Using Hastad's Switching Lemma to prove Parity $\notin AC^0$

Govind Balaji S | CS18BTECH11015
Vedant Singh | CS18BTECH11047
Shreyas Jayant Havaldar | CS18BTECH11042

Circuit Complexity

December 9, 2020

Definition

A restriction ρ is a mapping from $\{1, 2, ..., n\} \longrightarrow \{0, 1, *\}$. Given a function ϕ and a restriction ρ, the function restricted by ρ, denoted as $\phi|_\rho$ is defined as $\phi|_\rho(\vec{a}) = \phi(\vec{a})$ where

$$a_i = \begin{cases} a_i & \text{if } \rho(a_i) = * \\ \rho(a_i) & \textit{otherwise} \end{cases}$$

Random Restrictions

Definition

A constant simplification is one in which every occurence of a single literal is replaced by a constant $c \in \{0, 1\}$

Random Restrictions

- Such restrictions are used to decrease the size of the formula.
- These restrictions can also convert some non-trivial gates to trivial ones leading to further reduction in size.
- $\mathcal{R}_k$ denotes the set of all random restrictions that fix exactly $n - k$ variables in the formula.
- Simple observation : $|\mathcal{R}_k| = \binom{n}{k} 2^{n-k}$

Theorem 1

For every boolean function f, it is possible to fix one of its variables such that the resulting function f' satisfies

$$L(f') \leq \left(1 - \frac{1}{n}\right)^{\frac{3}{2}} L(f)$$

where $L(f)$ denotes the number of leaves (input gates) of f.

Proof

- We use $s = L(f)$ to denote the number of leaves in f and F for the minimal size formula on the DeMorgan basis that computes f.
- From the pigeonhole principle, there exists a variable a_i which occurs in at-least $\frac{s}{n}$ leaves.
- On fixing this we get, $s' \leq s\left(1 - \frac{1}{n}\right)$.
- But, we can do better!

Important Claim

Claim 1

If $z \in \{a_i, \neg a_i\}$ is a leaf in F, then the neighbor of z in the formula tree does not contain the variable a_i.

Proof

We prove this using contradiction, so let's assume that the neighbor G of z contain a leaf $z' \in \{a_i, \neg a_i\}$

- W.L.G, we can assume that $a_i \wedge G = H$ is a sub-formula of F.
- When $a_i = 0$, H becomes 0. When $a_i = 1$, H reduces to G.

Proof

- We can set all instances of a_i in G to be 1.
- This gives us a smaller formula $a_i \wedge G' = H'$ which computes the same function as H.
- But, we assumed that F is the minimal size formula for f.

This is a contradiction!

Proof

- Already seen the reduction by $\frac{s}{n}$ leaves.
- These leaves will have 1 neighbor each.
- We can make half of these vanish by choosing c smartly.
- Total reduction in size $\geq \frac{s}{n} + \frac{s}{2n} = \frac{3s}{2n}$.

$$s' \leq s\left(1 - \frac{3}{2n}\right) \leq s\left(1 - \frac{1}{n}\right)^{\frac{3}{2}}$$

Theorem 2

For every boolean function f, and for every integer $1 \leq k \leq n$, it is possible to fix $n - k$ variables so that the resulting function f' satisfies

$$L(f') \leq \left(\frac{k}{n}\right)^{\frac{3}{2}} L(f)$$

Proof

The proof is pretty straightforward and follows from the last theorem. We keep on repeating Theorem 1 $n-k$ times. On repeating, we get

$$
\begin{aligned}
s' &\leq s\left(1-\frac{1}{n}\right)^{\frac{3}{2}}\left(1-\frac{1}{n-1}\right)^{\frac{3}{2}}\dots\left(1-\frac{1}{k+1}\right)^{\frac{3}{2}} \\
&= s\left(\frac{n-1}{n}\right)^{\frac{3}{2}}\left(\frac{n-2}{n-1}\right)^{\frac{3}{2}}\dots\left(\frac{k}{k+1}\right)^{\frac{3}{2}} \\
&= s\left(\frac{k}{n}\right)^{\frac{3}{2}}
\end{aligned}
$$

Theorem 3

Let f be a boolean function and $\rho \in \mathcal{R}_k$ be a random restriction, then

$$\Pr\left[L(f|_\rho) \leq 4\left(\frac{k}{n}\right)^{\frac{3}{2}} L(f)\right] \geq \frac{3}{4}$$

Proof

- This time we deal with expectation.
- The expected size reduction on a random constant simplification is $\frac{3s}{2n}$.
- $\mathbb{E}[s'] \leq s\left(1 - \frac{1}{n}\right)^{\frac{3}{2}}$
- On repeating this k times, we get $\mathbb{E}[s'] \leq s\left(\frac{k}{n}\right)^{\frac{3}{2}}$
- From Markov's inequality, we have $\Pr[X \geq a.\mathbb{E}[X]] \leq \frac{1}{a}$

Quick Points

- The exponent $\frac{3}{2}$ is known as the *shrinkage exponent* Γ.
- Subsequent works tried to increase the exponent, and finally Hastad showed $\Gamma = 2$.
- Only applicable to circuits whose fan-out is atmost 1 (boolean formulas).
- Need a different approach for circuits in general.

Terminology

- t-CNF: AND of an arbitrary number of clauses, each being an OR of at most t literals.
- s-DNF: OR of an arbitrary number of clauses, each being an AND of at most s literals.
- $f|_{\rho\pi}$: The subfunction obtained on applying another restriction π of the remaining variables.
- *minterm* of f: Minimal subset of variables in f such that the function can be converted to a constant function evaluating to value 1 by assigning these subset of variables to 0 or 1 in some manner.
- $min(f)$: Length of the longest minterm of f, thus representing the largest minimal subset.
- p-random restriction: A random restriction which leaves a variable unassigned with probability p.

Hastad's Switching Lemma

Objective

To transform t-CNF into s-DNF where s is as small as possible.

Statement [1]

Let f be a t-CNF, and let ρ be a p-random restriction. Then $P\left[min(f|_{\rho}) > s\right] \leq (16pt)^s$

Proving Hastad's Switching Lemma [3]

We use the the non-probabilistic proof presented by Razborov to prove the statement of Hastad's Switching Lemma.

Terminology

- n: Total number of variables.
- s and $l \in \mathbb{Z}, 1 \leq s \leq l \leq n$
- $\mathcal{R}^l$: Set of all restrictions leaving exactly l variables unassigned.
- $Bad_f(l, s) := \{\rho \in \mathcal{R}^l \mid min(f|_\rho) > s\}$: All restrictions $\rho \in \mathcal{R}^l$ for which $f|_\rho$ cannot be written as an s-DNF.
- F: t-CNF formula for f

Lemma 1

If f is a t-CNF then: $|Bad_f(l, s)| \leq |\mathcal{R}^{l-s}| \cdot (4t)^s$

To show Lemma 1 implies Hastad's switching lemma

For a random restriction ρ in $\mathcal{R}^l$ for $l = pn$, for every $p \leq \frac{1}{2}$:

$$
\begin{aligned}
P\left[f|_{\rho} \text{ cannot be written as a s-DNF}\right] &\leq \left(\frac{|Bad_f(l,s)|}{|\mathcal{R}^l|}\right) \\
&\leq \left(\frac{\binom{n}{l-s}\cdot 2^{n-l+s}\cdot (4t)^s}{\binom{n}{l}\cdot 2^{n-l}}\right) \\
&\leq \left(\left(\frac{l}{n-l}\right)^s \cdot (8t)^s\right) \\
&= \left(\left(\frac{8tp}{1-p}\right)^s\right) \\
&\leq (16pt)^s
\end{aligned}
$$

$\square$

Proof of Lemma 1

We construct a mapping $M : A \to B$, such that B is a small set and we can give a way to retrieve every element $a \in A$ from the M(a) implying our mapping is injective and thus $|A| \leq |B|$.

$$M : Bad_f(l, s) \to \mathcal{R}^{l-s} \times S \text{ with } S \subseteq \{0,1\}^{ts+s}, |S| \leq (4t)^s$$

Thus we can reconstruct ρ from $M(\rho)$ and as stated above, we would have proven the lemma.

Proof of Lemma 1

- Fixing a bad restriction $\rho \in Bad_f(l, s)$. Now by definition, $f|_\rho$ must contain some minterm π' of size $s' \geq s + 1$.
- On applying ρ to F, some set of clauses, C' disappear due to one of the variables in those clauses being specified as 1.
- Now, some literals disappear from the set of remaining clauses, $C'' \subseteq C \setminus C'$, due to them being specified as 0.

Proof of Lemma 1

- No clause in F can be set to 0 by ρ as then f_ρ would have uniformly been 0 and likewise $f_{\rho\pi}$ cannot be constant as π' was a minterm of $f|_\rho$
- Let C_1 be the first clause of F, not set to 1 by ρ.
  Note: $\rho\pi'$ sets every clause to 1.
- The portion of π' responsible for assigning the values to the variables in C_1 are represented by π_1. Arbitrarily truncate if there are more than s variables.
- We define $\overline{\pi}_1$ as the restriction having the same support as π_1 setting the same literals to 0, thus not setting C_1 to 1.

Proof of Lemma 1

- a_1: $a_1 \in \{0,1\}^t$, a t-length binary string, such that j^{th} index of a_1 is 1 iff j^{th} variable in C_1 is specified by π_1, and by definition thus by $\overline{\pi}_1$. Thus a_1 is t-bit characteristic vector on the support of the restriction π_1, and by definition $\overline{\pi}_1$.
- Note: a_1 cannot have all bits as 0, as at least one index must be occupied by the value 1 as π_1 must specify at least 1 variable in C_1.

Why a_1?

- The utilization of a_1 is to reconstruct $\overline{\pi}_1$ given C_1.
- a_1 represents the support of $\overline{\pi}_1$ and thus what literals in C_1 must be set and the property that C_1 does not evaluate to 1 allows us to infer the restriction $\overline{\pi}_1$ itself.

Example

$$
\begin{array}{ccccccccccc}
C_1 & = & x_1 & \vee & \neg x_3 & \vee & x_7 & \vee & x_9 & \vee & x_{10} \\
\pi_1 & = & * & & 1 & & 1 & & * & & 0 \\
\overline{\pi}_1 & = & * & & 1 & & 0 & & * & & 0 \\
a_1 & = & 0 & & 1 & & 1 & & 0 & & 1
\end{array}
$$

Recursing the restrictions

- We know C_1 and a_1 and thus set the literals of C_1 whose index is occupied by 1 in a_1 and that this literal in C_1 is assigned 0 to obtain $\overline{\pi}_1$.
- If π_1 restricts less than s variables, replace π' with $\pi' \setminus \pi_1$ and ρ with $\rho\pi_1$ to find a clause C_2 using the same procedure as before.
- We define $\pi_2, \overline{\pi}_2, a_2$ for C_2 analogous to how we defined $\pi_1, \overline{\pi}_1, a_1$ for C_1.
- We repeat this procedure until we have identified some m clauses, where $m \leq s$.
- Note: $\forall i, j : i > j$, C_i contains some variable not present in C_j.
- Thus for $C_1, C_2 \ldots C_m$ we define $\pi = \pi_1\pi_2 \ldots \pi_m$ which restricts s variables.

Mapping the restrictions

- b: $b \in \{0,1\}^s$, a s-length binary string, such that j^{th} index of b is 1 iff j^{th} variable is set to same value by both π and $\overline{\pi}$. Thus every index j of $b := (\pi_{j^{th}} == \overline{\pi}_{j^{th}})$.
  Note: We are considering only the variables which have been specified by π to be represented in b. Do not confuse the j^{th} variable as x_j
- $M(\rho) := \langle \rho \cdot \overline{\pi}_1 \cdot \overline{\pi}_2 \ldots \overline{\pi}_m, a_1, a_2 \ldots, a_m, b \rangle$

Razborov's Proof

We are now left to prove the following:

- The mapping $\rho \mapsto M(\rho)$ is injective.
- Range of M is small.

We will show how to uniquely reconstruct ρ from $M(\rho)$.

Reconstructing unique C_1

Claim: The first clause of F not set to 1 by $\rho\overline{\pi}_1\overline{\pi}_2\ldots\overline{\pi}_m$ is C_1.
Proof:

- Recall: C_1 was the first clause of F not set to 1 by ρ.
- Any earlier clause must be set to 1 by ρ itself. They continue to be so for $\rho\overline{\pi}$ as well.
- For C_1, we chose $\overline{\pi}_1$ such that C_1 will not be set to 1. Since $\overline{\pi}_1$ restricted all variables common to C_1 and $\overline{\pi}$, $\overline{\pi}_2, \ldots \overline{\pi}_m$ can not set C_1 to 1.

Reconstructing unique $\overline{\pi}_1$

- a_1 reveals which literals of C_1 were set by π_1.
- We know $\overline{\pi}_1$ set the literals to 0.
- Combined with b, this uniquely determines what π_1 could have set them to.
- Thus we now know π_1 and $\overline{\pi}_1$ uniquely.

Now we can construct the restriction $\rho\pi_1\overline{\pi}_2\ldots\overline{\pi}_m$.

Reconstructing unique $\overline{\pi}_i$

- Similarly, identify the first clause of F not set to 1 by $\rho\pi_1\pi_2\ldots\pi_{i-1}\overline{\pi}_i\overline{\pi}_{i+1}\ldots\overline{\pi}_m$ as C_i.
- a_i reveals which literals of C_i were set by π_i.
- We know $\overline{\pi}_i$ set the literals to 0.
- Combined with b, this uniquely determines what π_i could have set them to.
- Thus we now know π_i and $\overline{\pi}_i$ uniquely.

Now we can construct the restriction $\rho\pi_1\pi_2\ldots\pi_{i-1}\pi_i\overline{\pi}_{i+1}\ldots\overline{\pi}_m$.

- Now we know $\overline{\pi}_1, \overline{\pi}_2, \ldots, \overline{\pi}_m$ uniquely.
- With this and $\rho\overline{\pi}_1\overline{\pi}_2\ldots\overline{\pi}_m$, we can construct the unique ρ.

Upper bounding the cardinality of the range

- $\rho\overline{\pi}_1\overline{\pi}_2\ldots\overline{\pi}_m \in \mathcal{R}^{l-s}$
- $b \in \{0,1\}^s$
- Each $a_j \in \{0,1\}^t$ has atleast one 1 and the total number of 1s across all a_j is s.

Upper bounding the cardinality of the range

- Let a_j have k_j ones. Then number of such $(a_1, \ldots, a_m)$ is

$$\prod_{j=1}^{m} \binom{t}{k_j} \leq \prod_{j=1}^{m} t^{k_j} = t^{\sum_{j=1}^{m} k_j} = t^s$$

- Number of such $k_1, \ldots k_m$ such that $k_1 + \cdots + k_m = s$ is $\binom{s-1}{m-1} \leq 2^s$.
- Thus range of $M(\rho)$ contains atmost $|\mathcal{R}^{l-s}| \times (2t)^s \times 2^s$ elements.

$\square$

Definition

Let $R(f)$ denote the minimal number r such that f can be made constant by fixing r variables to constants 0 and 1.

Example

- $R(f) = 1$ if f is AND or OR of all inputs.
- $R(\bigoplus) = n$

Lemma

If a boolean function f of n variables can be computed by a depth-$(d+1)$ alternating circuit of size S, then

$$R(f) \leq n - \frac{n}{c_d (\log S)^{d-1}} + 2 \log S$$

where $c_d > 0$ only depends on d.

Proof:

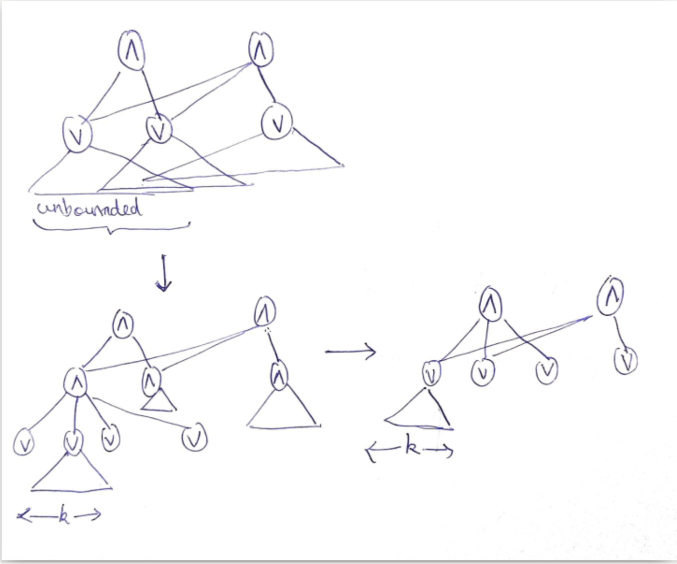
- Consider a depth-$(d+1)$ circuit of size S computing f.
- WLOG, let the bottom most layer be OR gates.
- Look at each OR gate of inputs as a 1-DNF. Apply switching lemma with $t = 1, s = 2\log S = k(\text{let}), p = 1/32$.

$$\begin{aligned}
\Pr[\text{a given 1-DNF does not become k-CNF}] &\leq (16pt)^s \\
&= (16 \times 1/32 \times 1)^{2\log S} \\
&= S^{-2} \\
\Pr[\text{atleast one 1-DNF does not become k-CNF}] &\leq S^{-1} \\
&< 1
\end{aligned}$$

- Choose such a ρ to restrict.

$PARITY \notin \mathbf{AC}^0$

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

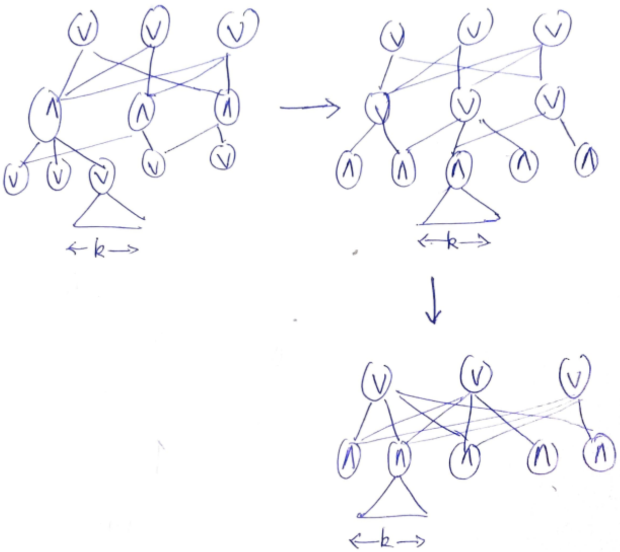


- Now we have k-CNFs at the bottom layer, and they collapse with the AND gates on the layer above, maintaining the depth to be $d + 1$. The function is on $n/32$ variables.
- Now do the following iteratively $d - 1$ times:

- Now depth $= d + 1 - i$. Number of variables $= \frac{n}{32(32k)^i}$. WLOG, the bottom two layers are k-CNFs.
- Apply switching lemma to bottom 2 layers, with $t = s = k, p = \frac{1}{32k}$.

$$\begin{aligned}\Pr[\text{given k-CNF does not become k-DNF}] &\leq \left(16 \times \frac{1}{32k} \times k\right)^{2 \log S} \\ &= S^{-2} \\ \Pr[\text{atleast one k-CNF does not become k-DNF}] &\leq S^{-1} \\ &< 1\end{aligned}$$

- Choose such a ρ. The bottom 2 layers become k-DNFs and the OR gates collapse with the 3rd layer.
- Now depth $= d + 1 - i - 1$. Number of variables $= \frac{n}{32(32k)^{i+1}}$, and the bottom two layers are k-DNFs.



k
k
k

- After $d-1$ iterations, we have depth $= 2$, Number of variables $= \frac{n}{32(32k)^{d-1}} = \frac{n}{c_d(\log S)^{d-1}}$. The circuit is either k-DNF or k-CNF.
- Trivially, fixing atmost k variables now, makes the function constant.
- Then the original function could be made constant by fixing

$$n - \frac{n}{c_d(\log S)^{d-1}} + 2\log S \text{ variables.}$$

$\square$

Theorem

Any depth - $(d+1)$ alternating circuit computing the parity of n variables require $2^{\Omega(n^{1/d})}$ gates.

Proof

Consider depth $d+1$ circuits. Then,

$$\begin{aligned} n &= R(PARITY) \\ &\leq n - \frac{n}{c_d(\log S)^{d-1}} + 2\log S \\ 2\log S &\geq \frac{n}{c_d(\log S)^{d-1}} \end{aligned}$$

Proof

$$2 \log S \geq \frac{n}{c_d (\log S)^{d-1}}$$

$$S \geq 2^{\left(\frac{n}{2c_d}\right)^{1/d}}$$

$$S \in 2^{\Omega(n^{1/d})}$$

$\square$

Corollary

$$PARITY \notin AC^0$$

References

[1] J Hastad. "Almost Optimal Lower Bounds for Small Depth Circuits". In: *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*. STOC '86. Berkeley, California, USA: Association for Computing Machinery, 1986, pp. 6–20. ISBN: 0897911938. DOI: 10.1145/12130.12132. URL: https://doi.org/10.1145/12130.12132.

[2] Stasys Jukna. "Boolean function complexity: advances and frontiers". In: vol. 27. Springer Science & Business Media, 2012, pp. 339–346.

[3] Alexander A. Razborov. "Bounded Arithmetic and Lower Bounds in Boolean Complexity". In: *Feasible Mathematics II*. Ed. by Peter Clote and Jeffrey B. Remmel. Boston, MA: Birkhäuser Boston, 1995, pp. 344–386. ISBN: 978-1-4612-2566-9.

Thank You!