Exponential Size Lower Bounds for Bounded-Depth Frege Proofs of the Pigeonhole Principle: Part II: Switching Lemmas

Paul Beame
University of Washington
Seattle, WA 98195

Recall

We needed to
(1) Choose values of k_i and n_i, with $k_i \leq k_{i+1}$, $3k_d < n_d$, and
(2) Prove a Switching Lemma for matching restrictions stating that:

For any matching disjunction F with term size at most k_i, if we choose a restriction π *at random* from among the restrictions that leave n_{i+1} range vertices unset, the probability that there does *not* exist a height k_{i+1} decision tree that represents $F\lceil_\pi$ is $< 1/S$

To motivate the proof of this we digress to the Boolean case...

Basic Form of Switching Lemma

Given:

$X_I = \{x_i\}_{i \in I}$ - set of Boolean variables

$\mathcal{R}_I$ - distribution on restrictions over X_I

F - a DNF formula in variable X_I, each of whose terms is of size $\leq r$.

Show:

For ρ chosen at random from $\mathcal{R}_I$, the probability that $F\lceil_\rho$ can be represented over $I\lceil_\rho$ by a decision tree of height $\leq s$ is at least $1 - \alpha(r, s)$.

Goals:

Elements of $\mathcal{R}_I$ should leave many variables unset

$\alpha(r, s)$ should decrease quickly with r and s

Restrictions for Parity

Boolean variables - $x_1, \ldots, x_n$
R_n^ℓ - restrictions that leave exactly ℓ unset variables

We will show canonical way to convert any DNF formula F to decision tree T_F computing F.

Switching Lemma: (essentially Håstad 1987) Given DNF formula F in variables $x_1, \ldots, x_n$ with terms of size at most r, for ρ chosen uniformly at random from R_n^ℓ the probability that $T_{F\restriction_\rho}$ has height at least s is at most $(6r\ell/n)^s$.

Canonical conversion of DNF formula F to a Boolean decision tree:

Express as an algorithm that queries variables:

0. Let $i = 1$ and τ_0 be the empty restriction.
1. If $F\lceil_{\tau_{i-1}}$ has no terms output 0. Otherwise, let D_i be the first term in F s.t. $D_i\lceil_{\tau_{i-1}} \neq 0$.
2. Let V_i be the set of variables in D_i not previously queried and let σ_i be the restriction to these variables that satisfies D_i.
3. Query all variables in V_i in order from smallest to largest.
4. Let π_i be the restriction given by the outcome of these queries.
5. Set $\tau_i = \tau_{i-1}\pi_i$.
6. If $\pi_i = \sigma_i$ (i.e. $D_i\lceil_{\tau_i} = 1$) then output 1. Otherwise, set $i \leftarrow i + 1$ and go to step 1.

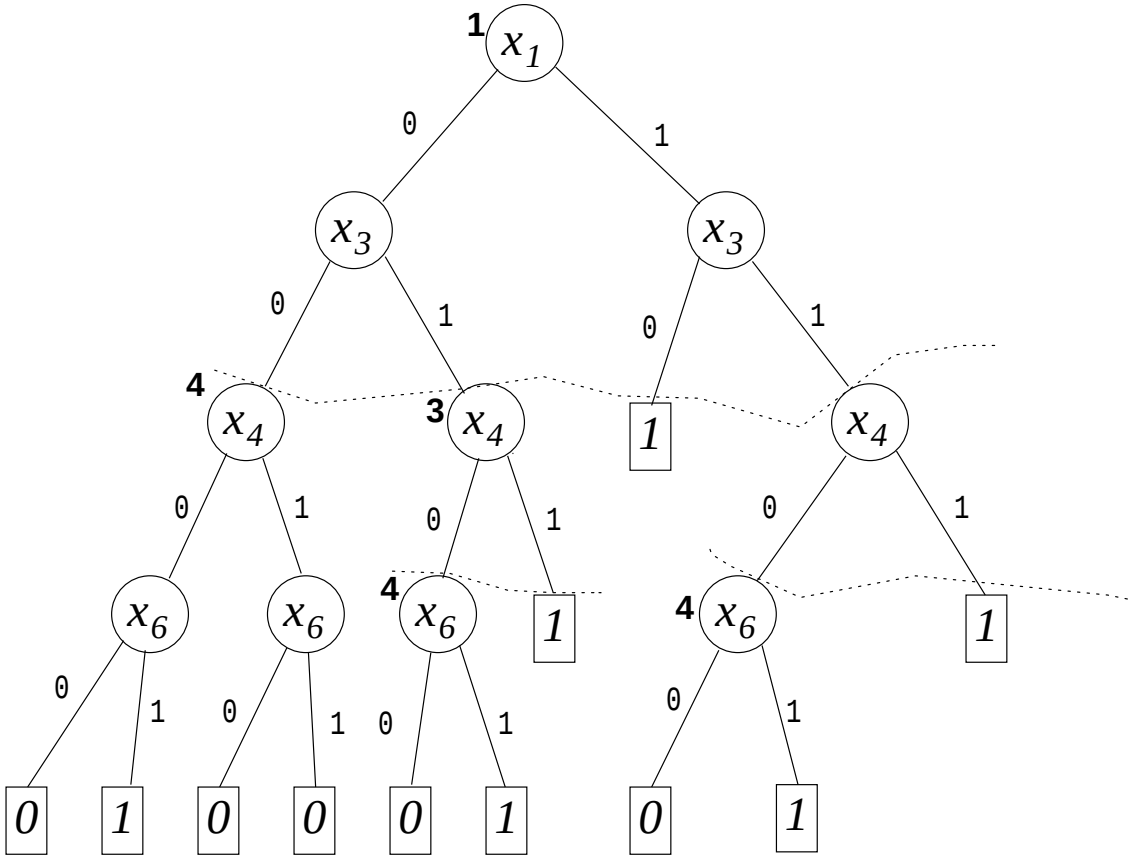
Canonical Decision Tree Expansion

F: $x_1 x_2 \bar{x}_3 + x_2 x_5 + x_3 x_4 + \bar{x}_4 x_6$

ρ: $x_2 \to 1,\ x_5 \to 0$

$F\lceil_\rho$: $x_1 \bar{x}_3 + x_3 x_4 + \bar{x}_4 x_6$

$T(F\lceil_\rho)$:



Proof of Switching Lemma:

Fix DNF formula $F = C_1 \vee C_2 \vee \ldots$ with term size $\leq r$.

Let $Bad(F)$ be those restrictions $\rho \in R_n^\ell$ such that $height(T_{F\restriction_\rho}) \geq s$.

We'll count $Bad(F)$ by giving a 1-1 map from $Bad(F)$ to a small set.

More precisely, we map any $\rho \in Bad(F)$ to a triple consisting of a restriction ρ' that sets s more variables than ρ does and two small pieces of extra information.

We prove that this is 1-1 by showing how, given F, ρ', and the extra information we can uniquely recover ρ.

Then we count the size of the image set to derive the bound on $|Bad(F)|$.

Let $\rho \in Bad(F)$, $F\lceil_\rho = C'_1 \vee C'_2 \vee \ldots$, and let π be the leftmost branch in $T_{F\lceil_\rho}$ that has length $\geq s$.

Consider the values of the internal variables of the algorithm constructing $T_{F\lceil_\rho}$ when its query
answers come from π. In particular:

(a) $\pi = \tau_k = \pi_1 \pi_2 \ldots \pi_k$ for some k
(b) For each $i < k$, $\pi_i \neq \sigma_i$ – the one assignment to V_i s.t. $\tau_{i-1}\sigma_i$ satisfies D_i, the first term in $F\lceil_\rho$ not set to 0 by τ_{i-1}.
(c) The V_i are disjoint so σ_i mutually compatible.

Now truncate π at depth s so that $|\pi| = s$. Then $\pi = \pi_1 \ldots \pi_{j-1}\pi_j$ where π_j may be truncated. Truncate V_j and σ_j similarly.

The image of bad ρ

Suppose that for each $i \leq k$, D_i is the restriction by ρ of term C_{ν_i} of F.

An encoding of $\rho \in Bad(F)$ consists of:
(1) $\rho' = \rho\sigma_1 \ldots \sigma_{j-1}\sigma_j$
(2) The value of π_i on V_i for each $i \leq j$.
(3) Positions of elements V_i in C_{ν_i} for each $i \leq j$.

Note: Without the information in (2) and (3) we cannot distinguish the ρ part of ρ' from the $\sigma_1 \ldots \sigma_j$ part.

Recovering ρ using F

Suppose that we have the encoding of ρ:

ρ', (2), and (3)

We reconstruct a series of restrictions

$$\rho_i = \rho\tau_i\sigma_{i+1}\ldots\sigma_j \qquad \text{for } i = 0,\ldots,j$$

to obtain ρ from ρ'.

0. Start with $\rho_0 = \rho'$ and $i = 1$.
1. Find first term C of F s.t. $C\lceil_{\rho_{i-1}} \neq 0$.
   (**Claim:** C is C_{ν_i}.)
2. Use information in (3) to determine
   V_i and σ_i.
3. Use information in (2) to determine π_i.
4. Change σ_i in ρ_{i-1} to π_i to obtain ρ_i.
   (Since $\tau_i = \tau_{i-1}\pi_i$, ρ_i is of right form.)
5. If $|V_1| + \cdots + |V_i| = s$ stop.

Otherwise set $i \leftarrow i + 1$ and go to 1.

Proof of Claim:

D_i = first term in $F\lceil_\rho$ s.t. $D_i\lceil_{\tau_{i-1}} \neq 0$.
$\Rightarrow C_{\nu_i}$ = first term in F not set to 0 by $\rho\tau_{i-1}$.
$\Rightarrow$ every term prior to C_{ν_i} in F is set to 0 by ρ_{i-1}.

If $i < j$ then, by choice of σ_i,
$\rho\tau_{i-1}\sigma_i$ forces C_{ν_i} to 1
$\Rightarrow C_{\nu_i}\lceil_{\rho_{i-1}} = 1 \neq 0$.

If $i = j$ the truncation of σ_j may mean that $C_{\nu_j}\lceil_{\rho_{j-1}} \neq 1$ but it still will not be 0.
□

Analysis

(1) $\rho' \in R_n^{\ell-s}$.
(2) Each C_{ν_i} has size $\leq r$ so can encode positions of elements of V_i by a vector β_i in $\{0,1\}^r \setminus \{0^r\}$ where 1 indicates element of V_i, Total # of 1's in all β_i's is s.
(3) Can encode values of all π_i given V_i by a single element of $\{0,1\}^s$.

Can easily show that

$$\frac{|R_n^{\ell-s}|}{|R_n^{\ell}|} = \frac{\binom{n}{\ell-s}2^{n-\ell+s}}{\binom{n}{\ell}2^{n-\ell}} \leq (2\ell/n)^s$$

Only 2^s possibilities for (3).

Remains to count (2):
$B(r,s)$= # of sequences of vectors
$\beta_i \in \{0,1\}^r \setminus \{0^r\}$ with total of s 1's in all β_i.

Claim: $B(r,s) \leq 2^{s-1} r^s$

\# of ways of partitioning s 1's among β_i
= \# of ways of partitioning s elements into
a sequence of non-empty groups
$= 2^{s-1}$,

To see this consider the following example:
e.g. $s = 8$, sequence $= 2,1,2,3$ described as $*ee*e**$ $\in \{*,e\}^7$ by marking the end of each group. (No need for the last mark.)

For a fixed partition of the s 1's among the β_i, there are $\leq r$ choices for the positions of each 1 within its β_i. $\square$

Sharper Bound: $B(r,s) \leq (r/\ln 2)^s$

Restrictions for PHP_n^{n+1}

Let $D = [1, n+1]$, $R = [1, n]'$.
Assume that $D \cap R = \emptyset$.
Variables P_{ij}, $i \in D$, $j \in R$.

Restrictions: Partial 1-1 mappings, i.e. elements of
$M_n = \{\text{partial bipartite matchings on } D \times R\}$.

$\pi \in M_n \Rightarrow$ restriction ρ_π such that

$$\rho_\pi(P_{ij}) = \begin{cases} 1 & \text{if } \{i,j\} \in \pi \\ 0 & \text{if } \{i,j\} \notin \pi \text{ but } \pi \text{ touches } i \text{ or } j \\ * & \text{otherwise} \end{cases}$$

Abuse of Notation: Identify π and ρ_π.

Observe that applying π sets all variables whose endpoints are touched by π. Use $D\restriction_\pi$ and $R\restriction_\pi$ to denote the sets of points in D and R for which variables remain.

Matching Decision trees

A *matching decision tree over* $D \cup R$ is a rooted directed tree T whose:
Internal nodes are labelled by elements of $D \cup R$,
Leaves are labelled by 0 or 1 so that:

1. (a) If the root of T is labelled by $i \in D$ then for each $j \in R$ there is one out-edge from the root labelled $\{i, j\}$.

(b) If the root of T is labelled by $j \in R$ then for each $i \in D$ there is one out-edge from the root labelled $\{i, j\}$.

(c) There are no other out-edges from the root of T.

2. Let $T^{\{i,j\}}$ be the tree whose root is the node connected to the root of T by the edge labelled $\{i, j\}$. $T^{\{i,j\}}$ is a matching decision tree over $D' \cup R'$ where $D' = D \setminus \{i\}$ and $R' = R \setminus \{j\}$.

Def: Define sets
$\text{br}(T) = \{\text{branches (root-leaf paths) in } T\}$.
$\text{br}_a(T) = \{\text{branches in } T \text{ with leaf label } a\}$.

The set of edge labels along any branch of T forms a partial matching. Identify a branch with its matching so $\text{br}(T)$ and $\text{br}_a(T)$ become sets of partial matchings.

For any matching decision tree T, let T^c be the same tree as T except that the leaf labels 0 and 1 are reversed,
i.e. $\text{br}_1(T^c) = \text{br}_0(T)$ and $\text{br}_0(T^c) = \text{br}_1(T)$.

Def: Matching decision tree T *represents* boolean function/formula/circuit f iff:

$\forall \pi \in \text{br}(T)$, $f\lceil_\pi \equiv$ the leaf label of π in T.

Note: Unlike ordinary Boolean decision trees, the fact that T represents f only determines the value of f on some inputs.

Intuition:

$T \approx "true"$ iff $\text{br}_1(T) = \text{br}(T)$,
i.e. every leaf of T has label 1.

$T \approx "false"$ iff $\text{br}_0(T) = \text{br}(T)$,
i.e. every leaf of T has label 0.

Bipartite matching switching lemma

Let M_n^ℓ be the set of restrictions on P_{ij}, $i \in D$, $j \in R$, $|R| = |D| - 1 = n$ consisting of all matchings with $n - \ell$ edges.

We'll show canonical conversion of matching disjunction to matching decision tree $T_S(F)$ where $S = D \cup R$ is the vertex domain.

Switching Lemma: (essentially (Beame, Impagliazzo, Krajíček, Pitassi, Pudlák, Woods 1992)) Let $10 \leq \ell \leq \sqrt{n/r}/10$ and ρ be chosen uniformly at random ¿from M_n^ℓ. Given matching disjunction F in variables P_{ij} over vertex set S with terms of size at most r, the probability that $T_{S\restriction_\rho}(F\restriction_\rho)$ has height at least s is at most $(1.5\ell^2\sqrt{r/n})^s$.

Canonical conversion of matching disjunction F to a matching decision tree over domain S:

Express as algorithm querying vertices in S for the unique edge with endpoints in S touching them.

0. Let $i = 1$ and τ_0 be the empty restriction.
1. If $F\lceil_{\tau_{i-1}} = 0$ output 0. Otherwise, let D_i be the first term in F such that $D_i\lceil_{\tau_{i-1}} \neq 0$.
2. Let V_i be the variables in D_i that are not matched in τ_{i-1} and let σ_i be the matching that sets all variables in V_i to 1, i.e. $D_i\lceil_{\tau_{i-1}\sigma_i} = 1$.
3. Let S_i be the set of *endpoints* of variables in V_i and query all vertices in S_i in order from smallest to largest.
4. Let π_i be the partial matching given by the outcome of these queries.
5. Set $\tau_i = \tau_{i-1}\pi_i$.
6. If $\pi_i = \sigma_i$ (i.e. $D_i\lceil_{\tau_i} = 1$) then output 1. Otherwise, set $i \leftarrow i + 1$ and go to step 1.

Proof of PHP Switching Lemma

Similar structure to Boolean case:
Fix $F = C_1 \vee C_2 \vee \ldots$ with term size $\leq r$.

Let $Bad(F)$ be those restrictions $\rho \in M_n^\ell$ such that $height(()T_{S\restriction_\rho}(F\restriction_\rho)) \geq s$.
Again count $Bad(F)$ by giving a 1-1 map from $Bad(F)$ to set based on algorithm constructing tree. For $\rho \in Bad(F)$ this consists of:

(1) $\rho' = \rho\sigma_1 \ldots \sigma_{j-1}\sigma_j$
(2) Values of π_i given V_i for each $i \leq j$.
(3) Positions of elements V_i in C_{ν_i} for each $i \leq j$.

(Note: not only are the V_i disjoint but they have disjoint sets of vertices S_i so that the σ_i are mutually compatible.)

The reconstruction argument is the same as in the Boolean case.

Unlike Boolean case, may have $|\sigma_i| \neq |\pi_i|$.

However, $|\sigma_i| = |S_i|/2$ and $|\pi_i| \leq |S_i|$.
$\Rightarrow |\sigma_i| \geq |\pi_i|/2$.
$\Rightarrow \rho' \in M_n^{\ell - j}$ where $j \geq s/2$.

Information in (3) is $B(r, j)$ from the Boolean case.

Trickiest part:
Encoding π_i given V_i (and thus S_i too.)

Observe: π_i has no vertices in common with ρ and $\sigma_{i'}$ for $i' \neq i$.
$\Rightarrow$ Every edge in π_i consists of an element of S_i paired with an element unset by ρ that is either in S_i or unset by ρ' too.

We'll use only one number of size $\ell + 1$ to encode each edge of π_i:

Order edges of π_i by their endpoints in S_i where $D < R$. (Using the smaller one if they have two.) The edges in π_i can be specified uniquely by the sequence consisting of the other endpoints of these edges.

e.g. $\sigma_i = \{\{1, 1'\}, \{3, 3'\}\}$, $S_i = \{1, 3, 1', 3'\}$,
$\pi_i = \{\{1, 3'\}, \{3, 4'\}, \{2, 1'\}\}$ in order.
The sequence for π_i would be $3', 4', 2$.

Each listed endpoint is unset by ρ and its 'side' (domain or range) is already known.
$\Rightarrow$ There are most $\ell + 1$ possibilities for each element of the list.

SO: Encode each edge of π_i simply by a number from $[1, \ell + 1]$.

BUT: In decoding π_i, ρ is not known, only ρ' and $\sigma_1 \ldots \sigma_i$ are known.

So for encoding π_i, number elements of $D\lceil_\rho$ beginning with the $\ell + 1 - j$ elements of $D\lceil_{\rho'}$ followed by the j vertices set by σ_1, σ_2, etc. Do the same for R except there are only ℓ elements in $R\lceil_\rho$.

Analysis

Size of image set for $\rho \in Bad(F)$

$$\leq \sum_{j \geq s/2} |M_n^{\ell-j}|(\ell+1)^s B(r,j)$$

Now $|M_n^\ell| = \binom{n}{\ell} \frac{(n+1)!}{(\ell+1)!}$ so

$$\frac{|M_n^{\ell-j}|}{|M_n^\ell|} = \frac{\ell!\ (n-\ell)!\ (\ell+1)!}{(\ell-j)!\ (n-\ell+j)!\ (\ell-j+1)!}$$

$$= \frac{\ell^{(j)}(\ell+1)^{(j)}}{(n-\ell+j)^{(j)}}$$

$$\leq \left(\frac{\ell(\ell+1)}{n-\ell}\right)^j$$

$\Rightarrow$ Fraction of restrictions in $Bad(F)$ is at most

$$\sum_{j \geq s/2} \frac{|M_n^{\ell-j}|}{|M_n^{\ell}|}(\ell+1)^s B(r,j)$$

$$\leq \ (\ell+1)^s \sum_{j \geq s/2} \left(\frac{r\ell(\ell+1)}{(n-\ell)\ln 2}\right)^j$$

Since $1/\ln 2 < 1.4427$ and $10 \leq \ell \leq .\sqrt{n/r}/10$, we have $(\ell+1) \leq 1.1\ell$ and $n \leq 1.02(n-\ell)$. Thus the series is at most

$$(1.1\ell)^s \sum_{j \geq s/2} (1.7r\ell^2/n)^j.$$

This is a geometric series with ratio $< .02$. Therefore it is at most

$$1.03(1.1\ell)^s(1.7r\ell^2/n)^{s/2}$$
$$\leq \ (1.5\ell^2\sqrt{r/n})^s.$$

□

This approach of proving a result about a canonical conversion of a DNF formula is often referred to as a Håstad-style switching lemma.

Håstad's original proof involved conditional probablity arguments that were eliminated in (Razborov 1994) which used on some ideas of Woods.

The method presented here is from (Beame 1994). It is an adaptation and extension of the method of (Razborov 1994) to decision trees.

Determining k_i and n_i

Suppose that we have an (S, d)-proof of PHP_n^{n+1} where $S \leq n^{n^{1/5^{d+1}}}$ and $n \geq 6^{d^2}$.
Recall that $k_0 = 1$ and $n_0 = n$.

We set $k_i = k = 5^d \log_n S$ and $n_{i+1} = n_i^{1/5}$ for $i > 0$ so that $n_i = n^{-1/5^i}$. Also,

$$1.5\sqrt{k} < 5^d n^{1/(2 \cdot 5^{d+1})} < n^{1/5^{d+1}} = n_d^{1/5}.$$

Now for $i < d$ apply the PHP Switching Lemma with $n = n_i$, $\ell = n_{i+1}$, $r = k$, and $s = k$. This gives a failure probability of at most

$$\begin{aligned}
(1.5\ell^2\sqrt{r/n})^s &= (1.5n_{i+1}^2\sqrt{k/n_i})^k \\
&= (1.5n_i^{1/5}\sqrt{k/n_i})^k \\
&= (1.5\sqrt{k}n_i^{-3/10})^k \\
&< (n^{-1/5^{i+1}})^{5^d \log_n S} \\
&\leq n^{-\log_n S} = 1/S
\end{aligned}$$

□

Summary

Therefore we have that if $S \leq n^{n^{1/5^{d+1}}}$ and $S \geq 6^{d^2}$ there is no size S, depth d Frege proof of PHP_n^{n+1}.

Note that all of the technique can be carried over to bounded-depth circuits as well as formulas so it may be applied to bounded-depth $e\mathcal{F}$-proofs too.

The second level exponent can be improved to $\Omega(1/(d4^d))$.

Cor: $PHP(f)$ is not provable in $S_2(f)$.

Cor: Polynomial-size $\mathcal{F}$ proofs of PHP_n^{n+1} require depth $\Omega(\log\log n)$.