Chapter 13

Lower Bounds

The very simple problem PARITY is too hard for first-order logic, no matter what numeric predicates we add. When we add counting, but remove ordering, PARITY is expressible. However, a different sort of parity problem becomes inexpressible. A related lower bound suggests that complete problems for P are inherently sequential.

13.1 Håstad’s Switching Lemma

Recall boolean query PARITY, which is true of boolean strings that have an odd number of ones. Using pebble games, we have shown that PARITY is not first-order in the absence of the numeric predicate BIT (Proposition 6.14, Proposition 6.44). This theorem is much more subtle with the inclusion of BIT.

Theorem 13.1 Query PARITY is not first-order expressible: PARITY \(\not\in\) FO.

The known proofs of Theorem 13.1 all prove the stronger result that PARITY is not in the non-uniform class AC⁰/poly or, equivalently, PARITY is not first-order, no matter what numeric predicates are available (Proposition 11.19). The proof we present here is via the Håstad Switching Lemma, following the treatment in [Bea96].

Let \(f\) be a boolean function, with boolean variables \(V_n = \{x_1, \ldots, x_n\}\). A restriction on \(V_n\) is a map \(\rho : V_n \to \{0, 1, \star\}\). The idea is that some of the variables are set to “0” or “1” and the others — those assigned “\(\star\)” — remain variables.
Restriction $\rho$ applied to function $f$ results in function $f|_{\rho}$ in which value $\rho(x_i)$ is substituted for $x_i$ in $f$, for each $x_i$ such that $\rho(x_i) \neq \star$. Thus, $f|_{\rho}$ is a function of the variables that have been assigned “$\star$”. Let $\mathcal{R}_n^r$ be the set of all restrictions on $V_n$ that map exactly $r$ variables to “$\star$”.

We state and prove the switching lemma using decision trees. Given a formula $F$ in disjunctive normal form (DNF) define the canonical decision tree $T(F)$ for $F$ as follows: Let $C_1 = \ell_1 \land \cdots \land \ell_i$ be the first term of $F$, so $F = C_1 \lor F'$. The top of $T(F)$ is a complete binary decision tree on the variables in $C_1$. Each leaf of the tree determines a restriction $\rho$ that assigns the appropriate value to the variables in $C_1$ and assign “$\star$” to all the other variables. There is a unique leaf that makes $C_1$ true and this should remain a leaf and be labeled “1”. To each other leaf, determining restriction $\rho$, we attach the canonical decision tree $T(F'|_{\rho})$.

Let $h(T)$ be the height of tree $T$. We now show that for any formula $F$ in DNF, if $F$ has only small terms, then when randomly choosing a restriction $\rho$ from $\mathcal{R}_n^r$, with high probability the height of the canonical decision tree of the resulting formula, $h(T(F|_{\rho}))$, is small.

It then follows that the negation of $F|_{\rho}$ can also be written in DNF — as the disjunction of the conjunction of each branch in the tree that leads to “0”. Thus, with high probability, a random restriction switches a DNF formula that has only small terms to a conjunctive normal form (CNF) formula.

**Lemma 13.2 (Håstad Switching Lemma)** Let $F$ be a DNF formula on $n$ variables, such that each of its terms has length at most $k$. Let $p \leq 1/7$, $r = pn$, and $s \geq 0$. Then,

$$\frac{|\{\rho \in \mathcal{R}_n^r | h(T(F|_{\rho})) \geq s\}|}{|\mathcal{R}_n^r|} < (7pk)^s.$$  

**Proof** The proof of Lemma 13.2 is a somewhat intricate counting argument. Let Stars$(k, s)$ be the set of all sequences $w = (S_1, S_2, \ldots, S_t)$ where each $S_i$ is a nonempty subset of $\{1, 2, \ldots, k\}$ and the sum of the cardinalities of the $S_i$’s equals $s$. We use the following upper bound on the size of Stars$(k, s)$.

**Lemma 13.3** For $k, s > 0$, $|\text{Stars}(k, s)| \leq (k/\ln 2)^s$.

A DNF formula is an “or” of “and”s. This is the dual of CNF.
13.1. HÄSTAD’S SWITCHING LEMMA

Proof We show by induction on \( s \) that \(|\text{Stars}(k, s)| \leq \gamma^s\), where \( \gamma \) is such that \((1 + 1/\gamma)^k = 2\). Since \((1 + 1/\gamma) < e^{1/\gamma}\), we have \( \gamma < k/\ln 2 \) and thus the lemma will follow.

Suppose that the lemma holds for any \( s' < s \). Let \( \beta \in \text{Stars}(k, s) \). Then \( \beta = (S_1, \beta') \), where \( \beta' \in \text{Stars}(k, s - i) \) and \( i = |S_1| \). Thus,

\[
|\text{Stars}(k, s)| = \sum_{i=1}^{\min(k,s)} \binom{k}{i} |\text{Stars}(k, s - i)|
\]

Thus, by the induction hypothesis,

\[
|\text{Stars}(k, s)| \leq \sum_{i=1}^{k} \binom{k}{i} \gamma^{s-i}
\]

\[
= \gamma^s \sum_{i=1}^{k} \binom{k}{i} (1/\gamma)^i
\]

\[
= \gamma^s [(1 + 1/\gamma)^k - 1] = \gamma^s.
\]

\[\square\]

Let \( R \subseteq \mathcal{R}_n \) be the set of restrictions \( \rho \) such that \( h(T(F|\rho)) \geq s \). We will define a 1:1 map,

\[
\alpha : R \to \mathcal{R}_n^{r-s} \times \text{Stars}(k, s) \times 2^s. \tag{13.4}
\]

Once we show that \( \alpha \) is one to one, it will follow that

\[
\frac{|R|}{|\mathcal{R}_n^r|} \leq \frac{|\mathcal{R}_n^{r-s}|}{|\mathcal{R}_n^r|} \cdot |\text{Stars}(k, s)| \cdot 2^s. \tag{13.5}
\]

Observe that \( |\mathcal{R}_n^{r-s}| = \binom{n}{r} 2^{n-r} \), so,

\[
|\mathcal{R}_n^{r-s}| = \frac{(r)(r-1) \cdots (r-s+1)}{(n-r+s)(n-r+s-1) \cdots (n-r+1)} \cdot 2^s \leq \left( \frac{2r}{n-r} \right)^s.
\]

Substituting this into Equation (13.5) and using Lemma 13.3 we have,

\[
\frac{|R|}{|\mathcal{R}_n^r|} \leq \left( \frac{2r}{n-r} \right)^s \cdot (k/\ln 2)^s \cdot 2^s
\]

\[
= \left( \frac{4rk}{(n-r)\ln 2} \right)^s
\]

\[
= \left( \frac{4pk}{(1-p)\ln 2} \right)^s
\]
when \( r = pn \). This is less than \((7pk)^s\) when \( p < 1/7 \).

It thus suffices to construct 1:1 map \( \alpha \) (Equation (13.4)). Let \( F = C_1 \lor C_2 \lor \ldots \). Let \( \rho \in R \), and let \( C_{i_1} \) be the first term of \( F \) that is not set to “0” in \( F|_\rho \).

Let \( b \) be the first \( s \) steps of the lexicographically first branch in \( T(F|_\rho) \) that has length at least \( s \). Let \( V_1 \) be the set of variables in \( C_{i_1}|_\rho \). Let \( a_1 \) be the assignment to \( V_1 \) that makes \( C_{i_1}|_\rho \) true. Let \( b_1 \) be the initial segment of \( b \) that assigns values to \( V_1 \). If \( b \) ends before all the values of \( V_1 \) are defined, then let \( b_1 = b \), and shorten \( a_1 \) so that it assigns values only to the variables that \( b_1 \) does. See Figure 13.6.

Define the set \( S_1 \subseteq \{1, 2, \ldots, k\} \) to include those \( j \) such that the \( j^{th} \) variable in \( C_{i_1} \) is set by \( a_1 \). \( S_1 \) is nonempty. Note that from \( C_{i_1} \) and \( S_1 \) we can reconstruct \( a_1 \).

If \( b \neq b_1 \), then \((b - b_1)\) is a path in \( T(F|_{\rho b_1}) \). Let \( C_{i_2} \) be the first term of \( F \) not set to “0” by \( \rho b_1 \). As above, we generate \( b_2, a_2, \) and \( S_2 \). Repeat this until the whole branch \( b \) is used up. We have \( b = b_1 b_2 \cdots b_t \), and let \( a = a_1 a_2 \cdots a_t \). Define the map \( \delta : \{1, \ldots, s\} \to \{0, 1\} \) such that \( \delta(j) = 1 \) if \( a \) and \( b \) assign the same value at their step \( j \), and \( \delta(j) = 0 \) if \( a \) and \( b \) assign different values to variable \( j \). We finally define the map \( \alpha \) as,

\[
\alpha(\rho) = \langle \rho a, (S_1, S_2, \ldots, S_t), \delta \rangle.
\]

From \( \alpha(\rho) \) we can reconstruct \( \rho \) as follows: \( C_{i_1} \) is the first clause that evaluates to “1” using \( \rho a \). From \( C_{i_1} \) and \( S_1 \) we reconstruct \( a_1 \). Then, using \( \delta \), we can compute the restriction \( \rho' = \rho b_1 a_2 \cdots a_t \). Next, \( C_{i_2} \) is the first clause evaluating to “1” using \( \rho' \). From this and \( S_2 \), we can compute \( a_2 \), and so on. Thus \( \alpha \) is 1:1. This completes the proof of Håstad’s Switching Lemma.

A striking consequence of the switching lemma is that \( \text{AC}^0 \) circuits have restrictions on which they are constant even though many variables are assigned to “\( \star \)”: 

**Theorem 13.7** Let \( C \) be an unbounded fan-in circuit with \( n \) inputs, having size \( s \) and depth \( d \). Let \( r \leq n/(14^d (\log s)^{d-1}) - (\log(s) - 1) \). Then there is a restriction \( \rho \in R_n \) for which \( C|_\rho \) is constant.

**Proof** We show inductively from the leaves up, that there is a restriction that turns all the gates into DNF or CNF formulas all of whose terms have length at most \( \log s \).
Figure 13.6: Decision tree $T(F|\rho)$ with path of length $s$, $b = b_1 b_2 \cdots b_t$. 
CHAPTER 13. LOWER BOUNDS

Assume that level one of the circuit — the nodes sitting above the inputs and their negations — consists of “or” gates. Thus, each of these gates $g$ is a DNF formula whose maximum term size is one. By Lemma [13.2] with $p = 1/14, n_1 = n/14, k = 1$, we have,

$$|\{\rho \in \mathcal{R}_n^{n_1} | h(T(g|_\rho)) \geq \log s\}| < (2)^{-\log s} \cdot |\mathcal{R}_n^{n_1}| .$$

Since there are at most $s$ gates at level one, the number of restrictions $\rho$ such that $h(T(g|_\rho)) \geq \log s$ for some $g$ is less than,

$$s \cdot (2)^{-\log s} \cdot |\mathcal{R}_n^{n_1}| = |\mathcal{R}_n^{n_1}| .$$

Thus, there is at least one restriction $\rho_1 \in \mathcal{R}_n^{n_1}$ under which all the gates at level one are CNF formulas with terms of size less than $\log s$. It follows that the “and” gates at level two are CNF formulas with terms of size less than $\log s$.

Let $g_2 = g|_{\rho_1}$ be any such gate. Using Lemma [13.2] with $k = \log s, p = 1/(14 \log s), n_2 = n_1/(14 \log s)$, we have,

$$|\{\rho \in \mathcal{R}_n^{n_2} | h(T(g_2|_\rho)) \geq \log s\}| < (2)^{-\log s} \cdot |\mathcal{R}_n^{n_1}| .$$

Thus, there is a restriction $\rho_2 \in \mathcal{R}_n^{n_2}$ under which every gate at level two is a DNF formula all of whose terms have length less than $\log s$.

Repeating this argument through all $d$ levels, we have a restriction $\rho = \rho_1 \rho_2 \cdots \rho_d \in \mathcal{R}_n^{n_d}$ such that the height $T(C|_\rho)$ of the decision tree of the root of the circuit is less than $\log s$. Observe that $n_d = n/(14^d(\log s)^{d-1})$. Let $b$ be the restriction corresponding to any branch of the decision tree. It follows that $C|_\rho b$ is constant and has at least $r = n_d - (\log(s) - 1)$ inputs. □

Suppose that circuit $C$ in Theorem [13.7] computes the parity of its $n$ inputs. Then any restriction of $C$ also computes the parity of its remaining inputs. Thus, if $1 \leq r$ in Theorem [13.7] then $C$ must not compute PARITY. It follows that if $C$ is a size $s$, depth $d$ circuit computing parity on $n$ inputs, then the following inequalities hold,

\[
1 > \frac{n}{(14^d(\log s)^{d-1}) - (\log(s) - 1)} \\
\log s > \frac{n}{(14^d(\log s)^{d-1})} \\
(\log s)^{d} > \frac{n}{(14^{d})} \\
s > 2^{1/n^{d}} .
\]
We thus have the following lower bound on the number of iterations of a first-order quantifier block needed to compute PARITY. This corollary is optimal by Exercise 4.19.

We use the “big omega” notation for lower bounds. The “equation” \( f(n) = \Omega(g(n)) \) is equivalent to \( g(n) = O(f(n)) \). It means that for almost all values of \( n \), \( f(n) \) is at least some constant multiple of \( g(n) \).

**Corollary 13.8** If \( \text{PARITY} \in \text{FO}[s(n)] \), then \( s(n) = \Omega(\log n / \log \log n) \), and this holds even in the presence of arbitrary numeric predicates.

**Exercise 13.9** Show that \( \text{PARITY} \) is first-order reducible to \( \text{REACH} \). Conclude that the same lower bound as in Corollary 13.8 holds for \( \text{REACH} \).