(1-\epsilon)}{\ln(\epsilon) - \ln(1-\epsilon)}}$	Non-Uniform
Viola 2009 [7]	$\frac{1}{\ln(n)}$	$n^{4+o(1)}$	P
Viola 2009 [7]	$(0, 1/2)$	$n^{4+O((1-2\epsilon)^{-2})}$	P
Us	$\frac{1}{\ln(n)}$	$n^{3+o(1)}$	P

Depth-3 ϵ-Promise Circuit Lower Bounds

Depth-3 Lower Bounds (Suppressing polylogarithmic factors):

Author	Size	Monotone
Trivial	ϵn	General
Chaudhuri, Radhakrishnan 1996 [4]	$(\epsilon n)^{\frac{64}{63}} - n$	General
Viola 2011 [8]	$n^{\Omega(-\ln(1-2\epsilon))}$	General
Us	$\epsilon^3 n^{2+\frac{\ln(1-\epsilon)}{\ln(\epsilon)}}$	Monotone
Us	$\epsilon^3 n^{2+\frac{\ln(1-\epsilon^2)}{2\ln(\epsilon)}}$	General

Higher Depth ϵ-Promise Circuit Upper Bounds

Upper Bounds (Constant ϵ):

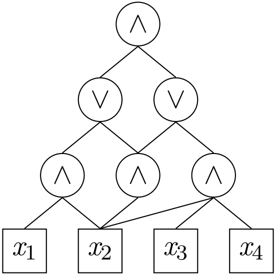
Author	Size	Uniformity
Ajtai 1990 [3]	$\text{poly}(n)$	LOGTIME
Chaudhuri, Radhakrishnan 1996 [4]	$n^{\frac{1}{1-2^{-O(d)}}}$	LOGTIME
Us	$n^{\frac{1}{1-2^{-(d-2)/2}}}$	Non-Uniform
Us	$n^{\frac{1}{1-(2/3)^{(d-2)/2}}}$	P

Higher Depth ϵ-Promise Circuit Lower Bounds

Lower Bounds:

Author	ϵ	Size
Trivial	any	ϵn
Chaudhuri, Radhakrishnan 1996 [4]	any	$(\epsilon n)^{\frac{1}{1-4^{-d}}} - n$
Viola 2011 [8]	$\frac{1}{2} - \frac{1}{\ln(n)^{d-2}}$	$\omega(\text{poly}(n))$

Depth 3 Circuits Terminology

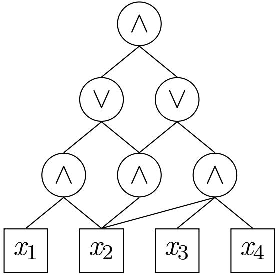


Focus on depth-3 promise Majority

- Negation of promise majority circuit, also promise majority. Assume lowest level gate is “AND”.

Depth 3 Circuits Terminology

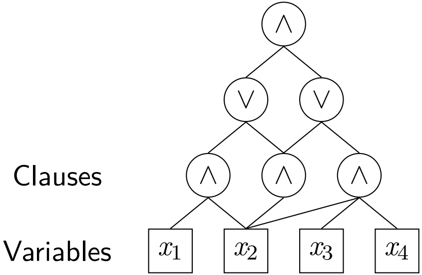
Variables



Focus on depth-3 promise Majority

- Negation of promise majority circuit, also promise majority. Assume lowest level gate is "AND".
- Call input bits "variables".

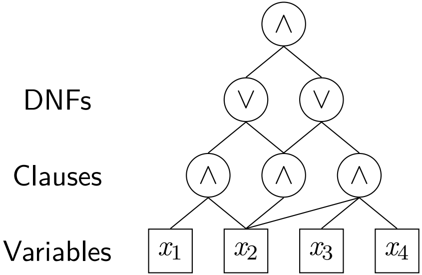
Depth 3 Circuits Terminology



Focus on depth-3 promise Majority

- Negation of promise majority circuit, also promise majority. Assume lowest level gate is "AND".
- Call input bits "variables".
- First level, AND gates "clauses".

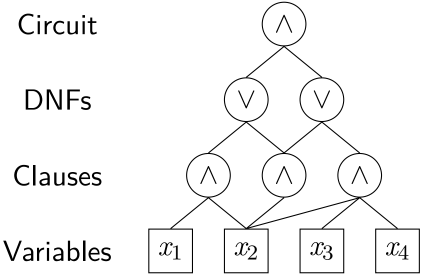
Depth 3 Circuits Terminology



Focus on depth-3 promise Majority

- Negation of promise majority circuit, also promise majority. Assume lowest level gate is “AND”.
- Call input bits “variables”.
- First level, AND gates “clauses”.
- Second level, OR gates “DNFs”.

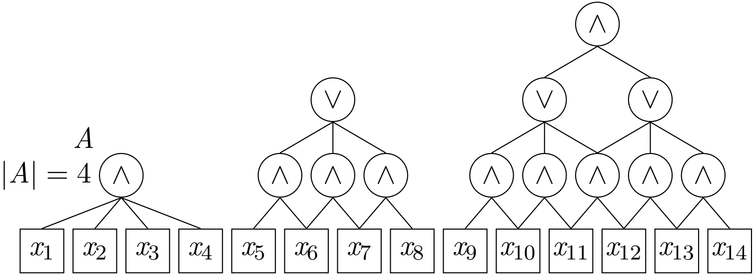
Depth 3 Circuits Terminology



Focus on depth-3 promise Majority

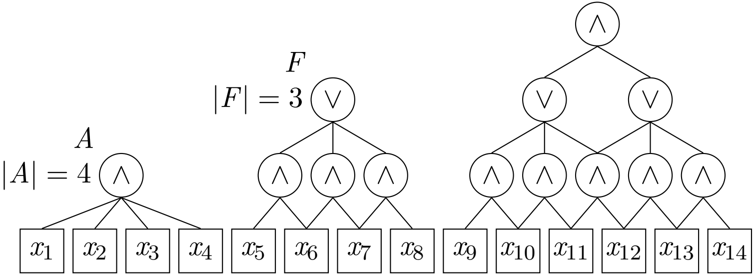
- Negation of promise majority circuit, also promise majority. Assume lowest level gate is “AND”.
- Call input bits “variables”.
- First level, AND gates “clauses”.
- Second level, OR gates “DNFs”.
- Third level, AND gate “circuits”.

Size Definitions



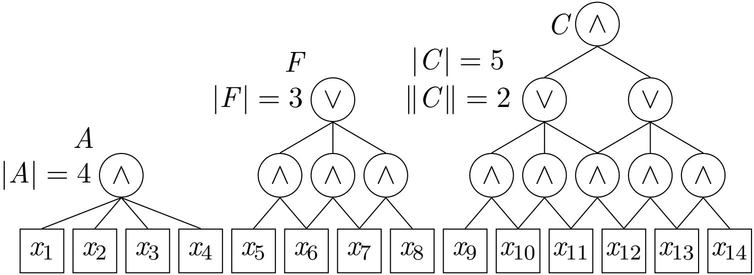
- Clause A, size $|A|$ is the number of variables in A.

Size Definitions



- Clause A, size $|A|$ is the number of variables in A.
- DNF F, size $|F|$ is the number of clauses in F.

Size Definitions



- Clause A, size $|A|$ is the number of variables in A.
- DNF F, size $|F|$ is the number of clauses in F.
- If C is a circuit, denote
  - $|C|$ as the number of clauses in C.
  - $\|C\|$ as the number of DNFs in C.
  - the size of C as $|C| + \|C\|$.

Monotone Lower Bound Idea

Idea: Lower bound the fan in at each layer.

Pretend $\epsilon \in (0, 1/2)$ is constant for simplicity. Let $\alpha = \frac{\ln(1-\epsilon)}{\ln(\epsilon)}$.

1. From Viola [7], clauses have size $\frac{\ln(n)}{\ln(1/\epsilon)}$.

Monotone Lower Bound Idea

Idea: Lower bound the fan in at each layer.

Pretend $\epsilon \in (0, 1/2)$ is constant for simplicity. Let $\alpha = \frac{\ln(1-\epsilon)}{\ln(\epsilon)}$.

1. From Viola [7], clauses have size $\frac{\ln(n)}{\ln(1/\epsilon)}$.
2. If DNFs have size $\tilde{o}(n^{1+\alpha})$, then we can hit every clause with fewer than ϵn variables.
   Thus clauses have size $\tilde{\Omega}(n^{1+\alpha})$.

Monotone Lower Bound Idea

Idea: Lower bound the fan in at each layer.
Pretend $\epsilon \in (0, 1/2)$ is constant for simplicity. Let $\alpha = \frac{\ln(1-\epsilon)}{\ln(\epsilon)}$.

1. From Viola [7], clauses have size $\frac{\ln(n)}{\ln(1/\epsilon)}$.
2. If DNFs have size $\tilde{o}(n^{1+\alpha})$, then we can hit every clause with fewer than ϵn variables.
   Thus clauses have size $\tilde{\Omega}(n^{1+\alpha})$.
3. If fewer than $\tilde{o}(n^{2+\alpha})$ clauses, can hit every DNF with fewer than $\frac{n}{\ln(n)^2}$ clauses.
   Thus circuit has $\tilde{\Omega}(n^{2+\alpha})$ clauses.

General Lower Bound Idea

Idea: Same as monotone EXCEPT level 2 bounds might fail.

General Lower Bound Idea

Idea: Same as monotone EXCEPT level 2 bounds might fail.

Issue: Negated variables might make DNF one while fixing adversarial bits.

General Lower Bound Idea

Idea: Same as monotone EXCEPT level 2 bounds might fail.

Issue: Negated variables might make DNF one while fixing adversarial bits.

Solution: Let $\beta = \frac{\ln(1-\epsilon^2)}{2\ln(\epsilon)}$. Fix adversarial bits *probabilistically*.

General Lower Bound Idea

Idea: Same as monotone EXCEPT level 2 bounds might fail.

Issue: Negated variables might make DNF one while fixing adversarial bits.

Solution: Let $\beta = \frac{\ln(1-\epsilon^2)}{2\ln(\epsilon)}$. Fix adversarial bits *probabilistically*.

Result: Likely won't set DNF to one.
Almost definitely will eliminate $\tilde{\Omega}(n^{1+\beta})$ clauses.

Main Tool

Greedy Algorithm for set cover.

Theorem

Let $S = \{S_1, ..., S_m\}$ be subsets of $[n]$ where each $i \in [m]$ has $|S_i| \geq w$. Then for any $t \in [n]$ there is some $T \subseteq [n]$ with $|T| = t$ so that T doesn't intersect with at most

$$me^{w \ln(1 - \frac{t}{n+1})}$$

of the sets in S.

Idea: Just greedily take the variable in the most sets.

Upper Bound

Idea: Amplify promise, iteratively reduce size with promise majority.

1. Use random walks on expander graph to amplify promise to $\frac{1}{\ln(n)^d}$.
   Only increases size by polylogarithmic factor.

Upper Bound

Idea: Amplify promise, iteratively reduce size with promise majority.

1. Use random walks on expander graph to amplify promise to $\frac{1}{\ln(n)^d}$. Only increases size by polylogarithmic factor.
2. Seperate input into groups of size $\tilde{\Omega}\left(n^{\frac{1}{2^d-1}}\right)$. Run depth-3 $\frac{1}{\ln(n)}$-promise majority circuit on each group.

Upper Bound

Idea: Amplify promise, iteratively reduce size with promise majority.

1. Use random walks on expander graph to amplify promise to $\frac{1}{\ln(n)^d}$. Only increases size by polylogarithmic factor.
2. Seperate input into groups of size $\tilde{\Omega}\left(n^{\frac{1}{2^d-1}}\right)$. Run depth-3 $\frac{1}{\ln(n)}$-promise majority circuit on each group.
3. Repeat with appropriate group d times.

Circuit has depth $2+2d$ and size $\tilde{\Omega}\left(n^{\frac{1}{1-2^{-d}}}\right)$.

Uniform Depth-3 Circuits

Best known is Viola's based of derandomization of Lautemann's proof $BPP \subseteq \Sigma_2 \cap \Pi_2$ [5].

Uniform Depth-3 Circuits

Best known is Viola's based of derandomization of Lautemann's proof $BPP \subseteq \Sigma_2 \cap \Pi_2$ [5].

Viola uses $o(\ln(n))$ length walks on expander graphs to get size-$n^{4+o(1)}$, depth-3 circuits for $\frac{1}{\ln(n)}$-promise majority.

Uniform Depth-3 Circuits

Best known is Viola's based of derandomization of Lautemann's proof $BPP \subseteq \Sigma_2 \cap \Pi_2$ [5].

Viola uses $o(\ln(n))$ length walks on expander graphs to get size-$n^{4+o(1)}$, depth-3 circuits for $\frac{1}{\ln(n)}$-promise majority.

We use walks more efficiently to get size-$n^{3+o(1)}$ depth-3 circuits.

Uniform Depth-3 Circuits

Best known is Viola's based of derandomization of Lautemann's proof $BPP \subseteq \Sigma_2 \cap \Pi_2$ [5].

Viola uses $o(\ln(n))$ length walks on expander graphs to get size-$n^{4+o(1)}$, depth-3 circuits for $\frac{1}{\ln(n)}$-promise majority.

We use walks more efficiently to get size-$n^{3+o(1)}$ depth-3 circuits.

We use this circuit to get small uniform upper bounds with more depth.

Monotone Depth-3 Lower Bound

What We Actually Show

Here we prove the simpler lower bounds for constant $\epsilon \in (0, 1/2)$ of:

Monotone

$$n^{2+\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)}$$

General

$$n^{2+\Omega\left(\frac{\epsilon^2}{\ln(1/\epsilon)}\right)}$$

The tighter bounds follow the same ideas with tighter analysis.

Biased Coin Distributions

Definition

Let D_ϵ be the distribution on $\{0,1\}^n$ that sets each bit independently to 1 with probability ϵ.

Biased Coin Distributions

Definition

Let D_ϵ be the distribution on $\{0,1\}^n$ that sets each bit independently to 1 with probability ϵ.

Example: $D_{1/3}$ with 3 coins:

outputs	probabilities
111	$\left(\frac{1}{3}\right)^3$
$011, 101, 110$	$\left(\frac{1}{3}\right)^2 \frac{2}{3}$
$100, 010, 001$	$\left(\frac{1}{3}\right)\left(\frac{2}{3}\right)^2$
000	$\left(\frac{2}{3}\right)^3$

Biased Coin Distributions

Definition

Let D_ϵ be the distribution on $\{0,1\}^n$ that sets each bit independently to 1 with probability ϵ.

Example: $D_{1/3}$ with 3 coins:

outputs	probabilities
111	$\left(\frac{1}{3}\right)^3$
$011, 101, 110$	$\left(\frac{1}{3}\right)^2 \frac{2}{3}$
$100, 010, 001$	$\left(\frac{1}{3}\right) \left(\frac{2}{3}\right)^2$
000	$\left(\frac{2}{3}\right)^3$

By central limit theorem, with probability almost one half, D_ϵ has less than ϵ fraction ones.

Restriction

Definition

We say $\rho \in \{0, 1, *\}^n$ is a restriction on n bits. We say the size of ρ, $|\rho|$, is the number of 1s and 0s in ρ.
If $f : \{0, 1\}^n \to \{0, 1\}$, then define $f \restriction_\rho$ as the function where the values from ρ are passed into f where it is 1 or 0, and otherwise the corresponding variable from the argument is passed in.

Restriction

Definition

We say $\rho \in \{0, 1, *\}^n$ is a restriction on n bits. We say the size of ρ, $|\rho|$, is the number of 1s and 0s in ρ.
If $f : \{0, 1\}^n \to \{0, 1\}$, then define $f \restriction_\rho$ as the function where the values from ρ are passed into f where it is 1 or 0, and otherwise the corresponding variable from the argument is passed in.

Example:

$$\rho = (1, *, 0, *)$$

$$f \restriction_\rho (x_1, x_2) = f(1, x_1, 0, x_2)$$

Restriction

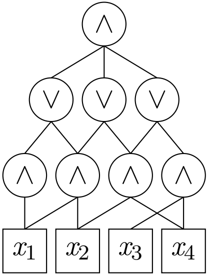
Definition

We say $\rho \in \{0,1,*\}^n$ is a restriction on n bits. We say the size of ρ, $|\rho|$, is the number of 1s and 0s in ρ.
If $f : \{0,1\}^n \to \{0,1\}$, then define $f\restriction_\rho$ as the function where the values from ρ are passed into f where it is 1 or 0, and otherwise the corresponding variable from the argument is passed in.

Example:

$$\rho = (1, *, 0, *)$$

$$f\restriction_\rho(x_1, x_2) = f(1, x_1, 0, x_2)$$



Restriction

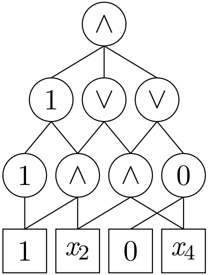
Definition

We say $\rho \in \{0,1,*\}^n$ is a restriction on n bits. We say the size of ρ, $|\rho|$, is the number of 1s and 0s in ρ.
If $f : \{0,1\}^n \to \{0,1\}$, then define $f\restriction_\rho$ as the function where the values from ρ are passed into f where it is 1 or 0, and otherwise the corresponding variable from the argument is passed in.

Example:

$$\rho = (1, *, 0, *)$$

$$f \restriction_\rho (x_1, x_2) = f(1, x_1, 0, x_2)$$



Restriction

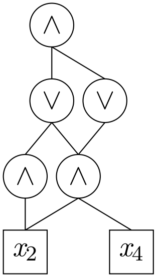
Definition

We say $\rho \in \{0, 1, *\}^n$ is a restriction on n bits. We say the size of ρ, $|\rho|$, is the number of 1s and 0s in ρ.
If $f : \{0, 1\}^n \to \{0, 1\}$, then define $f \restriction_\rho$ as the function where the values from ρ are passed into f where it is 1 or 0, and otherwise the corresponding variable from the argument is passed in.

Example:

$$\rho = (1, *, 0, *)$$

$$f \restriction_\rho (x_1, x_2) = f(1, x_1, 0, x_2)$$



Clause Lower Bound

Viola proved:

Theorem

Suppose that for constant $\epsilon \in (0, 1/2)$, and DNF F that

$$\Pr[F(D_\epsilon) = 0] \geq \textit{poly}(1/n).$$

Then for some $w = \Omega(\frac{\ln(n)}{\ln(1/\epsilon)})$, there is a restriction ρ restricting at most $m = \frac{\epsilon n}{\ln(n)}$ variables so that:

- *Any clause C in F with width less than w has $C \restriction_\rho = 0$.*
- $\Pr[F \restriction_\rho (D_\epsilon) = 0] \geq \Pr[F(D_\epsilon) = 0]$

Eliminates small clauses from a DNF that is likely to output a 0 on D_ϵ with few variables without setting the DNF to 1.

Viola's Clause Lower Bound Idea

For small sets $S_1, ..S_m \subseteq [n]$:

Insight: Maximal independent sets $\sim$ minimal set cover.

Viola's Clause Lower Bound Idea

For small sets $S_1, ..S_m \subseteq [n]$:

Insight: Maximal independent sets $\sim$ minimal set cover.

- Maximal independent $T \implies$ set cover of size $|T|w$.
- Independent $T \implies$ set cover requires size $|T|$.

Viola's Clause Lower Bound Idea

For small sets $S_1, ..S_m \subseteq [n]$:

Insight: Maximal independent sets $\sim$ minimal set cover.

- Maximal independent $T \implies$ set cover of size $|T|w$.
- Independent $T \implies$ set cover requires size $|T|$.

Large Independence: Not possible! Small width on D_ϵ outputs 1 too often.

Viola's Clause Lower Bound Idea

For small sets $S_1, ..S_m \subseteq [n]$:

Insight: Maximal independent sets $\sim$ minimal set cover.

- Maximal independent $T \implies$ set cover of size $|T|w$.
- Independent $T \implies$ set cover requires size $|T|$.

Large Independence: Not possible! Small width on D_ϵ outputs 1 too often.

Small Independence: Fix few variables in small cover to reduce width.

- Choose values to only increase probability of 0.
- Repeat until clause width 0.

Greedy Set Cover

In this talk, we use

Theorem

Let $S = \{S_1, ..., S_m\}$ be subsets of $[n]$ where each $i \in [m]$ has $|S_i| \geq w$. Then for any $t \in [n]$ there is some $T \subseteq [n]$ with $|T| = t$ so that T intersects all but at most

$$|S|e^{-w\frac{t}{n}}$$

of the sets in S.

Closer analysis gives that T intersects all but $|S|e^{-w\ln(1-\frac{t}{n+1})}$ sets.

Greedy Set Cover

In this talk, we use

Theorem

Let $S = \{S_1, ..., S_m\}$ be subsets of $[n]$ where each $i \in [m]$ has $|S_i| \geq w$. Then for any $t \in [n]$ there is some $T \subseteq [n]$ with $|T| = t$ so that T intersects all but at most

$$|S|e^{-w\frac{t}{n}}$$

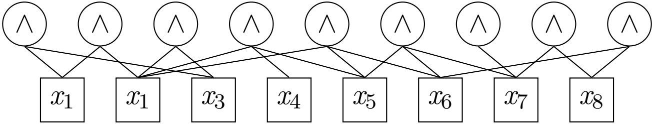
of the sets in S.

Closer analysis gives that T intersects all but $|S|e^{-w\ln(1-\frac{t}{n+1})}$ sets.
In particular, if

- S is the set of clauses in a monotone DNF, F, and
- ρ is some restriction restricting variables in T to 0,

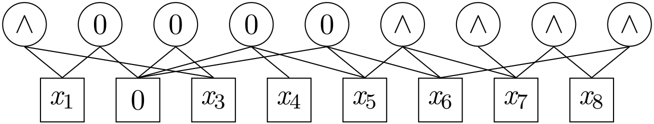
then $|F\restriction_\rho| \leq |F|e^{-w\frac{t}{n}}$ variables remaining.

Greedy Set Cover Proof



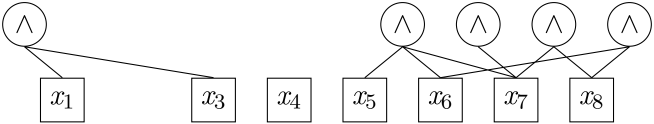
- The average number of sets an element is in is at least $\frac{w|S|}{n}$. So at least one variable, say x_2, is in at least $\frac{w|S|}{n}$ sets.

Greedy Set Cover Proof



- The average number of sets an element is in is at least $\frac{w|S|}{n}$. So at least one variable, say x_2, is in at least $\frac{w|S|}{n}$ sets.

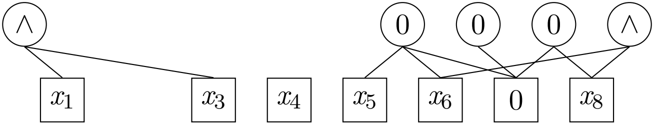
Greedy Set Cover Proof



- The average number of sets an element is in is at least $\frac{w|S|}{n}$. So at least one variable, say x_2, is in at least $\frac{w|S|}{n}$ sets.
- Let S_1 be the sets in S not containing x_1. Then:

$$|S_1| \le |S| - \frac{w}{n}|S| = \left(1 - \frac{w}{n}\right)|S| \le |S|e^{-w/n}.$$

Greedy Set Cover Proof



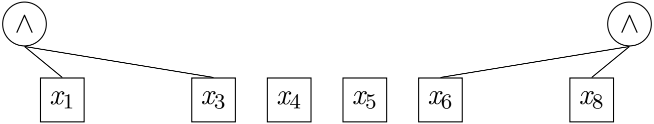
- The average number of sets an element is in is at least $\frac{w|S|}{n}$. So at least one variable, say x_2, is in at least $\frac{w|S|}{n}$ sets.
- Let S_1 be the sets in S not containing x_1. Then:

$$|S_1| \leq |S| - \frac{w}{n}|S| = \left(1 - \frac{w}{n}\right)|S| \leq |S|e^{-w/n}.$$

- Repeat with t times and $S_2, ..., S_t$ to get

$$|S_t| \leq |S| - \frac{w}{n}|S| \leq |S|e^{-\frac{tw}{n}}.$$

Greedy Set Cover Proof



- The average number of sets an element is in is at least $\frac{w|S|}{n}$. So at least one variable, say x_2, is in at least $\frac{w|S|}{n}$ sets.
- Let S_1 be the sets in S not containing x_1. Then:

$$|S_1| \leq |S| - \frac{w}{n}|S| = \left(1 - \frac{w}{n}\right)|S| \leq |S|e^{-w/n}.$$

- Repeat with t times and $S_2, ..., S_t$ to get

$$|S_t| \leq |S| - \frac{w}{n}|S| \leq |S|e^{-\frac{tw}{n}}.$$

- Then these t variables work.

Monotone DNF Size

Theorem

Let $\epsilon \in (0, 1/2)$ *and monotone DNF* F *be suchthat*

- *For all x with less than* ϵn *zeros,* $F(x) = 1$.
- $\Pr[F(D_\epsilon) = 0] \geq$ *poly*$(1/n)$.

Then F has $n^{1+\alpha}$ *clauses for some* $\alpha = \Omega(\frac{\epsilon}{\ln(1/\epsilon)})$.

All DNFs in circuit must satisfy condition 1.

But For DNF to "help" by much, it must satisfy condition 2.

Monotone DNF Size Proof

1. Using Viola's theorem, we fix $\frac{n}{\ln(n)}$ variables and F is left with only clauses longer larger than $w = \Omega(\frac{\ln(n)}{\ln(1/\epsilon)})$.

Monotone DNF Size Proof

1. Using Viola's theorem, we fix $\frac{n}{\ln(n)}$ variables and F is left with only clauses longer larger than $w = \Omega(\frac{\ln(n)}{\ln(1/\epsilon)})$.
2. Using greedy set cover, there is a restriction ρ of $\epsilon n/2$ variables so that ρ makes

$$|F \restriction_\rho| \leq |F| n^{-w \frac{\epsilon n}{2n}} = |F| n^{-\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)}.$$

Monotone DNF Size Proof

1. Using Viola's theorem, we fix $\frac{n}{\ln(n)}$ variables and F is left with only clauses longer larger than $w = \Omega(\frac{\ln(n)}{\ln(1/\epsilon)})$.
2. Using greedy set cover, there is a restriction ρ of $\epsilon n/2$ variables so that ρ makes
$$|F \restriction_\rho| \leq |F| n^{-w \frac{\epsilon n}{2n}} = |F| n^{-\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)}.$$
3. Input can have at least $\frac{\epsilon n}{3}$ more 0s and still be one, so:
$$\frac{\epsilon n}{3} \leq |F \restriction_\rho|.$$

Monotone DNF Size Proof

1. Using Viola's theorem, we fix $\frac{n}{\ln(n)}$ variables and F is left with only clauses longer larger than $w = \Omega(\frac{\ln(n)}{\ln(1/\epsilon)})$.
2. Using greedy set cover, there is a restriction ρ of $\epsilon n/2$ variables so that ρ makes

$$|F \restriction_\rho | \leq |F| n^{-w\frac{\epsilon n}{2n}} = |F| n^{-\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)}.$$

3. Input can have at least $\frac{\epsilon n}{3}$ more 0s and still be one, so:

$$\frac{\epsilon n}{3} \leq |F \restriction_\rho |.$$

4. Together

$$\begin{aligned} \frac{\epsilon n}{3} &\leq |F| n^{-\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)} \\ n^{1+\alpha} &\leq |F|. \end{aligned}$$

Monotone Circuit Size Lower Bounds

Theorem

Depth-3 Circuit C solving ϵ-promise majority has size $n^{2+\Omega(\frac{\epsilon}{\ln(1/\epsilon)})}$.

Idea: Eliminate many DNFs with few clauses.

Can eliminate too many DNFs if there are not enough clauses.

Eliminate All Large Clauses

Theorem

For any Circuit C with $|C| \leq n^c$*, there is a restriction* ρ *restricting* $c\frac{n}{\ln(n)}$ *variables such that* $C \restriction_\rho$ *has no clauses larger than* $\ln(n)^2$*.*

Eliminate All Large Clauses

Theorem

For any Circuit C with $|C| \leq n^c$, there is a restriction ρ restricting $c\frac{n}{\ln(n)}$ variables such that $C \restriction_\rho$ has no clauses larger than $\ln(n)^2$.

Focus on large clauses. Let F' be the DNF with clauses from C bigger than $\ln(n)^2$.

Eliminate All Large Clauses

Theorem

For any Circuit C with $|C| \leq n^c$, there is a restriction ρ restricting $c\frac{n}{\ln(n)}$ variables such that $C \restriction_\rho$ has no clauses larger than $\ln(n)^2$.

Focus on large clauses. Let F' be the DNF with clauses from C bigger than $\ln(n)^2$.

Eliminate with Greedy Cover Algorithm! Fix $c\frac{n}{\ln(n)}$ variables with restriction ρ so that

$$|F' \restriction_\rho| < |F| e^{-\ln(n)^2 \frac{cn}{\ln(n)n}} \leq n^c n^{-c} \leq 1.$$

Eliminate All Large Clauses

Theorem

For any Circuit C with $|C| \leq n^c$, there is a restriction ρ restricting $c\frac{n}{\ln(n)}$ variables such that $C \restriction_\rho$ has no clauses larger than $\ln(n)^2$.

Focus on large clauses. Let F' be the DNF with clauses from C bigger than $\ln(n)^2$.

Eliminate with Greedy Cover Algorithm! Fix $c\frac{n}{\ln(n)}$ variables with restriction ρ so that

$$|F' \restriction_\rho| < |F| e^{-\ln(n)^2 \frac{cn}{\ln(n)n}} \leq n^c n^{-c} \leq 1.$$

Conclude $|F' \restriction_\rho| = 0$, so C has no clauses bigger than $\ln(n)^2$.

Eliminate All Large Clauses

Theorem

For any Circuit C with $|C| \leq n^c$, there is a restriction ρ restricting $c\frac{n}{\ln(n)}$ variables such that $C \restriction_\rho$ has no clauses larger than $\ln(n)^2$.

Focus on large clauses. Let F' be the DNF with clauses from C bigger than $\ln(n)^2$.

Eliminate with Greedy Cover Algorithm! Fix $c\frac{n}{\ln(n)}$ variables with restriction ρ so that

$$|F' \restriction_\rho| < |F| e^{-\ln(n)^2 \frac{cn}{\ln(n)n}} \leq n^c n^{-c} \leq 1.$$

Conclude $|F' \restriction_\rho| = 0$, so C has no clauses bigger than $\ln(n)^2$.

NOTE: Similar algorithm works if the clauses are non-monotone, but must generalize theorem.

Monotone Lower Bound Final Ingredient

Simple version of final step in circuit lower bound.

Theorem

If F is a monotone DNF with clause width m^{1+x} for constant $x > 0$, $|F| =$ poly(n) and such that F computes "OR", then F must have $n \geq \tilde{\Omega}(m^{2+x})$.

Monotone Lower Bound Final Ingredient

Simple version of final step in circuit lower bound.

Theorem

If F is a monotone DNF with clause width m^{1+x} for constant $x > 0$, $|F| = poly(n)$ and such that F computes "OR", then F must have $n \geq \tilde{\Omega}(m^{2+x})$.

1. Use greedy set cover to get a restriction ρ restricting m variables such that:
$$|F \restriction_\rho| \leq |F| e^{-m^{1+x} \frac{m}{n}} = |F| e^{-m^{2+x} \frac{1}{n}}$$

Monotone Lower Bound Final Ingredient

Simple version of final step in circuit lower bound.

Theorem

If F is a monotone DNF with clause width m^{1+x} for constant $x > 0$, $|F| = \text{poly}(n)$ and such that F computes "OR", then F must have $n \geq \tilde{\Omega}(m^{2+x})$.

1. Use greedy set cover to get a restriction ρ restricting m variables such that:

$$|F \restriction_\rho| \leq |F| e^{-m^{1+x}\frac{m}{n}} = |F| e^{-m^{2+x}\frac{1}{n}}$$

2. See that $m < m^{1+x} \leq n$. So $F \restriction_\rho \neq 0$, and $|F \restriction_\rho| \geq 1$.

Monotone Lower Bound Final Ingredient

Simple version of final step in circuit lower bound.

Theorem

If F is a monotone DNF with clause width m^{1+x} for constant $x > 0$, $|F| =$ poly(n) and such that F computes "OR", then F must have $n \geq \tilde{\Omega}(m^{2+x})$.

1. Use greedy set cover to get a restriction ρ restricting m variables such that:
$$|F \restriction_\rho| \leq |F| e^{-m^{1+x} \frac{m}{n}} = |F| e^{-m^{2+x} \frac{1}{n}}$$
2. See that $m < m^{1+x} \leq n$. So $F \restriction_\rho \neq 0$, and $|F \restriction_\rho| \geq 1$.
3. Together:
$$\begin{aligned} 1 \leq& |F| e^{-m^{2+x} \frac{1}{n}} \\ \tilde{\Omega}(m^{2+x}) \leq& n \end{aligned}$$

Monoton Circuit Lower Bound Proof Idea

- Remove Large Clauses.
- Use DNF lower bounds to get each cause bigger than $n^{1+\alpha}$.
- Fix whole clauses with the idea from the previous slide to lower bound number of clauses.

Monoton Circuit Lower Bound Proof Idea

- Remove Large Clauses.
- Use DNF lower bounds to get each cause bigger than $n^{1+\alpha}$.
- Fix whole clauses with the idea from the previous slide to lower bound number of clauses.

  Issue: Some DNFs might be small.

Monoton Circuit Lower Bound Proof Idea

- Remove Large Clauses.
- Use DNF lower bounds to get each cause bigger than $n^{1+\alpha}$.
- Fix whole clauses with the idea from the previous slide to lower bound number of clauses.

Issue: Some DNFs might be small.

Solution: Focus on large DNFs during elimination.

Insight: Some large DNF must survive if few variables fixed.

Monotone Circuit Lower Bound Proof

Let C be a circuit solving ϵ-promise majority.

- Remove all clauses larger than $\ln(n)^2$ with a restriction ρ_1 which restricts $O\left(\frac{n}{\ln(n)}\right)$ variables.

Monotone Circuit Lower Bound Proof

Let C be a circuit solving ϵ-promise majority.

- Remove all clauses larger than $\ln(n)^2$ with a restriction ρ_1 which restricts $O\left(\frac{n}{\ln(n)}\right)$ variables.
- Let C' be $C \restriction_{\rho_1}$ only including DNFs with size at least $n^{1+\alpha}$ for $\alpha = \Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)$ from our DNF size lower bounds.

Monotone Circuit Lower Bound Proof

Let C be a circuit solving ϵ-promise majority.

- Remove all clauses larger than $\ln(n)^2$ with a restriction ρ_1 which restricts $O\left(\frac{n}{\ln(n)}\right)$ variables.
- Let C' be $C \restriction_{\rho_1}$ only including DNFs with size at least $n^{1+\alpha}$ for $\alpha = \Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)$ from our DNF size lower bounds.
- Use greedy set cover algorithm to select $\frac{n}{\ln(n)^3}$ clauses and set them to one in ρ_2 so that

$$\| C' \restriction_{\rho_2} \| \leq \| C' \| e^{-n^{1+\alpha} \frac{n}{\ln(n)^3 |C'|}}.$$

Monotone Circuit Lower Bound Proof

Let C be a circuit solving ϵ-promise majority.

- Remove all clauses larger than $\ln(n)^2$ with a restriction ρ_1 which restricts $O\left(\frac{n}{\ln(n)}\right)$ variables.
- Let C' be $C\restriction_{\rho_1}$ only including DNFs with size at least $n^{1+\alpha}$ for $\alpha = \Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)$ from our DNF size lower bounds.
- Use greedy set cover algorithm to select $\frac{n}{\ln(n)^3}$ clauses and set them to one in ρ_2 so that

$$\|C'\restriction_{\rho_2}\| \leq \|C'\| e^{-n^{1+\alpha}\frac{n}{\ln(n)^3 |C'|}}.$$

- See that $C\restriction_{\rho_1}\restriction_{\rho_2}$ still solves $\left(\epsilon - O\left(\frac{1}{\ln(n)}\right)\right)$-promise majority. If $\|C\| \leq n^3$, by a counting argument some DNF, F, must have $\Pr[F(D_\epsilon) = 0] \geq \text{poly}(1/n)$. Thus, $\|C'\restriction_{\rho_2}\| \geq 1$.

Monotone Circuit Lower Bound Proof

Let C be a circuit solving ϵ-promise majority.

- Remove all clauses larger than $\ln(n)^2$ with a restriction ρ_1 which restricts $O\left(\frac{n}{\ln(n)}\right)$ variables.
- Let C' be $C\restriction_{\rho_1}$ only including DNFs with size at least $n^{1+\alpha}$ for $\alpha = \Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)$ from our DNF size lower bounds.
- Use greedy set cover algorithm to select $\frac{n}{\ln(n)^3}$ clauses and set them to one in ρ_2 so that

$$\|C'\restriction_{\rho_2}\| \leq \|C'\| e^{-n^{1+\alpha}\frac{n}{\ln(n)^3|C'|}}.$$

- See that $C\restriction_{\rho_1}\restriction_{\rho_2}$ still solves $\left(\epsilon - O\left(\frac{1}{\ln(n)}\right)\right)$-promise majority. If $\|C\| \leq n^3$, by a counting argument some DNF, F, must have $\Pr[F(D_\epsilon) = 0] \geq$ poly$(1/n)$. Thus,$\|C'\restriction_{\rho_2}\| \geq 1$.
- Together:

$$\tilde{\Omega}(n^{2+\alpha}) = n^{2+\Omega\left(\frac{\epsilon}{\ln(1/\epsilon)}\right)} \leq |C'|.$$

General Depth-3 Lower Bounds

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

- Clause lower bounds, works!

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

- Clause lower bounds, works!
- DNF lower bounds, *almost* works.

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

- Clause lower bounds, works!
- DNF lower bounds, *almost* works.

  Following first proof, may set DNF to one early due to negations.
  Then, can't argue restriction left any clauses.
  Will fix next.

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

- Clause lower bounds, works!
- DNF lower bounds, *almost* works.

  Following first proof, may set DNF to one early due to negations. Then, can't argue restriction left any clauses.
  Will fix next.
- Circuit lower bounds, works!

Non Monotone Lower Bound Overview

Monotone Idea: Bound size at each level, using restrictions from set cover algorithm.

General Idea: Same!

- Clause lower bounds, works!
- DNF lower bounds, *almost* works.

  Following first proof, may set DNF to one early due to negations. Then, can't argue restriction left any clauses.
  Will fix next.
- Circuit lower bounds, works!
  - At worst, might eliminate or shrink DNFs and clauses early.
  - But circuit still solves a promise problem, so it still has large DNFs after restriction.

Probabilistic Restriction

Main Lemma

Lemma

For constant $\epsilon \in (0, 1/2)$, let F be a DNF with:

- *For all x with less than ϵn zeros, $F(x) = 1$.*
- $\Pr[F(D_\epsilon) = 0] \geq$ *poly*$(1/n)$

Let $\beta = \Omega(\frac{\epsilon^2}{\ln(1/\epsilon)})$. Then there is a random variable ρ which is a restriction on $\epsilon n/2$ variables such that:

- $F(D_\epsilon) = F \restriction_\rho (D_\epsilon)$.
- *Let F' be the DNF with clauses in $F \restriction_\rho$ bigger than $w = \Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$. Then: $\Pr[|F'| > |F|n^{-\beta}] \leq e^{-\Omega(n)}$.*

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

- Each restriction in the sequence adds one more restriction, sampled from D_ϵ.

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

- Each restriction in the sequence adds one more restriction, sampled from D_ϵ.
- Each restriction has a good chance of eliminating many clauses.

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

- Each restriction in the sequence adds one more restriction, sampled from D_ϵ.
- Each restriction has a good chance of eliminating many clauses.
- Focuses on deleting clauses bigger then w.

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

- Each restriction in the sequence adds one more restriction, sampled from D_ϵ.
- Each restriction has a good chance of eliminating many clauses.
- Focuses on deleting clauses bigger then w.

Use greedy set cover algorithm to choose variables like monotone case.

Instead of just setting them to 0, we set them to 1 with probability ϵ.

Probabilistic Restriction Idea

Idea: Define sequence of restrictions, each restricting one more variable such that:

- Each restriction in the sequence adds one more restriction, sampled from D_ϵ.
- Each restriction has a good chance of eliminating many clauses.
- Focuses on deleting clauses bigger then w.

Use greedy set cover algorithm to choose variables like monotone case.

Instead of just setting them to 0, we set them to 1 with probability ϵ.

Then by Chernoff bounds, its likely that we eliminate many clauses.

And by definition if we restrict the rest of the variables, it is the same as using D_ϵ.

Probabilistic Restriction Construction

First, define sequence of DNFs $F_1, ..., F_m$, and restrictions $\rho_0, ..., \rho_m$ for $m = \epsilon n/2$.

1. Let F_1 be the DNF only including clauses from F with width larger than $w = \Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$ from the clause lower bound.
   Let ρ_0 restrict nothing.

Probabilistic Restriction Construction

First, define sequence of DNFs $F_1, ..., F_m$, and restrictions $\rho_0, ..., \rho_m$ for $m = \epsilon n/2$.

1. Let F_1 be the DNF only including clauses from F with width larger than $w = \Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$ from the clause lower bound.
   Let ρ_0 restrict nothing.
2. There is some variable that is in at least $\frac{w|F_i|}{n}$ clauses of F_i, x_i.

   Let ρ_i be the restriction restricting ρ_{i-1} plus restricting x_i to one with probability ϵ, and 0 otherwise.

Probabilistic Restriction Construction

First, define sequence of DNFs $F_1, ..., F_m$, and restrictions $\rho_0, ..., \rho_m$ for $m = \epsilon n/2$.

1. Let F_1 be the DNF only including clauses from F with width larger than $w = \Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$ from the clause lower bound.
   Let ρ_0 restrict nothing.
2. There is some variable that is in at least $\frac{w|F_i|}{n}$ clauses of F_i, x_i.

   Let ρ_i be the restriction restricting ρ_{i-1} plus restricting x_i to one with probability ϵ, and 0 otherwise.
3. Define F_i to be the DNF which has the clauses in $F \restriction_{\rho_{i-1}}$ that have width greater than w.

Probabilistic Restriction Construction

First, define sequence of DNFs $F_1, ..., F_m$, and restrictions $\rho_0, ..., \rho_m$ for $m = \epsilon n/2$.

1. Let F_1 be the DNF only including clauses from F with width larger than $w = \Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$ from the clause lower bound.
   Let ρ_0 restrict nothing.
2. There is some variable that is in at least $\frac{w|F_i|}{n}$ clauses of F_i, x_i.

   Let ρ_i be the restriction restricting ρ_{i-1} plus restricting x_i to one with probability ϵ, and 0 otherwise.
3. Define F_i to be the DNF which has the clauses in $F \restriction_{\rho_{i-1}}$ that have width greater than w.

Then $\rho = \rho_m$, and F' is the DNF with clauses from $F_m \restriction_\rho$ bigger than w. See that $F \restriction_{\rho_m} (D_\epsilon) = F(D_\epsilon)$.

Probabilistic Restriction Analysis

At step i, either x_i or $\neg x_i$ is in at least is in $\frac{w|F_i|}{2n}$ clauses.

Probabilistic Restriction Analysis

At step i, either x_i or $\neg x_i$ is in at least is in $\frac{w|F_i|}{2n}$ clauses.
There is at least an ϵ chance of successfully eliminating $\frac{w|F_i|}{2n}$ clauses.

Probabilistic Restriction Analysis

At step i, either x_i or $\neg x_i$ is in at least is in $\frac{w|F_i|}{2n}$ clauses.
There is at least an ϵ chance of successfully eliminating $\frac{w|F_i|}{2n}$ clauses.
If k steps succeed, then

$$|F_m \restriction_{\rho_m}| \leq (1 - \frac{w}{2n})^k |F| \leq |F| e^{-\frac{wk}{2n}} \leq |F| n^{-\Omega\left(\frac{k}{\ln(1/\epsilon) n}\right)}.$$

Probabilistic Restriction Analysis

At step i, either x_i or $\neg x_i$ is in at least is in $\frac{w|F_i|}{2n}$ clauses.
There is at least an ϵ chance of successfully eliminating $\frac{w|F_i|}{2n}$ clauses.
If k steps succeed, then

$$|F_m \restriction_{\rho_m}| \leq (1 - \frac{w}{2n})^k |F| \leq |F| e^{-\frac{wk}{2n}} \leq |F| n^{-\Omega\left(\frac{k}{\ln(1/\epsilon)n}\right)}.$$

By Chernoff bound,

$$\Pr[k < \epsilon m/2] \leq e^{-\Omega(n)}.$$

Probabilistic Restriction Analysis

At step i, either x_i or $\neg x_i$ is in at least is in $\frac{w|F_i|}{2n}$ clauses.
There is at least an ϵ chance of successfully eliminating $\frac{w|F_i|}{2n}$ clauses.
If k steps succeed, then

$$|F_m \restriction_{\rho_m}| \leq (1 - \frac{w}{2n})^k |F| \leq |F| e^{-\frac{wk}{2n}} \leq |F| n^{-\Omega\left(\frac{k}{\ln(1/\epsilon)n}\right)}.$$

By Chernoff bound,

$$\Pr[k < \epsilon m/2] \leq e^{-\Omega(n)}.$$

Thus

$$\Pr[|F_m \restriction_{\rho_m}| > |F| n^{-\Omega\left(\frac{\epsilon m}{\ln(1/\epsilon)n}\right)}] \leq e^{-\Omega(n)}$$
$$\Pr[|F'| > |F| n^{-\Omega\left(\frac{\epsilon^2}{\ln(1/\epsilon)}\right)}] \leq e^{-\Omega(n)}.$$

Applying Restriction To Get DNF Bounds

1. Apply probabilistic restriction to get ρ, F' with
$$\Pr[|F'| > |F|n^{-\beta}] \leq e^{-\Omega(n)}.$$

Applying Restriction To Get DNF Bounds

1. Apply probabilistic restriction to get ρ, F' with
$$\Pr[|F'| > |F|n^{-\beta}] \leq e^{-\Omega(n)}.$$

2. By assumption, $\Pr[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)$. Thus:
$$\Pr_\rho[\Pr_{D_\epsilon}[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)] \geq 1/\text{poly}(n).$$

Applying Restriction To Get DNF Bounds

1. Apply probabilistic restriction to get ρ, F' with
$$\Pr[|F'| > |F|n^{-\beta}] \leq e^{-\Omega(n)}.$$

2. By assumption, $\Pr[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)$. Thus:
$$\Pr_{\rho}[\Pr_{D_\epsilon}[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)] \geq 1/\text{poly}(n).$$

3. $1/\text{poly}(n) > e^{-\Omega(n)}$, so some ρ' restricts $\epsilon n/2$ variables such that
$$\Pr[F \restriction_{\rho'} (D_\epsilon) = 0] \geq 1/\text{poly}(n),$$
$$|F'| < |F|e^{-\beta}.$$

Applying Restriction To Get DNF Bounds

1. Apply probabilistic restriction to get ρ, F' with
$$\Pr[|F'| > |F|n^{-\beta}] \leq e^{-\Omega(n)}.$$

2. By assumption, $\Pr[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)$. Thus:
$$\Pr_\rho[\Pr_{D_\epsilon}[F \restriction_\rho (D_\epsilon) = 0] \geq 1/\text{poly}(n)] \geq 1/\text{poly}(n).$$

3. $1/\text{poly}(n) > e^{-\Omega(n)}$, so some ρ' restricts $\epsilon n/2$ variables such that
$$\begin{aligned} \Pr[F \restriction_{\rho'} (D_\epsilon) = 0] &\geq 1/\text{poly}(n), \\ |F'| &< |F|e^{-\beta}. \end{aligned}$$

4. $F \restriction_{\rho'}$ has $\Omega(\epsilon n)$ clauses with width $\Omega\left(\frac{\ln(n)}{\ln(1/\epsilon)}\right)$: $|F'| \geq \Omega(\epsilon n)$. Thus:
$$\begin{aligned} \Omega(\epsilon n) \leq |F'| &\leq |F|n^{-\beta} \\ n^{1+\beta} &\leq |F|. \end{aligned}$$

General Circuit Lower Bounds

Use the same argument as the monotone DNF, with the lower bounds of $n^{1+\beta}$ on the second level.

Upper Bounds

Depth-3 Upper Bounds

Upper bounds use depth-3 circuits as a subroutine.

For constant ϵ, we use:

Existential: constant ϵ: Ajtai gave size $O\left(n^{2+\frac{\ln(1-\epsilon)}{\ln(\epsilon)-\ln(1-\epsilon)}}\right)$.

For $\epsilon = \frac{1}{\ln(n)}$, simplifies to $O\left(n^2\right)$.

Depth-3 Upper Bounds

Upper bounds use depth-3 circuits as a subroutine.

For constant ϵ, we use:

Existential: constant ϵ: Ajtai gave size $O\left(n^{2+\frac{\ln(1-\epsilon)}{\ln(\epsilon)-\ln(1-\epsilon)}}\right)$.

For $\epsilon = \frac{1}{\ln(n)}$, simplifies to $O\left(n^2\right)$.

P-Uniform: $\epsilon = \frac{1}{\ln(n)}$: Viola gives $n^{4+o(1)}$.

In appendix, we improve the circuit to get size $n^{3+o(1)}$.

Depth-3 Upper Bounds

Upper bounds use depth-3 circuits as a subroutine.
For constant ϵ, we use:

Existential: constant ϵ: Ajtai gave size $O\left(n^{2+\frac{\ln(1-\epsilon)}{\ln(\epsilon)-\ln(1-\epsilon)}}\right)$.

For $\epsilon = \frac{1}{\ln(n)}$, simplifies to $O\left(n^2\right)$.

P-Uniform: $\epsilon = \frac{1}{\ln(n)}$: Viola gives $n^{4+o(1)}$.

In appendix, we improve the circuit to get size $n^{3+o(1)}$.

Reminder, Idea: Amplify, recursively apply promise majority.

Amplification

- Easy to amplify constant ϵ promise to $1/\text{poly}(n)$ promise with depth-2 circuit.

Amplification

- Easy to amplify constant ϵ promise to $1/\text{poly}(n)$ promise with depth-2 circuit.

  Idea: Take majority of short walks on expander graphs (Used by Viola for depth-3 circuit).

  How: Short DNFs: majority of $O(\ln(\ln(n)))$ bits has polylogarithmic-size DNF.

  Chernoff: Expander Chernoff bound proves amplification

  Motivation $1/\ln(n)$-promise majority is easier.

  Results $o(n^2)$-sized depth-4 circuits using careful analysis of Ajtai's.

Amplification

- Easy to amplify constant ϵ promise to $1/\text{poly}(n)$ promise with depth-2 circuit.
  - Idea: Take majority of short walks on expander graphs (Used by Viola for depth-3 circuit).
  - How: Short DNFs: majority of $O(\ln(\ln(n)))$ bits has polylogarithmic-size DNF.
  - Chernoff: Expander Chernoff bound proves amplification
  - Motivation $1/\ln(n)$-promise majority is easier.
  - Results $o(n^2)$-sized depth-4 circuits using careful analysis of Ajtai's.
- Can we get very small promise majority with just amplification and a single depth-3 promise majority?

Amplification

- Easy to amplify constant ϵ promise to $1/\text{poly}(n)$ promise with depth-2 circuit.

  Idea: Take majority of short walks on expander graphs (Used by Viola for depth-3 circuit).

  How: Short DNFs: majority of $O(\ln(\ln(n)))$ bits has polylogarithmic-size DNF.

  Chernoff: Expander Chernoff bound proves amplification

  Motivation $1/\ln(n)$-promise majority is easier.

  Results $o(n^2)$-sized depth-4 circuits using careful analysis of Ajtai's.

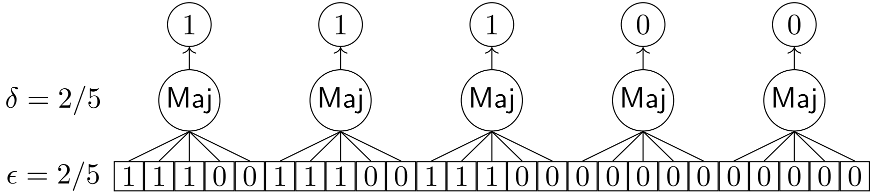
- Can we get very small promise majority with just amplification and a single depth-3 promise majority?
  - Not without much better amplification!
  - Existing techniques increase size faster than promise, so that depth-3 promise majority circuits solving promise majority are still 'large'.

Iteratively Computing Majority Idea

$\epsilon = 2/5$ $\boxed{1}\boxed{1}\boxed{1}\boxed{0}\boxed{0}\boxed{1}\boxed{1}\boxed{1}\boxed{0}\boxed{0}\boxed{1}\boxed{1}\boxed{1}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}\boxed{0}$

Since amplification increases size too fast, decrease size.

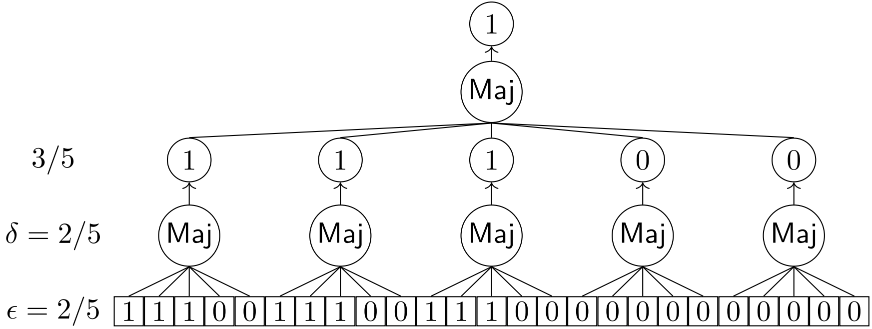
Iterative Computing Majority Idea



Since amplification increases size too fast, decrease size.

Idea: Run promise majority on small groups to get new bits.

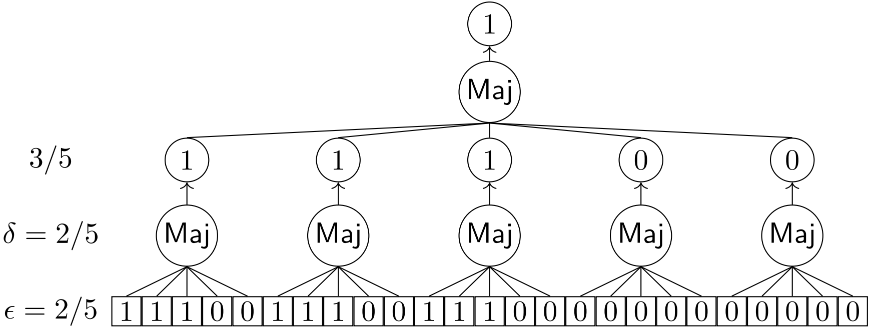
Iteratively Computing Majority Idea



Since amplification increases size too fast, decrease size.

Idea: Run promise majority on small groups to get new bits.
Problem: For large ϵ, may violate promise.

Iteratively Computing Majority Idea



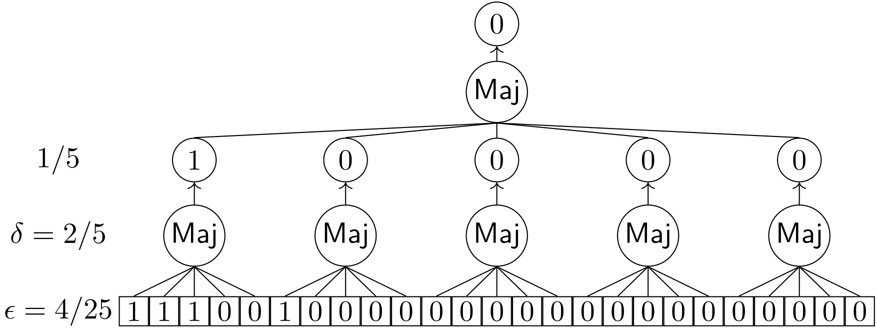
Since amplification increases size too fast, decrease size.

Idea: Run promise majority on small groups to get new bits.

Problem: For large ϵ, may violate promise.

Insight: ϵ-promise input ran in groups through δ-promise circuits gives $\frac{\epsilon}{\delta}$-promise input.

Iterative Computing Majority Idea



Since amplification increases size too fast, decrease size.

Idea: Run promise majority on small groups to get new bits.

Problem: For large ϵ, may violate promise.

Insight: ϵ-promise input ran in groups through δ-promise circuits gives $\frac{\epsilon}{\delta}$-promise input.

Solution: Amplify, then run in groups.

Iterately Computing Majority

Using this idea:

Theorem

If there are depth-3 circuits with size n^α solving $\frac{1}{\ln(n)}$-promise majority, then for any positive integer k, there are depth-$(1+2k)$ circuits solving $\frac{1}{\ln(n)^k}$-promise majority with size

$$kn^{\frac{1}{1-\left(\frac{\alpha-1}{\alpha}\right)^k}}.$$

Combined with depth 2 amplification, we get our upper bounds for higher depths.

Sub-Quadratic Size Promise Majority

As special cases, we get, using Ajtai's circuit, we get:

Theorem

There exists a depth-4 circuits computing ϵ-promise majority with size $o(n^2)$.

And using our circuit, we get:

Theorem

There exists a depth-6, P-Uniform circuits computing ϵ-promise majority with size $o(n^2)$.

Viola's original circuit needed depth-8 to get sub-quadratic size.

Open Problems

Open Problems

- Wanted fine grained tradeoff in depth vs size during derandomization. Particularly, if quadratic derandomization costs depth-3.

Open Problems

- Wanted fine grained tradeoff in depth vs size during derandomization. Particularly, if quadratic derandomization costs depth-3.
  - Did show existing derandomization techniques have this.
  - Might not be only way to derandomize. Need to find explicit problem OR find a new way to derandomize.

Open Problems

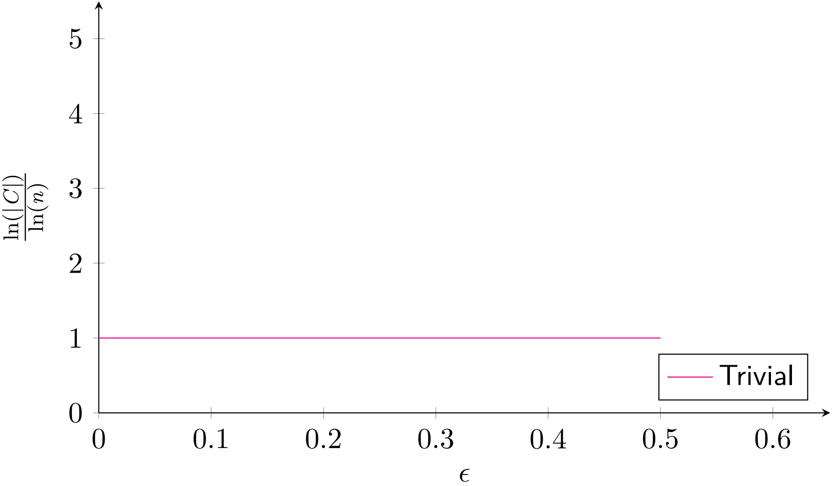
- Wanted fine grained tradeoff in depth vs size during derandomization. Particularly, if quadratic derandomization costs depth-3.
  - Did show existing derandomization techniques have this.
  - Might not be only way to derandomize. Need to find explicit problem OR find a new way to derandomize.
- Missing explicit, depth-4 quadratic sized circuits.
  - Seems related to other psuedorandom objects. Can be rephrased as distribution over dispersers.

Open Problems

- Wanted fine grained tradeoff in depth vs size during derandomization. Particularly, if quadratic derandomization costs depth-3.
  - Did show existing derandomization techniques have this.
  - Might not be only way to derandomize. Need to find explicit problem OR find a new way to derandomize.
- Missing explicit, depth-4 quadratic sized circuits.
  - Seems related to other psuedorandom objects. Can be rephrased as distribution over dispersers.
- Results aren't tight!
  - Upper and lower bounds don't match.
  - Are the best circuits monotone?
  - Do any uniform circuits have optimal size?
  - Upper bounds for large depth don't match known lower bounds (Chaudhuri and Radhakrishnan are asymptotically close [4]).

Depth-3 Bounds, Constant ϵ

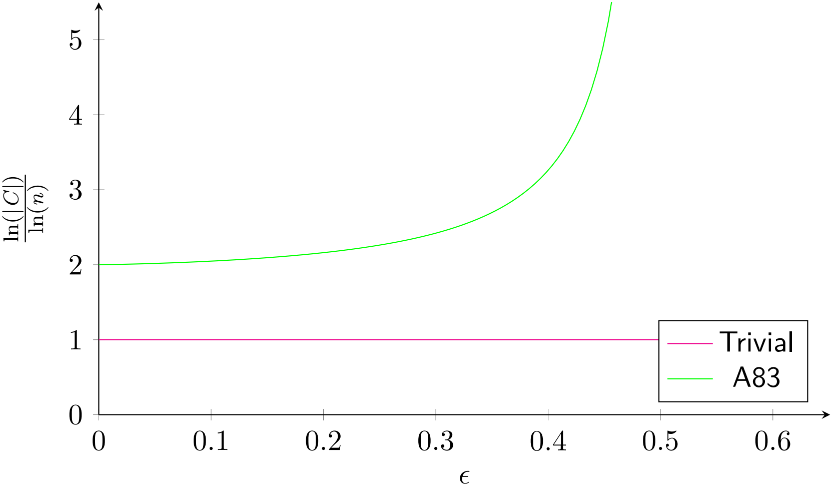
Note: Graphs *slightly* adjusted for visibility.
Y-axis is c for circuit size n^c.



Depth-3 Bounds, Constant ϵ

Note: Graphs *slightly* adjusted for visibility.

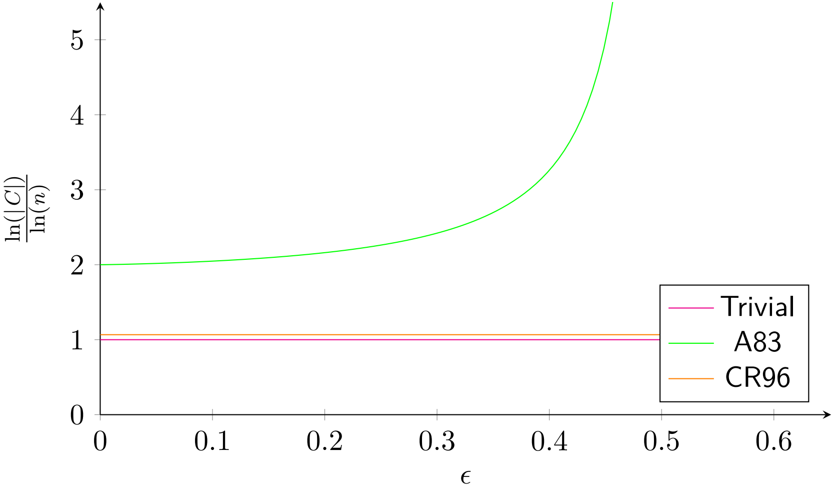
Y-axis is c for circuit size n^c.



Depth-3 Bounds, Constant ϵ

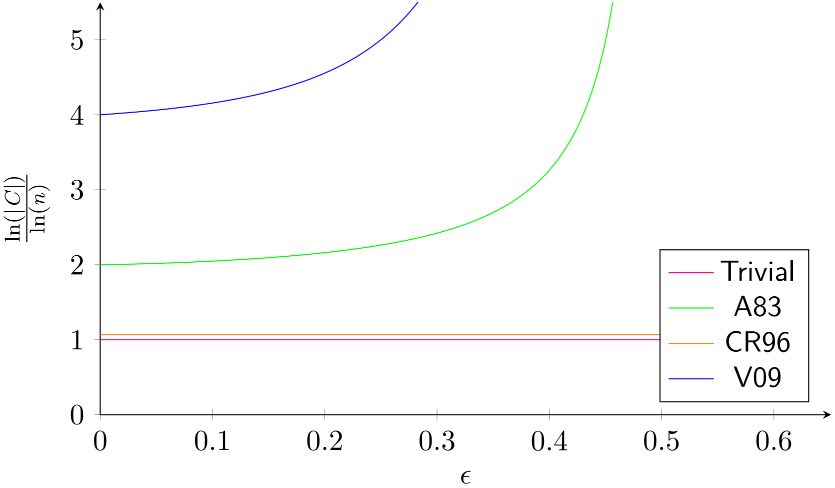
Note: Graphs *slightly* adjusted for visibility.

Y-axis is c for circuit size n^c.



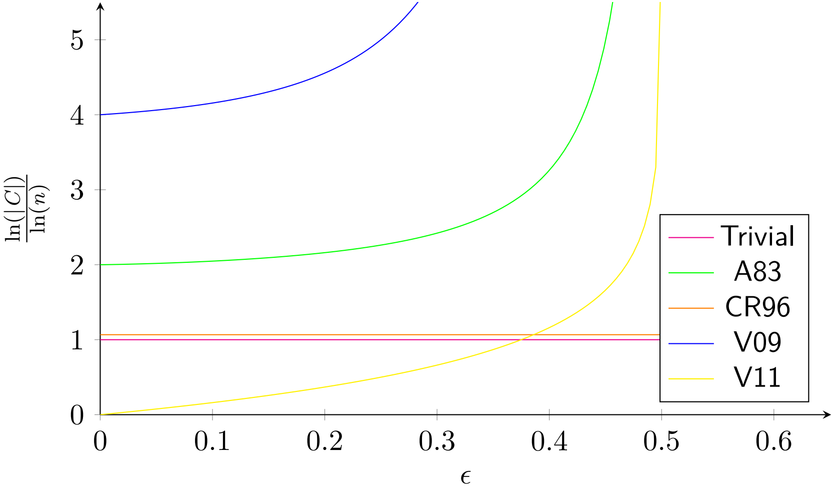
Depth-3 Bounds, Constant ϵ

Note: Graphs *slightly* adjusted for visibility.
Y-axis is c for circuit size n^c.



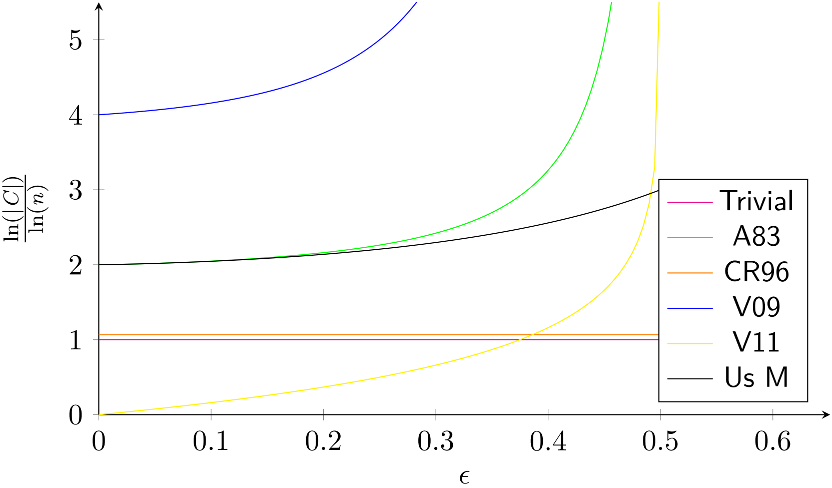
Depth-3 Bounds, Constant ϵ

Note: Graphs *slightly* adjusted for visibility.
Y-axis is c for circuit size n^c.



Depth-3 Bounds, Constant ϵ

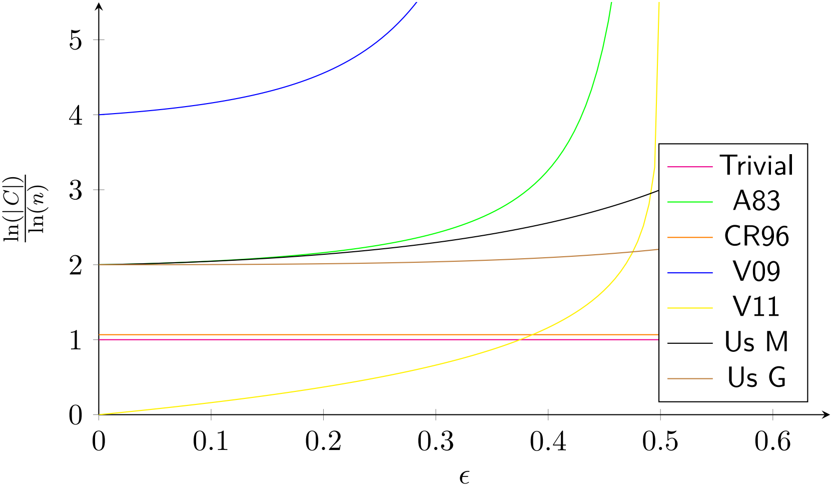
Note: Graphs *slightly* adjusted for visibility.
Y-axis is c for circuit size n^c.



Depth-3 Bounds, Constant ϵ

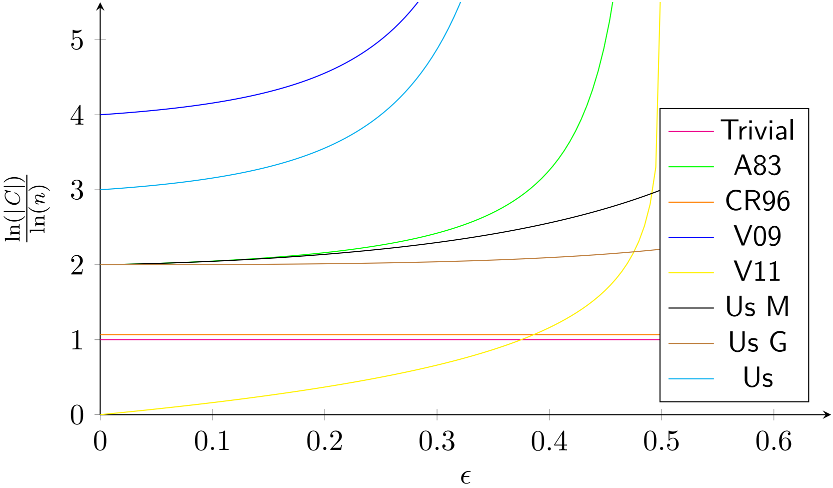
Note: Graphs *slightly* adjusted for visibility.

Y-axis is c for circuit size n^c.



Note: Graphs *slightly* adjusted for visibility.

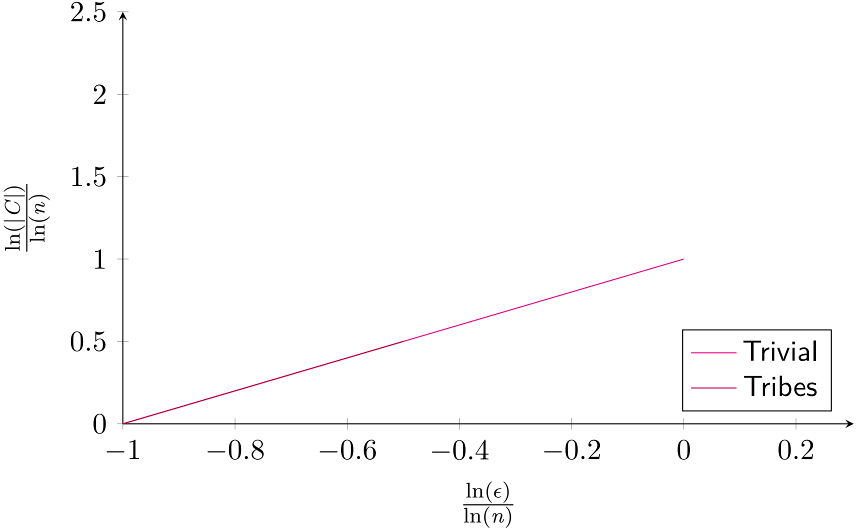
Y-axis is c for circuit size n^c.



Depth-3 Bounds, Small ϵ

Note: *now* might have many more wires than gates.

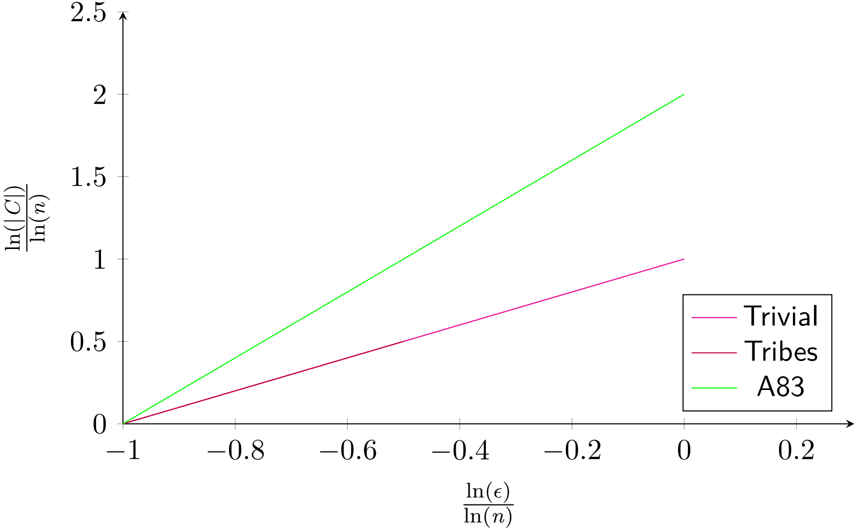
X-axis is c if $\epsilon = n^c$.



Depth-3 Bounds, Small ϵ

Note: *now* might have many more wires than gates.

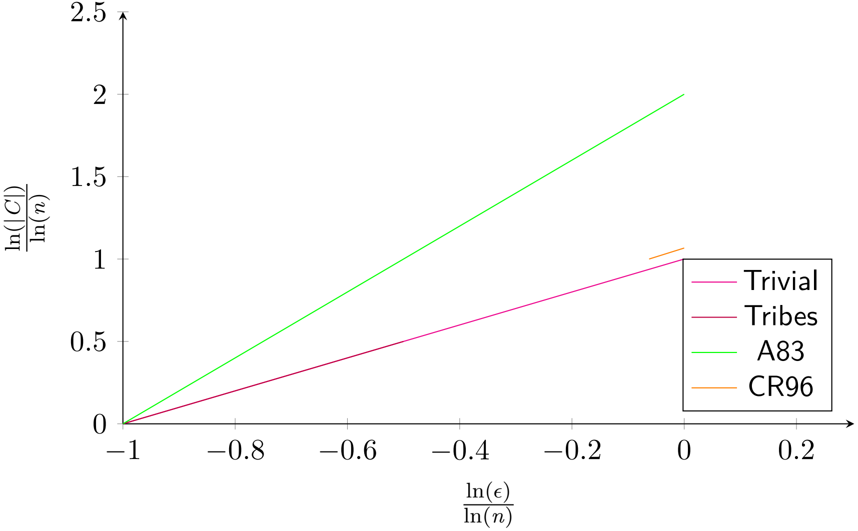
X-axis is c if $\epsilon = n^c$.



Depth-3 Bounds, Small ϵ

Note: *now* might have many more wires than gates.

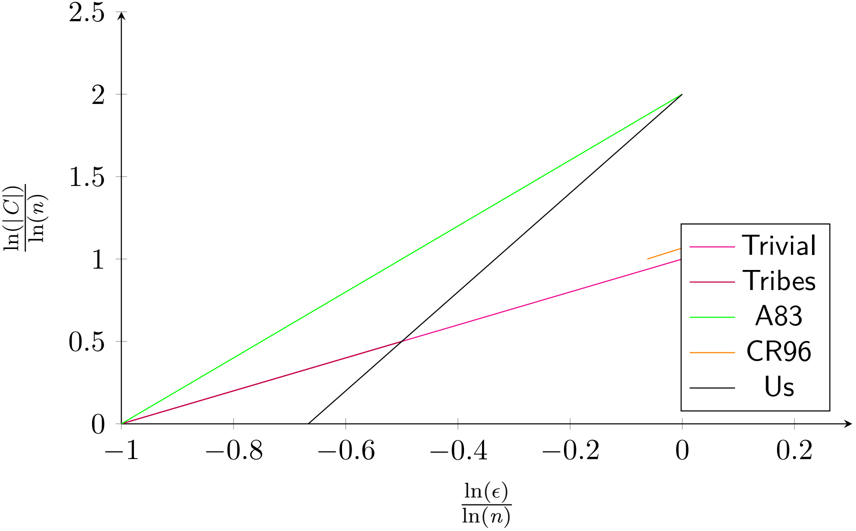
X-axis is c if $\epsilon = n^c$.



Depth-3 Bounds, Small ϵ

Note: *now* might have many more wires than gates.

X-axis is c if $\epsilon = n^c$.



References

References I

Leonard Adleman.
Two theorems on random polynomial time.
In *Proceedings of the 19th Annual Symposium on Foundations of Computer Science*, SFCS '78, page 75–83, USA, 1978. IEEE Computer Society.

Miklós Ajtai.
Sigma11-formulae on finite structures.
Ann. Pure Appl. Log., 24:1–48, 1983.

Miklós Ajtai.
Approximate counting with uniform constant-depth circuits.
In *Advances In Computational Complexity Theory*, volume 13, pages 1–20, 1993.

References II

Shiva Chaudhuri and Jaikumar Radhakrishnan.
Deterministic restrictions in circuit complexity.
In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 30–36, New York, NY, USA, 1996. Association for Computing Machinery.

Clemens Lautemann.
Bpp and the polynomial hierarchy.
Information Processing Letters, 17(4):215 – 217, 1983.

Nutan Limaye, Srikanth Srinivasan, and Utkarsh Tripathi.
More on $\mathrm{AC}^0[\oplus]$ and variants of the majority function.
In *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2019)*, volume 150, pages 22:1–22:14, 2019.

Emanuele Viola.
On approximate majority and probabilistic time.
Computational Complexity, 18:337–375, 2009.

Emanuele Viola.
Randomness buys depth for approximate counting.
In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 230–239, 2011.