

Dimension, Halfspaces, and the Density of Hard Sets*

Ryan C. Harkins[†]

John M. Hitchcock[‡]

Abstract

We use the connection between resource-bounded dimension and the online mistake-bound model of learning to show that the following classes have polynomial-time dimension zero.

1. The class of problems which reduce to nondense sets via a majority reduction.
2. The class of problems which reduce to nondense sets via an iterated reduction that composes a bounded-query truth-table reduction with a conjunctive reduction.

Intuitively, polynomial-time dimension is a means of quantifying the size and complexity of classes within the exponential time complexity class E. The class P has dimension 0, E itself has dimension 1, and any class with dimension less than 1 cannot contain E. As a corollary, it follows that all sets which are hard for E under these types of reductions are exponentially dense. The first item subsumes two previous results and the second item answers a question of Lutz and Mayordomo. Our proofs use Littlestone’s Winnow2 algorithm for learning r -of- k threshold functions and Maass and Turán’s algorithm for learning halfspaces.

1 Introduction

Recent work has found applications of computational learning theory to the resource-bounded measure [10] and dimension [12] of complexity classes. Lindner, Schuler, and Watanabe [8] studied connections between computational learning theory and resource-bounded measure [10], primarily focusing on the PAC (probably approximately correct) model. They also observed that any admissible subclass of P/poly that is learnable in Littlestone’s online mistake-bound model [9] has p-measure 0. Later Hitchcock [6] developed a general tool for applying online learning algorithms in resource-bounded dimension. Intuitively, polynomial-time dimension is a means of quantifying the size and complexity of classes within the exponential time complexity class E. The class P has p-dimension 0, E itself has p-dimension 1, and any class with p-dimension less than 1 cannot contain E. To show that a class has p-dimension 0, it suffices to show that it is reducible to a learnable concept class family. This idea was used to show that the following classes have p-dimension 0.

- (1) $P_{\text{ctt}}(\text{DENSE}^c)$.
- (2) $P_{\text{dtt}}(\text{DENSE}^c)$.
- (3) $P_{n^\alpha - T}(\text{DENSE}^c)$, for all $\alpha < 1$.

*This research was supported in part by NSF grants 0515313 and 0652601.

[†]Department of Computer Science, University of Wyoming. rharkins@cs.uwyo.edu.

[‡]Department of Computer Science, University of Wyoming. jhitchco@cs.uwyo.edu.

Here $P_r(\text{DENSE}^c)$ is the class of all problems which reduce to nondense sets under \leq_r^P reductions, where a problem is nondense if its census function is subexponential. The result for (3) improved previous work [4, 13, 15] and solved one of Lutz and Mayordomo’s twelve problems in resource-bounded measure [14]. Results (1) and (2) improved previous separations due to Watanabe [18]. The classes in (2) and (3) were reduced to disjunctions, which can be learned by Littlestone’s Winnow algorithm [9]. We obtain further results in this direction using more sophisticated learning algorithms and concept classes that generalize disjunctions.

In our first result we show that the class

(4) $P_{\text{maj}}(\text{DENSE}^c)$

of problems which reduce to nondense sets via majority reductions has p-dimension 0. Our proof gives a reduction to r -of- k threshold functions and applies Littlestone’s Winnow2 algorithm. This subsumes the results about (1) and (2) above and answers a question of Fu [5].

Our second result concerns iterated reductions and answers the following question of Lutz and Mayordomo [13]:

(Q) Does the class $P_{\text{btt}}(P_{\text{ctt}}(\text{DENSE}^c))$ have measure 0 in E ?

Agrawal and Arvind [1] showed that $P_{\text{btt}}(P_{\text{ctt}}(\text{SPARSE})) \subseteq P_{\text{m}}(\text{LT}_1)$, where LT_1 is the class of problems that have a nonuniform family of depth-1 weighted linear threshold circuits. Equivalently, LT_1 is the class of problems where each input length is a halfspace. We use their technique to reduce

(5) $P_{\alpha \log n - \text{tt}}(P_{\text{ctt}}(\text{DENSE}^c))$, for all $\alpha < 1$

to a subexponential-size family of halfspaces. We then apply the online learning algorithm of Maass and Turán [16] to learn these halfspaces and conclude that the classes in (5) have p-dimension 0. This strongly answers (Q) in the affirmative.

This paper is organized as follows. Section 2 contains preliminaries about halfspaces, learning, and dimension. The majority reductions result is in section 3 and the iterated reductions result is in section 4. Section 5 concludes with some observations for NP and directions for further work.

2 Preliminaries

A language L is a subset of $\{0, 1\}^*$. For the length of a string x , we write $|x|$. By $L_{=n}$ we denote the set all strings in L of length n , and by $L_{\leq n}$ we denote the set of all strings in L with length at most n . We also use the notation $|X|$ to denote the number of strings in a set X when X is finite. Let L be a language.

- L is *sparse* if for all $n \in \mathbb{N}$, $|L_{\leq n}| \leq p(n)$, where $p(n)$ is a polynomial.
- L is *dense* if for some $\epsilon > 0$, for all but finitely many n , $|L_{\leq n}| > 2^{n^\epsilon}$.
- L is *io-dense* if for some $\epsilon > 0$, for infinitely many n , $|L_{\leq n}| > 2^{n^\epsilon}$.

We write SPARSE , DENSE , and $\text{DENSE}_{\text{i.o.}}$ for the classes of sparse, dense, and io-dense languages, respectively. Note that $L \in \text{DENSE}^c$ if for all $\epsilon > 0$, for infinitely many n , $|L_{\leq n}| < 2^{n^\epsilon}$, and $L \in \text{DENSE}_{\text{i.o.}}^c$ if for all $\epsilon > 0$, for all sufficiently large n , $|L_{\leq n}| < 2^{n^\epsilon}$.

We assume the reader is familiar with the various notions of polynomial-time reductions. If a reduction $g(x)$ produces a single query, then $|g(x)|$ refers to the size of that query. If it produces multiple queries, then $|g(x)|$ is the number of queries produced.

2.1 Threshold Circuits

A *weighted linear threshold gate* with n inputs is determined by a weight vector $\hat{w} \in \mathbb{Q}^n$ and a threshold $T \in \mathbb{Q}$ such that on inputs $x \in \{0, 1\}^n$, where x is considered an n -valued vector (x_1, x_2, \dots, x_n) , the gate will output 1 if and only if $\sum_{1 \leq i \leq n} w_i x_i > T$. An *exact weighted linear threshold gate* is defined similarly, except that the gate will output 1 if and only if $\sum_{1 \leq i \leq n} w_i x_i = 0$. As this is an inner product on vectors, we use the notation $\hat{w} \cdot \hat{x}$ for $\sum_{1 \leq i \leq n} w_i x_i$.

A linear threshold circuit has a linear threshold gate at its root. A language L is in the class LT_1 if there exists a family of nonuniform, depth-1 weighted linear threshold circuits defined by a family of weight vectors $\{\hat{w}_n\}_{n \geq 0}$ such that for all $x \in \{0, 1\}^*$, $x \in L$ if and only if $\hat{w}_{|x|} \cdot \hat{x} > 0$. Similarly, L is in the class ELT_1 if there exists a family of nonuniform, depth-1 exact weighted linear threshold circuits defined by a family of weight vectors $\{\hat{w}_n\}_{n \geq 0}$ such that for all $x \in \{0, 1\}^*$, $x \in L$ if and only if $\hat{w}_{|x|} \cdot \hat{x} = 0$.

Topologically, a linear threshold gate on n inputs describes a halfspace S in $\{0, 1\}^n$, and an exact linear threshold gate describes a hyperplane H in $\{0, 1\}^n$, where strings in $\{0, 1\}^n$ are viewed as binary vectors.

For more information on LT_1 and ELT_1 , we refer the reader to Agrawal and Arvind [1], from which we will make several useful extensions in Section 4.

2.2 Dimension and Learning

Resource-bounded dimension was introduced by Lutz [12] as a refinement of resource-bounded measure [10]. Each class X of languages has a p-dimension $\text{dim}_p(X) \in [0, 1]$, and if $\text{dim}_p(X) < 1$, then X has p-measure 0. In this paper we do not use the original definition of p-dimension, but instead the result that if X reduces to a learnable concept class family, then $\text{dim}_p(X) = 0$ [6]. For more information on measure and dimension we refer to [3, 7, 11, 14].

A concept is a set $C \subseteq U$ for some universe U , or equivalently, a function mapping U to $\{0, 1\}$. A concept class is a set \mathcal{C} of concepts. In our setting, U is simply the set of all binary strings $\{0, 1\}^*$ and a concept is a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$.

The online-mistake bound model of learning was introduced by Littlestone [9]. An online learner, given a concept class \mathcal{C} and a universe U , attempts to learn a target concept $C \in \mathcal{C}$. The learner is presented with a sequence of examples x_1, x_2, \dots in U . In the i^{th} stage, the learner must predict whether $x_i \in C$. The answer for x_i is then revealed, the learner may adjust its strategy, and the next concept x_{i+1} is presented for classification. The learner makes a mistake if it incorrectly classifies an example. The mistake bound of a learning algorithm for a concept class \mathcal{C} is the maximum over all $C \in \mathcal{C}$ of the number of mistakes made when learning C , over all possible sequences of examples. The running time of the learner is the time required to predict the classification of an example.

We now recall the definitions of reducibility to learnable concept classes from [6]. Let $L \subseteq \{0, 1\}^*$ and let $\mathcal{C} = (\mathcal{C}_n | n \in \mathbb{N})$ be a sequence of concept classes. For a time bound $r(n)$, we say L reduces to \mathcal{C} in $r(n)$ time if there is a reduction f computable in $O(r(n))$ time such that for infinitely many n , there is a concept $C_n \in \mathcal{C}_n$ such that for all $x \in \{0, 1\}^{\leq n}$, $x \in L$ if and only if $f(x, 1^n) \in C_n$. Note that the reduction is not required to hold for all n , but only infinitely many n .

Let $\mathcal{L}(t, m)$ be the set of all sequences of concept classes \mathcal{C} such that for each $C_n \in \mathcal{C}$, there is an algorithm that learns C_n in $O(t(n))$ time with mistake bound $m(n)$. Then the class $\mathcal{RL}(r, t, m)$ is the class of languages that reduce to some sequence of concept classes in $\mathcal{L}(t, m)$ in $r(n)$ time.

Theorem 2.1 (Hitchcock [6]). *For every $c \in \mathbb{N}$, the class $\mathcal{RL}(2^{cn}, 2^{cn}, o(2^n))$ has p-dimension 0.*

Because $X \subseteq Y$ implies $\dim_p(X) \leq \dim_p(Y)$, the task of proving that a class has p-dimension 0 can be reduced to showing the class is a subset of $\mathcal{RL}(2^{cn}, 2^{cn}, o(2^n))$ for some constant c .

2.3 Learning Algorithms

We make use of two learning algorithms. The first is the second of Littlestone’s Winnow algorithms [9], which can be used to learn Boolean r -of- k functions on n variables. In an r -of- k function there is a subset V of the n variables with $|V| \leq k$ such that the function evaluates to 1 if at least r of the variables in V are set to 1. Winnow2 has two parameters: a weight update multiplier α and a threshold value θ . Initially, each of the variables x_i has a weight $w_i = 1$. Winnow2 operates by predicting that an example x is in the concept if and only if $\sum_{1 \leq i \leq n} w_i x_i > \theta$. The weights are updated following each mistake by the following rubric:

- If Winnow2 incorrectly predicts that x is in the target concept, then for each x_i such that $x_i = 1$, set $w_i = w_i/\alpha$.
- If Winnow2 incorrectly predicts that x is not in the target concept, then for each x_i such that $x_i = 1$, set $w_i = \alpha \cdot w_i$.

Littlestone showed that for $\alpha = \frac{1}{2r}$ and $\theta = n$, Winnow2 has a mistake bound on learning r -of- k functions of $8r^2 + 5k + 14kr \ln n$. Winnow2 also classifies examples in polynomial time.

The second learning algorithm we use is Maass and Turán’s [16] algorithm for learning half-spaces. They first describe the Convex Feasibility Problem: given a separation oracle and a guarantee r for an unknown convex body P , find a point in P . By a guarantee, they mean a number such that the volume of the convex body P (in d dimensions) within the ball of radius r around $\hat{0}$ is at least r^{-d} .

Theorem 2.2 (Maass and Turán [16]). *Assume that there is an algorithm A^* solving the Convex Feasibility Problem with query complexity $q(d, \log r)$ (where q is a function of both the dimension d and the guarantee r) and time complexity $t(d, \log r)$. Then there is a learning algorithm A for learning a halfspace in d dimensions and n values such that the mistake bound of A is $q(d, 4d(\log d + \log n + 3)) + 1$ and the running time is at most $t(d, 4d(\log d + \log n + 3)) + q(d, 4d(\log d + \log n + 3)) \cdot p(d, \log n)$ for some polynomial p .*

Using Vaidya’s algorithm for learning convex bodies [17], which is an algorithm for the Convex Feasibility Problem, they show that learning a halfspace on d dimensions and n values (in our case, with the binary alphabet, $n = 2$) has a mistake bound of $O(d^2(\log d + \log n))$ and a polynomial running time.

3 Majority Reductions

We say that $A \leq_{\text{maj}}^P B$ if there is a polynomial-time computable function $f : \{0, 1\}^* \rightarrow \mathcal{P}(\{0, 1\}^*)$ such that for all $x \in \{0, 1\}^*$, $x \in A$ if and only if

$$|f(x) \cap B| \geq \frac{|f(x)|}{2}.$$

The following lemma is routine and says that if B is nondense, then we can assume that the majority reduction makes the same number of queries for all inputs of each length.

Lemma 3.1. *Let $A \in \text{P}_{\text{maj}}(\text{DENSE}^c)$. Then there exists a $B \in \text{DENSE}^c$, a majority reduction f computable in polynomial time, and a polynomial r such that for all $x \in \{0, 1\}^*$, $|f(x)| = r(|x|)$.*

We now prove our first main result.

Theorem 3.2. $\text{P}_{\text{maj}}(\text{DENSE}^c)$ has p-dimension 0.

Proof. It suffices to show that there is a concept class family $\mathcal{CF} \in \mathcal{L}(2^{cn}, o(2^n))$ and a reduction g computable in 2^{cn} time such that for all $A \in \text{P}_{\text{maj}}(\text{DENSE}^c)$, A reduces to \mathcal{CF} by g .

Let $A \in \text{P}_{\text{maj}}(\text{DENSE}^c)$. Then there is a $p(n)$ -time-bounded majority reduction f that makes exactly $r(n)$ queries for each n , and a set $B \in \text{DENSE}^c$ such that for all $x \in \{0, 1\}^*$, $x \in A$ if and only if $|f(x) \cap B| \geq \frac{|f(x)|}{2}$.

Let $Q_n = \bigcup_{|x| \leq n} f(x)$ be the set of all queries made by f up through length n . Then $|Q_n| \leq 2^{n+1}p(n)$. Enumerate Q_n as q_1, \dots, q_N . Then each subset $R \subseteq Q_n$ can be identified with its characteristic string $\chi_R \in \{0, 1\}^N$ according to this enumeration.

Let $M(n) = |Q_n \cap B|$ for each n and let $\delta \in (0, 1)$. Then $M(n) \leq |B_{\leq p(n)}| < 2^{n^\delta}$ for infinitely many n because B is nondense. Our target concept is the $\frac{r(n)}{2}$ -of- $M(n)$ threshold function $h : \{0, 1\}^N \rightarrow \{0, 1\}$ defined using the variable set $V = \{i \mid \chi_R[i] = 1\}$. Since for all $x \in \{0, 1\}^*$, $f(x)$ makes exactly $r(|x|)$ number of queries and $x \in A$ if and only if $|f(x) \cap B| \geq \frac{r(|x|)}{2}$, we have $x \in A$ if and only if $h(\chi_{f(x)}) = 1$.

Given x , $\chi_{f(x)}$ can be computed in $O(2^{2n})$ time, and thus Winnow2 can classify examples in $O(2^{2n})$ time. Whenever $M(n) < 2^{n^\delta}$, Winnow2 makes at most $2r^2(n) + 5 \cdot 2^{n^\delta} + 7 \cdot 2^{n^\delta} r(n) \ln 2^{n+1}p(n) = o(2^n)$ mistakes while learning h . Thus $\text{P}_{\text{maj}}(\text{DENSE}^c) \subseteq \mathcal{RL}(2^{2n}, 2^{2n}, o(2^n))$ and the theorem follows by Theorem 2.1. \square

We remark that as r -of- k threshold functions are a special case of halfspaces, we could also use the halfspace learning algorithm instead of Winnow2 to prove Theorem 3.2.

As $\text{P}_{\text{dtt}}(\text{DENSE}^c) \subseteq \text{P}_{\text{maj}}(\text{DENSE}^c)$ and $\text{P}_{\text{ctt}}(\text{DENSE}^c) \subseteq \text{P}_{\text{maj}}(\text{DENSE}^c)$, Theorem 3.2 subsumes two results from [6]. We also have the following corollary answering Fu's question [5] about hard sets for exponential time.

Corollary 3.3. $\text{E} \not\subseteq \text{P}_{\text{maj}}(\text{DENSE}^c)$. That is, every $\leq_{\text{maj}}^{\text{P}}$ -hard set for E is dense.

4 Iterated Reductions

Our proof that $\text{P}_{\alpha \log n - \text{tt}}(\text{P}_{\text{ctt}}(\text{DENSE}^c))$ has p-dimension zero follows the proof technique of Agrawal and Arvind [1] that

$$\text{P}_{\text{btt}}(\text{P}_{\text{ctt}}(\text{SPARSE})) \subseteq \text{P}_{\text{m}}(\text{LT}_1)$$

to reduce the class to a family of halfspaces. We then use Maass and Turán's [16] learning algorithm to learn these halfspaces. As long as the reduction runs in 2^{n^α} time for some $\alpha < 1$, the halfspaces have subexponential size and the mistake bound is $2^{o(n)}$. Therefore by Theorem 2.1,

$$\dim_{\text{p}}(\text{R}_{\text{m}}^{2^{n^\alpha}}(\text{LT}_1)) = 0, \tag{4.1}$$

where $\text{R}_{\text{m}}^{t(n)}(\text{LT}_1)$ is the class of languages that reduce to some LT_1 language by a many-one reduction that runs in time $O(t(n))$.

Instead of $P_{\alpha \log n - \text{tt}}(P_{\text{ctt}}(\text{DENSE}^c))$ we will focus first on the smaller class

$$P_{\text{btt}}(P_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c)).$$

The benefit is that the nondense sets will be small almost everywhere rather than infinitely often, and the bounded queries permit us to make simpler alterations to Agrawal and Arvind's lemmas. We will then show how to extend these results first to the $\alpha \log n - \text{tt}$ case and then to the nondense case.

The following technical lemma by Agrawal and Arvind is useful in that it permits us to assume that any many-one reduction to threshold circuits could be adapted so that for all $x, y \in \{0, 1\}^n$, $|f(x)| = |f(y)|$. This simplifies arguments by circumventing the possibility that a reduction might map each $x \in \{0, 1\}^n$ to a different threshold circuit.

Lemma 4.1. *Let $A \in P_{\text{m}}(\text{ELT}_1)$ (resp. $A \in P_{\text{m}}(\text{LT}_1)$). Then there exist $L \in \text{ELT}_1$ (resp. $L \in \text{LT}_1$), an FP function f , and a polynomial r , such that for every x , for every $n \geq r(|x|)$, $x \in A$ iff $f(x, 1^n) \in H(\hat{w}_n)$ (resp. $f(x, 1^n) \in S_+(\hat{w}_n)$), where \hat{w}_n are the weight vectors associated with L .*

Because we will deal with reductions running in superpolynomial time, Lemma 4.1 is insufficient for our needs. However, we observe that the restriction to a polynomial time bound can be relaxed, as the proof of Lemma 4.1 only utilizes the closure of polynomials under multiplication and composition with other polynomials. Thus we can extend this result to any family of time bounds closed under the same operations.

Lemma 4.2. *Let Δ be a family of computable functions that is closed under multiplication and composition with polynomials, and let $F\Delta$ be the functional class with bounds in Δ . Then Lemma 4.1 holds with $f \in F\Delta$ and $r \in \Delta$.*

Proof. Let $A \in R_{\text{m}}^{\Delta}(\text{ELT}_1)$. Then there is an $F\Delta$ function g and $B \in \text{ELT}_1$ such that $x \in A$ iff $g(x) \in B$. Let $\{\hat{d}_n\}_{n \geq 0}$ be the weight vectors associated with B .

The idea is, for each n , run together consecutive weight vectors associated with B to form a single weight vector, and then use a sliding window to focus the original reduction onto the proper place in the new vector. As long as the new reduction produces a large enough query, the new weight vector, and every weight vector larger than it, is guaranteed to contain the original weight vector.

Specifically, we define a new family of weight vectors $\{\hat{c}_n\}_{n \geq 0}$ as follows. Let m be the largest integer such that $n \geq \frac{1}{2}m(m+1)$, and let $\hat{c}_n = (\hat{d}_1, \hat{d}_2, \dots, \hat{d}_m, \hat{0}_{n-m(m+1)/2})$. It is clear that these weight vectors define a language $L \in \text{ELT}_1$.

Let $r' \in \Delta$ such that for any $x \in \{0, 1\}^*$, $|g(x)| \leq r'(x)$. Let $r \in \Delta$ such that $r(n) \geq \frac{1}{2}r'(n)(r'(n) + 1)$ for all $n \geq 0$. Since Δ is closed under multiplication, we know such an r exists. For $x \in \{0, 1\}^*$ such that $|g(x)| = i$, define $f(x, 1^n) = (\hat{0}_{i(i-1)/2}, g(x), \hat{0}_{n-i(i+1)/2})$ for all $n \geq r(|x|)$. Then we have that $x \in A$ iff $f(x, 1^n) \in L$ for every $n \geq r(|x|)$, which is essentially the statement of the lemma.

The case for LT_1 is proved the same way. □

The following corollary captures the full power of this lemma. We say that a time bound $t(n)$ is *subexponential* if for all $\epsilon > 0$, $t(n) < 2^{n^\epsilon}$ for all sufficiently large n . We write se for the class of all subexponential time bounds.

Corollary 4.3. *Let $A \in \mathbb{R}_m^{\text{se}}(\text{ELT}_1)$ (similarly, $A \in \mathbb{R}_m^{\text{se}}(\text{LT}_1)$). Then there is a subexponential-time function f and a language $B \in \text{ELT}_1$ ($B \in \text{LT}_1$) such that for every $n \in \mathbb{N}$, for all $x, y \in \{0, 1\}^{\leq n}$, for all $q_i \in f(x, 1^n)$ and $q_j \in f(y, 1^n)$, $|q_i| = |q_j|$.*

Agrawal and Arvind showed that $\text{SPARSE} \subseteq \mathbb{P}_m(\text{ELT}_1)$. We extend this in the following lemma.

Lemma 4.4. $\text{DENSE}_{\text{i.o.}}^c \subseteq \mathbb{R}_m^{\text{se}}(\text{ELT}_1)$.

Proof. Let $S \in \text{DENSE}_{\text{i.o.}}^c$. Then for all but finitely many n , $|S_{\leq n}| \leq 2^{n^\epsilon}$ for all $\epsilon > 0$. Let $S_{=n} = \{s_{n,1}, s_{n,2}, \dots, s_{n,m(n)}\}$, where $m(n) \leq t(n)$ and t is a subexponential function. Let the string $s_{n,i}$ also stand for the natural number representing the lexicographic rank of the string $s_{n,i}$ in $\{0, 1\}^n$ for every n and $1 \leq i \leq m(n)$. Define $T_n(z)$ to be the polynomial $\prod_{i=1}^{m(n)} (z - s_{n,i})$. Clearly, $T_n(z)$ is a polynomial in z of degree bounded by $t(n)$. For each string $z \in \{0, 1\}^n$, we overload notation and write $T_n(z)$ to mean T_n applied to the lexicographic index of z in $\{0, 1\}^n$. Then for each $z \in \{0, 1\}^n$, we have $z \in S$ iff $T_n(z) = 0$.

Rewriting $T_n(z)$, we have $T_n(z) = \sum_{1 \leq j \leq t(n)} a_j z^j$. For $1 \leq j \leq t(n)$, we can write z^j as $\sum_{1 \leq r \leq n \cdot t(n)} 2^r y_{j,r}$, where the $y_{j,r}$ essentially denotes the bits in the binary representation of z^j . Thus it follows that $T_n(z)$ can be rewritten as a linear combination $\sum_{1 \leq j \leq t(n)} \sum_{1 \leq r \leq n \cdot t(n)} w_{j,r} y_{j,r}$ of the bits $y_{j,r}$ defined above.

Now we can define a language $L \in \text{ELT}_1$ using these linear functions to define the corresponding weighted exact threshold gates in the circuit family accepting L . As there will be $n \cdot t(n)^2$ variables, which is subexponential, we have that $S \leq_m^{\text{se}} L$. \square

The notion of using the coefficients of a multivariate polynomial as the weights of threshold circuit is one that appears often in the proofs of Agrawal and Arvind, and thus in our extension of their proofs. To formalize this, Agrawal and Arvind made use of the following technical lemma.

Lemma 4.5. *Let $\{F_n(\hat{x})\}_{n \geq 1}$, $F_n(\hat{x})$ defined over \mathbb{Q}^n , be a family of degree k multinomials (for a constant $k > 0$). Let the family of weight vectors $\{\hat{c}_n\}_{n > 0}$ and the FP function f be such that for every $\hat{x} \in \mathbb{Q}^n$, $F_n(\hat{x}) = \hat{c}_n \cdot f(\hat{x})$ where $f(\hat{x}) \in \mathbb{Q}^m$. Then the function f reduces the set*

$$A = \bigcup_{n \geq 1} \{x \in \{0, 1\}^* \mid F_n(x) = 0\}$$

to the set in ELT_1 defined by weight vectors $\{\hat{c}_n\}_{n > 0}$. Also, f reduces the set

$$B = \bigcup_{n \geq 1} \{x \in \{0, 1\}^* \mid F_n(x) > 0\}$$

to the set in LT_1 defined by weight vectors $\{\hat{c}_n\}_{n > 0}$ (where a string x of length n is interpreted as an n -dimensional 0-1 vector when it is an argument to F_n).

We note that Lemma 4.5 is sufficient for most of our proofs. As long as the degree of the multinomials remains bound by a constant, the closure of subexponential functions under composition with polynomials will guarantee that even a multinomial of subexponential size will reduce to a threshold circuit of subexponential size. As we will eventually concern ourselves with multinomials of unbounded degree, we provide a sketch of the proof to provide an intuition in how to extend these results.

Proof Sketch of Lemma 4.5. In polynomial time, we can rewrite F_n as a sum of monomials. A simple counting argument will yield that for a k -degree multinomial, there are $\binom{n+k}{k}$ such monomials. To construct a threshold circuit, it suffices to order the monomials and use the coefficients of each monomial for the weights. □

Lemma 4.6. $R_{\text{ctt}}^{\text{se}}(\text{ELT}_1) = R_{\text{m}}^{\text{se}}(\text{ELT}_1)$.

Proof. Let A be a set that is conjunctively reducible to some set $B \in \text{ELT}_1$. Then there is an se-computable function f such that for every $x \in \{0, 1\}^*$, $f(x)$ is a list of queries such that $x \in A$ iff for every q in $f(x)$, $q \in B$. Using Corollary 4.3, there exist $B' \in \text{ELT}_1$ defined by a family of weight vectors $\{\hat{c}_n\}_{n \geq 1}$, an se-computable function g , and a subexponential function r such that for every x , for every $j \geq r(|x|)$, $x \in A$ iff $g(x, 1^j) \subseteq H(\hat{c}_j)$.

Since f is a conjunctive reduction, there is a subexponential p such that for every x , $g(x, 1^{r(|x|)})$ has exactly $p(|x|)$ queries (this can be achieved simply by repeating the last query a suitable number of times). Define

$$F_{p(n)r(n)}(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_{p(n)}) = \sum_{i=1}^{p(n)} (\hat{c}_{r(n)} \cdot \hat{q}_i)^2$$

where $\hat{q}_i \in \{0, 1\}^{r(n)}$ for $1 \leq i \leq p(n)$. The set L is defined as

$$L = \bigcup_{n \geq 1} \left\{ x \in \{0, 1\}^{p(n)r(n)} \mid F_{p(n)r(n)}(x) = 0 \right\}.$$

Note that as an argument to $F_{p(n)r(n)}$, x is interpreted as a 0-1 vector.

It is easy to see that $x \in A$ iff $(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_{p(|x|)}) \in L$ where $g(x, 1^{r(|x|)}) = \{\hat{q}_1, \dots, \hat{q}_{p(|x|)}\}$. Lemma 4.5 implies that L is in $R_{\text{m}}^{\text{se}}(\text{ELT}_1)$. □

To show $P_{\text{btt}}(\text{ELT}_1) \subseteq P_{\text{m}}(\text{LT}_1)$, Agrawal and Arvind divide a k -tt reduction into each separate condition η , and then note that

$$P_{\eta}(A) \subseteq P_{\text{b}\oplus}(P_{\text{bc}}(P_{1\text{-tt}}(A))),$$

where $P_{\text{b}\oplus}$ is the closure under the bounded parity reduction and P_{bc} is the closure under the bounded conjunctive reduction. They then show that

$$P_{\text{b}\oplus}(P_{\text{bc}}(P_{1\text{-tt}}(\text{ELT}_1))) \subseteq P_{\text{m}}(\text{LT}_1),$$

and finish their proof by showing $P_{\text{m}}(\text{LT}_1)$ is closed under the join operation, i.e. it is possible to create a linear threshold circuit from all 2^{2^k} k -tt conditions in polynomial time. We proceed in the same fashion.

Lemma 4.7. $R_{\text{b}\oplus}^{\text{se}}(\text{LT}_1) = R_{\text{m}}^{\text{se}}(\text{LT}_1)$.

Proof sketch. We form a multinomial using the queries produced by the bounded parity reduction. For each x , we assume there will be exactly k queries for some odd constant k , each of size n , and we assume that for $A \in \text{LT}_1$ with associated weight vectors $\{\hat{w}_n\}_{n \geq 0}$, $x \in A$ iff $\hat{w}_{|x|} \cdot \hat{x} > 0$, and $x \notin A$ iff $\hat{w}_{|x|} \cdot \hat{x} < 0$. (For a proof of this, see [1], Lemma 3.2.) Let

$$F_{kn}(\hat{x}_1, \dots, \hat{x}_k) = (\hat{w}_n \cdot \hat{x}_1) \cdots (\hat{w}_n \cdot \hat{x}_k).$$

This multinomial preserves the parity as $F_{kn}(\hat{x}_1, \dots, \hat{x}_k) > 0$ iff an odd number of the queries satisfy $\hat{w}_n \cdot \hat{x}_i > 0$. Furthermore, the size of the input to F_{kn} can be computed in subexponential time. As F_{kn} is a degree k multinomial, we can use Lemma 4.5 to finish the proof. \square

Lemma 4.8. $R_{\text{bd}}^{\text{se}}(\text{ELT}_1) = R_{\text{m}}^{\text{se}}(\text{ELT}_1)$.

Proof. The proof uses the exact same multinomial as in Lemma 4.7. Disjunction is preserved in that $F_{kn}(\hat{x}_1, \dots, \hat{x}_k) = 0$ so long there exists an i such that $\hat{w}_n \cdot \hat{x}_i = 0$. \square

For the next two lemmas, we use a special reduction defined by Agrawal and Arvind. A τ -reduction is a 2-tt reduction where $X \leq_{\tau} Y$ if on input x , the reduction produces two queries q_1 and q_2 such that $x \in X$ if and only if $q_1 \in Y$ and $q_2 \notin Y$.

Lemma 4.9. $R_{\tau}^{\text{se}}(\text{ELT}_1) \subseteq R_{\text{m}}^{\text{se}}(\text{LT}_1)$.

Proof. Using the queries q_1 and q_2 produced by the τ -reduction in reducing $A \in R_{\tau}^{\text{se}}(\text{ELT}_1)$ to $B \in \text{ELT}_1$, the following degree 4 multinomial suffices to prove the lemma:

$$F_{2n}(\hat{q}_1, \hat{q}_2) = (1 - 2(l \cdot \hat{q}_1 \cdot \hat{w}_n)^2)(\hat{q}_2 \cdot \hat{w}_n)^2,$$

where \hat{w}_n is a weight vector associated with B , and l is the least common multiplier of the denominators of the weights in \hat{w}_n . \square

Lemma 4.10. $P_{\text{bc}}(R_{1-\text{tt}}^{\text{se}}(\text{ELT}_1)) \subseteq R_{\tau}^{\text{se}}(\text{ELT}_1)$.

Proof. For $A \in P_{\text{bc}}(R_{1-\text{tt}}^{\text{se}}(\text{ELT}_1))$, there is a reduction f to $B \in \text{ELT}_1$ with k queries such that $x \in A$ iff m of the k queries are in B and $k-m$ are not in B . We label these queries $\hat{q}_1, \hat{q}_2, \dots, \hat{q}_m$ and $\hat{r}_1, \hat{r}_2, \dots, \hat{r}_{k-m}$. Thus $x \in A$ iff $\hat{q}_i \in B$ for all $1 \leq i \leq m$ and no $\hat{r}_j \in B$ for all $1 \leq j \leq k-m$. We can look at this as the combination of a bounded conjunctive reduction and a bounded disjunctive reduction, both requiring subexponential time. By Lemma 4.6 and Lemma 4.8, we can alter these reductions to a single query each to a language $B' \in \text{ELT}_1$. Call these single queries \hat{q} and \hat{r} . Then $x \in A$ iff $\hat{q} \in B'$ and $\hat{r} \notin B'$. This transformation can be carried out in subexponential time, so the lemma follows. \square

We are ready to prove the simplified version of our main result.

Theorem 4.11. $P_{\text{btt}}(P_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c))$ has p-dimension 0.

Proof. Through Lemma 4.4, Lemma 4.6, Lemma 4.10, and Lemma 4.7 respectively, the following holds for each truth-table condition η :

$$\begin{aligned} P_{\eta}(P_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c)) &\subseteq P_{\text{b}\oplus}(P_{\text{bc}}(P_{1-\text{tt}}(P_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c)))) \\ &\subseteq P_{\text{b}\oplus}(P_{\text{bc}}(P_{1-\text{tt}}(P_{\text{ctt}}(R_{\text{m}}^{\text{se}}(\text{ELT}_1)))))) \\ &\subseteq P_{\text{b}\oplus}(P_{\text{bc}}(P_{1-\text{tt}}(R_{\text{ctt}}^{\text{se}}(\text{ELT}_1)))) \\ &\subseteq P_{\text{b}\oplus}(P_{\text{bc}}(P_{1-\text{tt}}(R_{\text{m}}^{\text{se}}(\text{ELT}_1)))) \\ &\subseteq P_{\text{b}\oplus}(P_{\text{bc}}(R_{1-\text{tt}}^{\text{se}}(\text{ELT}_1))) \\ &\subseteq P_{\text{b}\oplus}(R_{\tau}^{\text{se}}(\text{ELT}_1)) \\ &\subseteq R_{\text{m}}^{\text{se}}(\text{LT}_1). \end{aligned}$$

For k queries, there are 2^{2^k} truth-table conditions. Through the reduction above, each condition corresponds to a different set of weights $\hat{c}_{n,j}$, $1 \leq j \leq 2^{2^k}$, defining a threshold circuit that is subexponential in size. Let us say this size is $s(n)$. Thus we can create a single linear threshold circuit with weights $\hat{d}_n = (\hat{c}_{n,1}, \hat{c}_{n,2}, \dots, \hat{c}_{n,2^{2^k}})$ as the join of all these individual circuits. The size of this new circuit is $2^{2^k} \cdot s(n) \leq 2^{n^\delta}$ for some δ such that $0 < \delta < 1$. Let $L \in \text{LT}_1$ be the set defined by the weight vectors $\{\hat{d}_n\}$.

Let $A \in \text{P}_{\text{btt}}(\text{P}_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c))$. Let g be the k -tt reduction, and suppose that $g(x)$ uses the condition which corresponds to $\hat{c}_{|x|,j}$. Let \hat{q}_x be the many-one query corresponding to that condition produced by the reduction above. Then the reduction f mapping A to L is defined by

$$f(x) = (\hat{0}_{(j-1)s(|x|)}, \hat{q}_x, \hat{0}_{(2^{2^k} - (j+1))s(|x|)}).$$

Then $x \in A$ iff $f(x) \in L$. It follows that

$$\text{P}_{\text{btt}}(\text{P}_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c)) \subseteq \text{R}_{\text{m}}^{2^{n^\delta}}(\text{LT}_1),$$

which yields the theorem by (4.1). □

The following extension shows how to handle the case of $\alpha \log n$ many queries.

Theorem 4.12. $\text{P}_{\alpha \log n\text{-tt}}(\text{P}_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c))$ has p -dimension 0.

Proof. With $\alpha \log n$ queries, we are challenged to show that we can use the general construction described above while still keeping a time bound of $2^{o(n)}$. The main difference lies in that, for each truth table condition η , $\text{P}_\eta(C)$ becomes $\text{P}_{n^\alpha \oplus}(\text{P}_{\alpha \log n\text{-c}}(\text{P}_{1\text{-tt}}(C)))$. Thus the multinomials used in Lemmas 4.7, 4.8, and 4.10 are not of constant degree.

We can still show $\text{P}_{\alpha \log n\text{-c}}(\text{P}_{1\text{-tt}}(\text{ELT}_1)) \subseteq \text{R}_{\text{r}}^{\text{se}}(\text{ELT}_1)$. We need only handle the portion dealing with the disjunctive query, for we can already handle unbounded number of conjunctive queries to a language in ELT_1 . In an extension of Lemma 4.8, we use the reduction described in Lemma 4.5, extended to handle multinomials of unbounded degree, to create a single query of size

$$\binom{2^{n^\epsilon} + \alpha \log n}{\alpha \log n} \leq (2^{n^\epsilon} + \alpha \log n)^{\alpha \log n} \leq (2^{2n^\epsilon})^{\alpha \log n} \leq 2^{2\alpha n^\epsilon \log n}$$

which is still subexponential.

We cannot exactly show that $\text{P}_{n^\alpha \oplus}(\text{LT}_1) \subseteq \text{R}_{\text{m}}^{\text{se}}(\text{LT}_1)$, but the bounds we make are sufficient. Using a similar method as in the proof of Lemma 4.7 the resulting query to a linear threshold circuit has size

$$\binom{2^{n^\epsilon} + n^\alpha}{n^\alpha} \leq (2^{n^\epsilon} + n^\alpha)^{n^\alpha} \leq (2^{2n^\epsilon})^{n^\alpha} \leq 2^{2n^{\alpha+\epsilon}}$$

which, for sufficiently small ϵ is $2^{o(n)}$.

The join then proceeds as above. We note that the size of the final query is bounded by $2^{n^\alpha} \cdot 2^{n^{\alpha+\epsilon}}$ for sufficiently small ϵ , which remains $2^{o(n)}$. □

Now we are prepared to state our main result.

Theorem 4.13 (Main Theorem). $P_{\alpha \log n\text{-tt}}(P_{\text{ctt}}(\text{DENSE}^c))$ has p-dimension 0.

Proof. We note that the construction to show that $P_{\alpha \log n\text{-tt}}(P_{\text{ctt}}(\text{DENSE}_{\text{i.o.}}^c))$ has p-dimension 0 describes a reduction f that runs in time $2^{n^{k\epsilon}}$, for some $\epsilon > 0$ related to the density of the nondense sets involved and k is a constant determined by the iterated reductions. Since a language $L \in \text{DENSE}_{\text{i.o.}}^c$ has density $|L_{\leq n}| \leq 2^{n^{\frac{1}{2k}}}$ for all but finitely many n , we can easily assume that $\epsilon < \frac{1}{2k}$, which satisfies our requirement that the reduction to a linear threshold circuit require subexponential time.

The reduction f always carries out a specific construction while making the assumption that the density of $L_{\leq n}$ is always bounded by 2^{n^ϵ} for some fixed ϵ . As long as this assumption is met, the reduction is correct.

In the broader case, we can only count on the nondense language being sufficiently sparse infinitely often. The $\alpha \log n\text{-tt}$ reduction to a nondense set can be viewed as a reduction g running in time n^d for some constant d . Thus a query to a nondense set can be of size n^d , meaning that we have to worry about the density of $L_{\leq n^d}$, not simply $L_{\leq n}$. The concern is that while for every $\epsilon > 0$, $|L_{\leq n}| < 2^{n^\epsilon}$ infinitely often, it is sufficiently dense at each $L_{\leq n^d}$ to make the reduction f fail for all but finitely many cases.

However, we recall that for $L \in \text{DENSE}^c$, for all $\delta > 0$, for infinitely many n , $|L_{\leq n}| < 2^{n^\delta}$. It is easy to show that, for sufficiently small δ , that if $|L_{\leq m}| < 2^{n^\delta}$ for $n^d \leq m \leq (n+1)^d$, then $|L_{\leq n^d}| < 2^{(n^d)^\epsilon}$. Since there must be infinitely many such m , it holds that $|L_{\leq n^d}| < 2^{(n^d)^\epsilon}$ infinitely often, which is sufficient to show p-dimension 0. □

Corollary 4.14. $E \not\subseteq P_{\alpha \log n\text{-tt}}(P_{\text{ctt}}(\text{DENSE}^c))$.

Theorem 4.13 gives the answer to Lutz and Mayordomo's question [13].

Corollary 4.15. $P_{\text{btt}}(P_{\text{ctt}}(\text{DENSE}^c))$ has measure 0 in E .

5 Conclusion

We conclude with a brief remark about the density of hard sets for NP. If NP has positive p-dimension, then it follows from our results that

$$\text{NP} \not\subseteq P_{\text{maj}}(\text{DENSE}^c)$$

and

$$\text{NP} \not\subseteq P_{\alpha \log n\text{-tt}}(P_{\text{ctt}}(\text{DENSE}^c))$$

for all $\alpha < 1$. These conclusions are stronger than what is known from the hypothesis $P \neq \text{NP}$. If $P \neq \text{NP}$, then $\text{NP} \not\subseteq P_{\text{btt}}(P_{\text{ctt}}(\text{SPARSE}))$ [2], but nothing is known about majority reductions.

One direction for further research is to improve the $\alpha \log n\text{-tt}$ bound in Theorem 4.13. Can the bound be improved to $n^\alpha\text{-tt}$? Or ideally, to subsume the main result in [6], can it be improved to $n^\alpha\text{-T}$? A more basic direction is to find further applications of learning algorithms in resource-bounded measure and dimension.

References

- [1] M. Agrawal and V. Arvind. Geometric sets of low information content. *Theoretical Computer Science*, 158(1–2):193–219, 1996.
- [2] V. Arvind, Y. Han, L. Hemachandra, J. Köbler, A. Lozano, M. Mundhenk, M. Ogiwara, U. Schöning, R. Silvestri, and T. Thierauf. Reductions to sets of low information content. In K. Ambos-Spies, S. Homer, and U. Schöning, editors, *Complexity Theory: Current Research*, pages 1–45. Cambridge University Press, 1993.
- [3] K. B. Athreya, J. M. Hitchcock, J. H. Lutz, and E. Mayordomo. Effective strong dimension in algorithmic information and computational complexity. *SIAM Journal on Computing*, 37(3):671–705, 2007.
- [4] B. Fu. With quasilinear queries EXP is not polynomial time Turing reducible to sparse sets. *SIAM Journal on Computing*, 24(5):1082–1090, 1995.
- [5] B. Fu. Personal communication, 2006.
- [6] J. M. Hitchcock. Online learning and resource-bounded dimension: Winnow yields new lower bounds for hard sets. *SIAM Journal on Computing*, 36(6):1696–1708, 2007.
- [7] J. M. Hitchcock, J. H. Lutz, and E. Mayordomo. The fractal geometry of complexity classes. *SIGACT News*, 36(3):24–38, September 2005.
- [8] W. Lindner, R. Schuler, and O. Watanabe. Resource-bounded measure and learnability. *Theory of Computing Systems*, 33(2):151–170, 2000.
- [9] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1987.
- [10] J. H. Lutz. Almost everywhere high nonuniform complexity. *Journal of Computer and System Sciences*, 44(2):220–258, 1992.
- [11] J. H. Lutz. The quantitative structure of exponential time. In L. A. Hemaspaandra and A. L. Selman, editors, *Complexity Theory Retrospective II*, pages 225–254. Springer-Verlag, 1997.
- [12] J. H. Lutz. Dimension in complexity classes. *SIAM Journal on Computing*, 32(5):1236–1259, 2003.
- [13] J. H. Lutz and E. Mayordomo. Measure, stochasticity, and the density of hard languages. *SIAM Journal on Computing*, 23(4):762–779, 1994.
- [14] J. H. Lutz and E. Mayordomo. Twelve problems in resource-bounded measure. *Bulletin of the European Association for Theoretical Computer Science*, 68:64–80, 1999. Also in *Current Trends in Theoretical Computer Science: Entering the 21st Century*, pages 83–101, World Scientific Publishing, 2001.
- [15] J. H. Lutz and Y. Zhao. The density of weakly complete problems under adaptive reductions. *SIAM Journal on Computing*, 30(4):1197–1210, 2000.

- [16] W. Maass and G. Turán. How fast can a threshold gate learn? In S. J. Hanson, G. A. Drastal, and R. L. Rivest, editors, *Computational Learning Theory and Natural Learning Systems, Vol. I: Constraints and Prospects*, pages 381–414. MIT Press, 1994.
- [17] P. M. Vaidya. A new algorithm for minimizing convex functions over convex sets. In *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science*, pages 338–349. IEEE Computer Society, 1989.
- [18] O. Watanabe. Polynomial time reducibility to a set of small density. In *Proceedings of the Second Structure in Complexity Theory Conference*, pages 138–146. IEEE Computer Society, 1987.