

The Complexity and Distribution of Hard Problems

TR 92-23

David W. Juedes and Jack H. Lutz

August 13, 1992

Iowa State University of Science and Technology
Department of Computer Science
226 Atanasoff
Ames, IA 50011

The Complexity and Distribution of Hard Problems^{*}

David W. Juedes and Jack H. Lutz
Department of Computer Science
Iowa State University
Ames, IA 50011

Abstract

Measure-theoretic aspects of the \leq_m^P -reducibility structure of the exponential time complexity classes $E = \text{DTIME}(2^{\text{linear}})$ and $E_2 = \text{DTIME}(2^{\text{polynomial}})$ are investigated. Particular attention is given to the *complexity* (measured by the size of complexity cores) and *distribution* (abundance in the sense of measure) of languages that are \leq_m^P -hard for E and other complexity classes.

Tight upper and lower bounds on the size of complexity cores of hard languages are derived. The upper bounds say that the \leq_m^P -hard languages for E are *unusually simple*, in the sense that they have smaller complexity cores than most languages in E . It follows that the \leq_m^P -complete languages for E form a measure 0 subset of E (and similarly in E_2).

This latter fact is seen to be a special case of a more general theorem, namely, that *every* \leq_m^P -degree (e.g., the degree of all \leq_m^P -complete languages for NP) has measure 0 in E and in E_2 .

1 Introduction

A decision problem (i.e., language) $A \subseteq \{0,1\}^*$ is said to be *hard* for a complexity class \mathcal{C} if every language in \mathcal{C} is efficiently reducible to A . If A is also an element of \mathcal{C} , then A is *complete* for \mathcal{C} . The most common interpretation of “efficiently reducible” here is “polynomial time many-one reducible,” abbreviated “ \leq_m^P -reducible.” (See section 2 for notation and terminology used in this introduction.) For example, in most usages, “NP-complete” means

^{*}This research was supported in part by National Science Foundation Grants CCR-8809238 and CCR-9157382, with matching funds from Rockwell International, and in part by DIMACS, where the second author was a visitor while part of this work was carried out.

“ \leq_m^P -complete for NP,” the completeness notion introduced by Karp [Kar72] and Levin [Lev73].

Decision problems that are \leq_m^P -hard for NP are *presumably intractable*, since they cannot be decided in polynomial time if $P \neq NP$. Decision problems that are \leq_m^P -hard for the exponential time complexity class $E = \text{DTIME}(2^{\text{linear}})$ are *provably intractable* because (i) they cannot be decided in polynomial time if $P \neq E$; and (ii) it has been proven, via diagonalization [HS65], that $P \neq E$. Problems that are \leq_m^P -hard (in fact, \leq_m^P -complete) for E have been exhibited by Stockmeyer and Chandra [SC79] and others.

It should be noted that a language is \leq_m^P -hard for E if and only if it is \leq_m^P -hard for the larger exponential complexity class $E_2 = \text{DTIME}(2^{\text{polynomial}})$. (This follows immediately from the fact that E_2 is the downward closure of E under the reducibility \leq_m^P .)

In this paper, we investigate the *complexity* (measured by size of complexity cores) and *distribution* (i.e., abundance in the sense of measure[†]) of languages that are \leq_m^P -hard for E (equivalently, E_2) and other complexity classes, including NP. We give tight lower bounds and, perhaps surprisingly, tight upper bounds on the sizes of complexity cores of hard languages. More generally, we analyze measure-theoretic aspects of the \leq_m^P -reducibility structure of exponential time complexity classes. We prove that \leq_m^P -hard problems are rare, in the sense that they form a p-measure 0 set; and that every \leq_m^P -degree has measure 0 in exponential time.

Complexity cores, first introduced by Lynch [Lyn75] have been studied extensively [Du85, ESY85, Orp86, OS86, BD87, Huy87, RO87, BDR88, DB89, Ye90, etc.]. Intuitively, a complexity core of a language A is a fixed set K of inputs such that *every* machine whose decisions are consistent with A fails to decide efficiently on all but finitely many elements of K . The meaning of “efficiently” is a parameter of the definition that varies according to the context. (See section 4 for a precise definition.)

Orponen and Schöning [OS86] have established two lower bounds on the sizes of complexity cores of hard languages. First, every \leq_m^P -hard language for E has a dense P-complexity core. Second, if $P \neq NP$, then every \leq_m^P -hard language for NP has a non-sparse polynomial complexity core.

In section 4 below, we extend the first of these results to languages that are weakly \leq_m^P -hard for E . (A language A is \leq_m^P -hard for E if every element of E is \leq_m^P -reducible to A . A language A is *weakly* \leq_m^P -hard for E

[†]I.e., *resource-bounded measure* as developed by Lutz [Lut92a] and described in section 3 of the present paper.

if any nonnegligible (i.e., non-measure 0) set of languages in E is reducible to A . Lutz has conjectured that “weakly \leq_m^P -hard” is more general than “ \leq_m^P -hard”, but this has not been proven.) Specifically, we prove that every language that is weakly \leq_m^P -hard for E or E_2 has a dense exponential complexity core. It follows that, if NP does not have measure 0 in E or E_2 (a hypothesis conjectured by Lutz to be true), then every \leq_m^P -hard language for NP has a dense exponential complexity core. This conclusion is much stronger than Orponen and Schöning’s conclusion that every such language has a non-sparse polynomial complexity core, though it is achieved at the cost of a stronger hypothesis.

In section 5 we investigate the resource-bounded measure of the lower \leq_m^P -spans, the upper \leq_m^P -spans, and the \leq_m^P -degrees of languages in E and E_2 . (The *lower* \leq_m^P -*span* of A is the set of all languages that are \leq_m^P -reducible to A . The *upper* \leq_m^P -*span* of A is the set of all languages to which A is reducible. The \leq_m^P -*degree* of A is the intersection of these two spans.) We prove the Small Span Theorem, which says that, if A is in E or E_2 , then at least one of the upper and lower spans must have resource-bounded measure 0. This implies that the \leq_m^P -hard languages for E form a set of p-measure 0. It also implies that *every* \leq_m^P -degree (e.g., the degree of all \leq_m^P -complete languages for NP) has measure 0 in E and in E_2 .

Languages that are \leq_m^P -hard for E are typically considered “at least as complex as” any element of E . Very early, Berman [Ber76] established limits to this interpretation by proving that no \leq_m^P -complete language is P-immune, even though E contains P-immune languages. (In fact, Mayor-domo [May92] has recently shown that almost every language in E is P-bi-immune.) In section 6 below we prove a much stronger limitation on the complexity of \leq_m^P -hard languages for E . We prove that every \leq_m^P -hard language for E is decidable in $\leq 2^{4n}$ steps on a dense set of inputs which is also decidable in $\leq 2^{4n}$ steps. This implies that *every* $\text{DTIME}(2^{4n})$ -complexity core of *every* \leq_m^P -hard language for E has a *dense complement*. Since almost every language in E has $\{0, 1\}^*$ as a $\text{DTIME}(2^{4n})$ -complexity core (as proven in section 4), this says that \leq_m^P -hard languages for E are *unusually simple*, in that they have *unusually small* complexity cores. Intuitively, we interpret this to mean that the condition of being \leq_m^P -hard for E forces a language to have a high level of organization, thereby forcing it to be unusually simple in some respects.

2 Preliminaries

Here we present the basic assumptions, notation, and terminology that we use throughout the paper. To begin with, we write \mathbf{N} for the set of natural numbers, \mathbf{Z} for the set of integers, and \mathbf{Z}^+ for set of positive integers.

We deal primarily with *strings*, *languages*, *functions*, and *classes*. Strings are finite sequences of characters over the alphabet $\{0, 1\}$; we write $\{0, 1\}^*$ for the set of all strings. Languages are sets of strings. Functions usually map $\{0, 1\}^*$ into $\{0, 1\}^*$. A class is either a set of languages or a set of functions.

If $x \in \{0, 1\}^*$ is a string, we write $|x|$ for the *length* of x . If $A \subseteq \{0, 1\}^*$ is a language, then we write A^c , $A_{\leq n}$, and $A_{=n}$ for $\{0, 1\}^* - A$, $A \cap \{0, 1\}^{\leq n}$, and $A \cap \{0, 1\}^n$ respectively. The sequence of strings over $\{0, 1\}$, $s_0 = \lambda$, $s_1 = 0$, $s_2 = 1$, $s_3 = 00$, ..., is referred to as the standard enumeration of $\{0, 1\}^*$.

We say that a property $\phi(n)$ of natural numbers holds *almost everywhere* (*a.e.*) if $\phi(n)$ is true for all but finitely many $n \in \mathbf{N}$. Similarly, $\phi(n)$ holds *infinitely often* (*i.o.*) if $\phi(n)$ is true for infinitely many $n \in \mathbf{N}$. We write $\llbracket \phi \rrbracket$ for the Boolean value of a condition ϕ . That is, $\llbracket \phi \rrbracket = 1$ if ϕ is true, 0 if ϕ is false.

We use the string pairing function $\langle x, y \rangle = bd(x)01y$, where $bd(x)$ is x with each bit doubled (e.g., $bd(1101) = 11110011$). Note that $|\langle x, y \rangle| = 2|x| + |y| + 2$ for all $x, y \in \{0, 1\}^*$. For each $g : \{0, 1\}^* \rightarrow \{0, 1\}^*$ and $k \in \mathbf{N}$, we also define the function $g_k : \{0, 1\}^* \rightarrow \{0, 1\}^*$ by $g_k(x) = g(\langle 0^k, x \rangle)$ for all $x \in \{0, 1\}^*$.

If A is a finite set, we denote its cardinality by $|A|$. A language D is *dense* if there exists some constant $\epsilon > 0$ such that $|D_{\leq n}| > 2^{n^\epsilon}$ a.e. A language S is *sparse* if there exists a polynomial p such that $|S_{\leq n}| \leq p(n)$ a.e.. A language S is *co-sparse* if S^c is sparse.

All *machines* here are deterministic Turing machines. The language accepted by a machine M is denoted by $L(M)$. The partial function computed by a machine M is denoted by $f_M : \{0, 1\}^* \rightarrow \{0, 1\}^*$. For a fixed machine M , the function $time_M(x)$ represents the number of steps that M uses on input x .

If $t(n)$ is a time bound, then we write

$$\text{DTIME}(t(n)) = \{L(M) \mid (\exists c)(\forall x) time_M(x) \leq c \cdot t(|x|) + c\}$$

for the set of languages computable in $O(t(n))$ time. Similarly, we write

$$\text{DTIMEF}(t(n)) = \{f_M \mid (\exists c)(\forall x) time_M(x) \leq c \cdot t(|x|) + c\}.$$

for the set of functions computable in $O(t(n))$ -time. We write P and PF for the set of languages and functions, respectively, that are computable in polynomial time. We are especially interested in the classes of languages computable in exponential time. Our notation for the exponential time classes differs slightly from that of [BDG88, BDG90]. We write

$$\begin{aligned} E &= \bigcup_{c=1}^{\infty} \text{DTIME}(2^{cn}) \text{ and} \\ E_2 &= \bigcup_{c=1}^{\infty} \text{DTIME}(2^{n^c}) \end{aligned}$$

for the classes of languages computable in $\text{DTIME}(2^{\text{linear}})$ and $\text{DTIME}(2^{\text{polynomial}})$, respectively. The other standard complexity classes that we use here, such as NP , PH , $PSPACE$, etc., are defined precisely as in [BDG88, BDG90].

If A and B are languages, then a *polynomial time, many-one reduction* (briefly \leq_m^P -reduction) of A to B is a function $f \in PF$ such that $A = f^{-1}(B) = \{x \mid f(x) \in B\}$. A \leq_m^P -reduction of A is a function $f \in PF$ that is a \leq_m^P -reduction of A to some language B . Note that f is a \leq_m^P -reduction of A if and only if f is \leq_m^P -reduction of A to $f(A) = \{f(x) \mid x \in A\}$. We say that A is *polynomial time, many-one reducible* (briefly, \leq_m^P -reducible) to B , and we write $A \leq_m^P B$, if there exists a \leq_m^P -reduction f of A to B . In this case, we also say that $A \leq_m^P B$ via f .

A language H is \leq_m^P -hard for a class \mathcal{C} of languages if $A \leq_m^P H$ for all $A \in \mathcal{C}$. A language C is \leq_m^P -complete for \mathcal{C} if $C \in \mathcal{C}$ and C is \leq_m^P -hard for \mathcal{C} . If $\mathcal{C} = NP$, this is the usual notion of NP-completeness [GJ79]. In this paper we are especially concerned with languages that are \leq_m^P -hard or \leq_m^P -complete for E or E_2 .

3 Resource-Bounded Measure

In this section we review some fundamentals of the resource-bounded measure formulated by Lutz in [Lut92a, Lut92c]. Here we restrict our discussion to the resource bounds that give measure structure to the complexity classes E and E_2 . For a more detailed and general development of resource-bounded measure, see [Lut92a, Lut92c].

Resource-bounded measure is a resource-bounded generalization of classical Lebesgue measure on the set of infinite binary sequences, $\{0, 1\}^\infty$. Since

we are primarily concerned with sets of languages, we note the one-to-one correspondence between the set of all infinite binary sequences, $\{0, 1\}^\infty$, and the set of all languages, $\mathcal{P}(\{0, 1\}^*)$. For each language $A \subseteq \{0, 1\}^*$, we associate A with its *characteristic sequence* $\chi_A \in \{0, 1\}^\infty$ and vice versa, where χ_A is defined as the infinite binary sequence satisfying $\chi_A[i] = \llbracket s_i \in A \rrbracket$ for all $i \in \mathbf{N}$. (Recall from §2, that s_0, s_1, s_2, \dots is the standard enumeration of $\{0, 1\}^*$.)

Partial specifications are the basis for *measurement* in resource-bounded measure. For $x \in \{0, 1\}^*$ and $A \subseteq \{0, 1\}^*$, we say that x is a *prefix*, or *partial specification*, of A if x is a prefix of χ_A , i.e., if there exists $y \in \{0, 1\}^\infty$ such that $\chi_A = xy$. In this case, we write $x \sqsubseteq A$. The set of all languages A for which x is a partial specification,

$$C_x = \{A \subseteq \{0, 1\}^* \mid x \sqsubseteq A\},$$

is the *cylinder specified by* the string $x \in \{0, 1\}^*$. We say that the *measure*, or *length* of the set C_x is $2^{-|x|}$. (Note that if $y \in \{0, 1\}^\infty$ is chosen probabilistically in the usual random experiment, then the probability that $y \in C_x$ is $2^{-|x|}$. Moreover, if we associate a real number in the unit interval with each sequence $x \in \{0, 1\}^\infty$, then the length of the set C_x is exactly $2^{-|x|}$.)

Resource-bounded measure is formulated in terms of uniform systems of density functions. A *density function* is a function $d : \{0, 1\}^* \rightarrow [0, \infty)$ satisfying

$$d(x) \geq \frac{d(x0) + d(x1)}{2}$$

for all $x \in \{0, 1\}^*$. The *global value* of a density function d is $d(\lambda)$. An n -dimensional *density system* (n -DS) is a function $d : \mathbf{N}^n \times \{0, 1\}^* \rightarrow [0, \infty)$ such that $d_{\vec{k}}$ is a density function for every $\vec{k} \in \mathbf{N}^n$.

The *set covered by* a density function d is

$$S[d] = \bigcup_{\substack{x \in \{0, 1\}^* \\ d(x) \geq 1}} C_x.$$

A density function d *covers* a set X of languages if $X \subseteq S[d]$. A *null cover* of a set X of languages is a 1-DS d such that, for all $k \in \mathbf{N}$, d_k covers X with global value $d_k(\lambda) \leq 2^{-k}$. It is easy to show [Lut92c] that a set X of languages has classical Lebesgue measure 0 (i.e., probability 0 in the coin-tossing random experiment) if and only if there exists a null cover of X .

The primary difference between classical Lebesgue measure and resource-bounded measure is the requirement that the density systems be uniformly computable in some resource bound.

To formalize the concept of computing density systems, resource-bounded measure approximates density systems by dyadic rational numbers. We let $\mathbf{D} = \{m2^{-n} | m, n \in \mathbf{N}\}$ be the set of *nonnegative dyadic rationals*. A *computation* of an n -DS d is a function $\hat{d} : \mathbf{N}^{n+1} \times \{0, 1\}^* \rightarrow \mathbf{D}$ such that

$$|\hat{d}_{\vec{k}, r}(x) - d_{\vec{k}}(x)| \leq 2^{-r} \quad (3.1)$$

for all $\vec{k} \in \mathbf{N}^n$, $r \in \mathbf{N}$, and $x \in \{0, 1\}^*$. Thus a computation of an n -DS d can be used to approximate d to any arbitrary precision.

In order to have a uniform criteria for computational complexity, we consider all functions of the form $f : X \rightarrow Y$, where each of the sets X , Y is \mathbf{N} , $\{0, 1\}^*$, \mathbf{D} , or some cartesian product of these sets, to really map $\{0, 1\}^*$ into $\{0, 1\}^*$. For example, a function $f : \mathbf{N}^2 \times \{0, 1\}^* \rightarrow \mathbf{N} \times \mathbf{D}$ is formally interpreted as a function $\tilde{f} : \{0, 1\}^* \rightarrow \{0, 1\}^*$. Under this interpretation, $f(i, j, w) = (k, q)$ means that $\tilde{f}(\langle 0^i, \langle 0^j, w \rangle \rangle) = \langle 0^k, \langle u, v \rangle \rangle$, where u and v are the binary representations of the integer and fractional parts of q , respectively. Moreover, we only care about the values of \tilde{f} for arguments of the form $\langle 0^i, \langle 0^j, w \rangle \rangle$, and we insist that these values have the form $\langle 0^k, \langle u, v \rangle \rangle$ for such arguments. We now have enough structure to consider the complexity of the *computations* we defined above.

Let Δ be a class of functions mapping $\{0, 1\}^*$ into $\{0, 1\}^*$. A Δ -*computation* of an n -DS d is a computation \hat{d} such that $\hat{d} \in \Delta$. An n -DS is Δ -*computable* if there exists a Δ -computation \hat{d} of d .

For our purposes, we assume that Δ is one of the following resource-bounded classes of functions.

$$\begin{aligned} p &= \bigcup_{c=1}^{\infty} \text{DTIMEF}(n^c) \\ p_2 &= \bigcup_{c=1}^{\infty} \text{DTIMEF}(n^{\log^{(c)} n}) \end{aligned}$$

Each of the above classes, Δ , naturally induces measure structure on a class of languages, $R(\Delta)$, where $R(p) = E$ and $R(p_2) = E_2$. We now have enough notation to formalize measure in complexity classes.

Definition. Let X be a set of languages and let X^c denote the complement of X .

- (1) A Δ -null cover of X is a null cover of X that is Δ -computable.
- (2) X has Δ -measure 0, and we write $\mu_\Delta(X) = 0$, if there exists a Δ -null cover of X .
- (3) X has Δ -measure 1, and we write $\mu_\Delta(X) = 1$, if $\mu_\Delta(X^c) = 0$.
- (4) X has measure 0 in $R(\Delta)$, and we write $\mu(X \mid R(\Delta)) = 0$, if $\mu_\Delta(X \cap R(\Delta)) = 0$.
- (5) X has measure 1 in $R(\Delta)$, and we write $\mu(X \mid R(\Delta)) = 1$, if $\mu(X^c \mid R(\Delta)) = 0$. In this case, we say that X contains *almost every* language in $R(\Delta)$.

It is shown in [Lut92a, Lut92c] that these definitions endow $R(\Delta)$ with internal measure-theoretic structure. Specifically, if \mathcal{I} is either the collection \mathcal{I}_Δ of all Δ -measure 0 sets or the collection $\mathcal{I}_{R(\Delta)}$ of all sets of measure 0 in $R(\Delta)$, then \mathcal{I} is a “ Δ -ideal,” i.e., is closed under subsets, finite unions, and “ Δ -unions” (countable unions that can be generated by functions in Δ). More importantly, it is shown that the ideal $\mathcal{I}_{R(\Delta)}$ is a *proper* ideal, i.e., that $R(\Delta)$ does *not* have measure 0 in $R(\Delta)$.

Many of our measure-theoretic proofs do not proceed directly from the above definitions. Instead we use the fact that the set of all Δ -measure 0 sets is a Δ -ideal. More precisely, we use the fact that a Δ -union of Δ -measure 0 sets has Δ -measure 0.

Definition.(Lutz [Lut92a]) Let $X, X_0, X_1, X_2, \dots \subseteq \{0, 1\}^\infty$. X is a Δ -union of the Δ -measure 0 sets X_0, X_1, X_2, \dots if $X = \bigcup_{j=0}^{\infty} X_j$ and there exists a Δ -computable 2-DS d such that each d_j is a null cover of X_j .

Lemma 3.1.(Lutz[Lut92a]). If X is a Δ -union of Δ -measure 0 sets, then X has Δ -measure 0.

4 Complexity Cores: Lower Bounds

Orponen and Schöning [OS86] have shown that every \leq_m^P -hard language for E has a dense polynomial complexity core. In this section we extend this result by proving that every weakly \leq_m^P -hard language for E has a dense exponential complexity core. We begin by explaining our terminology.

Given a machine M and an input $x \in \{0, 1\}^*$, we write $M(x) = 1$ if M accepts x , $M(x) = 0$ if M rejects x , and $M(x) = \perp$ in any other case (i.e., if M fails to halt or M halts without deciding x). If $M(x) \in \{0, 1\}$, we write $\text{time}_M(x)$ for the number of steps used in the computation of $M(x)$. If $M(x) = \perp$, we define $\text{time}_M(x) = \infty$. We partially order the set $\{0, 1, \perp\}$ by $\perp < 0$ and $\perp < 1$, with 0 and 1 incomparable. A machine M is *consistent* with a language $A \subseteq \{0, 1\}^*$ if $M(x) \leq \llbracket x \in A \rrbracket$ for all $x \in \{0, 1\}^*$.

Definition. Let $t : \mathbf{N} \rightarrow \mathbf{N}$ be a time bound and let $A, K \subseteq \{0, 1\}^*$. Then K is a $\text{DTIME}(t(n))$ -complexity core of A if, for every $c \in \mathbf{N}$ and every machine M that is consistent with A , the “fast set”

$$F = \{x \mid \text{time}_M(x) \leq c \cdot t(|x|) + c\}$$

satisfies $|F \cap K| < \infty$. (By our definition of $\text{time}_M(x)$, $M(x) \in \{0, 1\}$ for all $x \in F$. Thus F is the set of all strings that M “decides efficiently”.)

Note that every subset of a $\text{DTIME}(t(n))$ -complexity core of A is a $\text{DTIME}(t(n))$ -complexity core of A . Note also that, if $s(n) = O(t(n))$, then every $\text{DTIME}(t(n))$ -complexity core of A is a $\text{DTIME}(s(n))$ -complexity core of A .

Definition. Let $A, K \subseteq \{0, 1\}^*$.

1. K is a *polynomial complexity core* (or, briefly, a *P-complexity core*) of A if K is a $\text{DTIME}(n^k)$ -complexity core of A for all $k \in \mathbf{N}$.
2. K is an *exponential complexity core* of A if there is a real number $\epsilon > 0$ such that K is a $\text{DTIME}(2^{n^\epsilon})$ -complexity core of A .

Intuitively, a P-complexity core of A is a set of infeasible instances of A , while an exponential complexity core of A is a set of extremely hard instances of A .

Remark. The above definition quantifies over all machines consistent with A , while the standard definition of complexity cores (cf. [BDG90]) quantifies only over machines that *decide* A . This difference renders our definition stronger than the standard definition when A is not recursive. For example, consider *tally* languages (i.e., languages $A \subseteq \{0\}^*$). Under our definition, every $\text{DTIME}(n)$ -complexity core K of every tally language must satisfy $|K - \{0\}^*| < \infty$. However, under the standard definition, *every* set $K \subseteq \{0, 1\}^*$ is *vacuously* a complexity core for *every* nonrecursive language (tally

or otherwise). Thus by quantifying over all machines consistent with A , our definition makes the notion of complexity core meaningful for nonrecursive languages A . This enables one to eliminate the extraneous hypothesis that A is recursive from several results. In some cases (e.g., the fact that A is P-bi-immune if and only if $\{0, 1\}^*$ is a P-complexity core for A [BS85]), this improvement is of little interest. However in section 6 below, we show that *every* \leq_m^P -hard language H for E has unusually small complexity cores. This upper bound holds regardless of whether H is recursive.

It should also be noted that standard existence theorems on complexity cores (e.g., every language $A \notin P$ has an infinite P-complexity core [Lyn75]; every \leq_m^P -hard language for E has a dense P-complexity core [OS86]) remain true under our definition. Thus no harm is done by quantifying over all machines consistent with A .

Much of our work here uses languages that are “incompressible by many-one reductions,” an idea originally exploited by Meyer [Mey77]. The following definitions develop this notion.

Definition. The *collision set* of a function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ is

$$C_f = \{x \in \{0, 1\}^* \mid (\exists y < x)f(y) = f(x)\}.$$

Here, we are using the standard ordering $s_0 < s_1 < s_2 < \dots$ of $\{0, 1\}^*$.

Note that f is one-to-one if and only if $C_f = \emptyset$.

Definition. A function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ is *one-to-one almost everywhere* (or, briefly, *one-to-one a.e.*) if its collision set is C_f is finite.

Definition. Let $A, B \subseteq \{0, 1\}^*$ and let $t : \mathbf{N} \rightarrow \mathbf{N}$. A $\leq_m^{\text{DTIME}(t)}$ -reduction of A to B is a function $f \in \text{DTIME}(t)$ such that $A = f^{-1}(B)$, i.e., such that, for all $x \in \{0, 1\}^*$, $x \in A$ iff $f(x) \in B$. A $\leq_m^{\text{DTIME}(t)}$ -reduction of A is a function f that is a $\leq_m^{\text{DTIME}(t)}$ -reduction of A to $f(A)$.

It is easy to see that f is a $\leq_m^{\text{DTIME}(t)}$ -reduction of A if and only if there exists a language B such that f is a $\leq_m^{\text{DTIME}(t)}$ -reduction of A to B .

Definition. Let $t : \mathbf{N} \rightarrow \mathbf{N}$. A language $A \subseteq \{0, 1\}^*$ is *incompressible by $\leq_m^{\text{DTIME}(t)}$ -reductions* if every $\leq_m^{\text{DTIME}(t)}$ -reduction of A is one-to-one

a.e. A language $A \subseteq \{0,1\}^*$ is *incompressible by \leq_m^P -reductions* if it is incompressible by $\leq_m^{\text{DTIME}(q)}$ -reductions for all polynomials q .

Intuitively, if f is a $\leq_m^{\text{DTIME}(t)}$ -reduction of A to B and C_f is large, then f compresses many questions “ $x \in A$?” to fewer questions “ $f(x) \in B$?” If A is incompressible by \leq_m^P -reductions, then very little such compression can occur.

Our first observation, an obvious generalization of a result of Balcázar and Schöning [BS85] (see Corollary 4.2 below), relates incompressibility to complexity cores.

Lemma 4.1. If $t : \mathbf{N} \rightarrow \mathbf{N}$ is time constructible then every language that is incompressible by $\leq_m^{\text{DTIME}(t)}$ -reductions has $\{0,1\}^*$ as a $\text{DTIME}(t)$ -complexity core.

Proof. Let A be a language that does not have $\{0,1\}^*$ as a $\text{DTIME}(t)$ -complexity core. It suffices to prove that A is not incompressible by $\leq_m^{\text{DTIME}(t)}$ -reductions. This is clear if $A = \emptyset$ or $A = \{0,1\}^*$, so assume that $\emptyset \neq A \neq \{0,1\}^*$. Fix $u \in A$ and $v \in A^c$. Since $\{0,1\}^*$ is not a $\text{DTIME}(t)$ -complexity core of A , there exist $c \in \mathbf{N}$ and a machine M such that M is consistent with A and the fast set

$$F = \{x \mid \text{time}_M(x) \leq c \cdot t(|x|) + c\}$$

is infinite. Define a function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ by

$$f(x) = \begin{cases} u & \text{if } M(x) = 1 \text{ in } \leq c \cdot t(|x|) + c \text{ steps} \\ v & \text{if } M(x) = 0 \text{ in } \leq c \cdot t(|x|) + c \text{ steps} \\ x & \text{otherwise.} \end{cases}$$

Since t is time-constructible, $f \in \text{DTIME}(t)$. Since M is consistent with A , f is a $\leq_m^{\text{DTIME}(t)}$ -reduction of A to A . Since F is infinite, at least one of the sets $f^{-1}(\{u\})$, $f^{-1}(\{v\})$ is infinite, so the collision set C_f is infinite. Thus A is not incompressible by $\leq_m^{\text{DTIME}(t)}$ -reductions. \square

Corollary 4.2. Let $c \in \mathbf{N}$.

1. (Balcázar and Schöning [BS85]). Every language that is incompressible by \leq_m^P -reductions has $\{0,1\}^*$ as a P -complexity core.
2. Every language that is incompressible by $\leq_m^{\text{DTIME}(2^{cn})}$ -reductions has $\{0,1\}^*$ as a $\text{DTIME}(2^{cn})$ -complexity core.

3. Every language that is incompressible by $\leq_m^{\text{DTIME}(2^{n^c})}$ -reductions has $\{0, 1\}^*$ as a $\text{DTIME}(2^{n^c})$ -complexity core. \square

We now prove that, in E and E_2 , almost every language is incompressible by $\leq_m^{\text{DTIME}(t)}$ -reductions, for exponential time bounds t .

Theorem 4.3. Let $c \in \mathbf{Z}^+$ and define the sets

$$X = \{A \subseteq \{0, 1\}^* \mid A \text{ is incompressible by } \leq_m^{\text{DTIME}(2^{cn})}\text{-reductions}\},$$

$$Y = \{A \subseteq \{0, 1\}^* \mid A \text{ is incompressible by } \leq_m^{\text{DTIME}(2^{n^c})}\text{-reductions}\}.$$

Then $\mu_p(X) = \mu_{p_2}(Y) = 1$. Thus almost every language in E is incompressible by $\leq_m^{\text{DTIME}(2^{cn})}$ -reductions, and almost every language in E_2 is incompressible by $\leq_m^{\text{DTIME}(2^{n^c})}$ -reductions.

Proof. Let $c \in \mathbf{Z}^+$. We prove that $\mu_p(X) = 1$. The proof that $\mu_{p_2}(Y) = 1$ is analogous.

Let $f \in \text{DTIMEF}(2^{(c+1)n})$ be a function that is universal for $\text{DTIMEF}(2^{cn})$, in the sense that

$$\text{DTIMEF}(2^{cn}) = \{f_i \mid i \in \mathbf{N}\}.$$

For each $i \in \mathbf{N}$, define a set Z_i of languages as follows: If the collision set C_{f_i} is finite, then $Z_i = \emptyset$. Otherwise, if C_{f_i} is infinite, then Z_i is the set of all languages A such that f_i is a $\leq_m^{\text{DTIME}(2^{cn})}$ -reduction of A .

Define a function $d : \mathbf{N} \times \mathbf{N} \times \{0, 1\}^* \rightarrow [0, \infty)$ as follows: Let $i, k \in \mathbf{N}$ be arbitrary, let $w \in \{0, 1\}^*$, and let $b \in \{0, 1\}$.

$$(i) \quad d_{i,k}(\lambda) = 2^{-k}.$$

$$(ii) \quad \text{If } s_{|w|} \notin C_{f_i}, \text{ then } d_{i,k}(wb) = d_{i,k}(w).$$

$$(iii) \quad \text{If } s_{|w|} \in C_{f_i}, \text{ then fix the least } j \in \mathbf{N} \text{ such that } f_i(s_j) = f_i(s_{|w|}) \text{ and set}$$

$$d_{i,k}(wb) = 2 \cdot d_{i,k}(w) \cdot \llbracket b = w[j] \rrbracket.$$

It is clear that d is a 2-DS. Since $f \in \text{DTIMEF}(2^{(c+1)n})$ and the computation of $d_{i,k}(w)$ only uses values $f_i(u)$ for strings u with $|u| = O(\log |w|)$, it is also clear that $d \in \text{p}$, so d is a p-computable 2-DS.

We now show that $Z_i \subseteq S[d_{i,k}]$ for all $i, k \in \mathbf{N}$. If C_{f_i} is finite, then this is clear (because $Z_i = \emptyset$), so assume that C_{f_i} is infinite and let $A \in Z_i$. Let

w be a string consisting of the first l bits of the characteristic sequence of A , where s_{l-1} is the k^{th} element of C_{f_i} . This choice of l ensures that clause (iii) of the definition of d is invoked exactly k times in the recursive computation of $d_{i,k}(w)$. Since f_i is a $\leq_m^{\text{DTIME}(2^{cn})}$ -reduction of A (because $A \in Z_i$), we have $b = w[j]$ in each of these k invocations, so

$$d_{i,k}(w) = 2^k \cdot d_{i,k}(\lambda) = 1.$$

Thus $A \in C_w \subseteq S[d_{i,k}]$. This confirms that $Z_i \subseteq S[d_{i,k}]$ for all $i, k \in \mathbf{N}$. It follows easily that, for each $i \in \mathbf{N}$, d_i is a p-null cover of Z_i . This implies that

$$X^c = \bigcup_{k=0}^{\infty} Z_k$$

is a p-union of p-measure 0 sets, whence $\mu_p(X) = 1$ by Lemma 3.1. \square

Corollary 4.4. Almost every language in E and almost every language in E_2 is incompressible by \leq_m^P -reductions. \square

Corollary 4.5.(Meyer[Mey77]). There is language $A \in E$ that is incompressible by \leq_m^P -reductions. \square

Corollary 4.6. Let $c \in \mathbf{Z}^+$.

1. Almost every language in E has $\{0, 1\}^*$ as a $\text{DTIME}(2^{cn})$ -complexity core.
2. Almost every language in E_2 has $\{0, 1\}^*$ as a $\text{DTIME}(2^{n^c})$ -complexity core. \square

We now consider complexity cores of \leq_m^P -hard languages. Our starting point is the following two known facts.

Fact 4.7.(Orponen and Schöning [OS86]). Every language that is \leq_m^P -hard for E (equivalently, for E_2) has a dense P-complexity core.

Fact 4.8.(Orponen and Schöning [OS86]). If $P \neq NP$, then every language that is \leq_m^P -hard for NP has a nonsparse P-complexity core.

We first extend Fact 4.7. For this we need a definition. The *lower* \leq_m^P -span of a language $A \subseteq \{0, 1\}^*$ is

$$P_m(A) = \{B \subseteq \{0, 1\}^* \mid B \leq_m^P A\},$$

i.e., the set of all languages lying “at or below” A in the \leq_m^P -reducibility structure of the set of all languages. Recall that a language A is \leq_m^P -hard for a complexity class \mathcal{C} if $\mathcal{C} \subseteq P_m(A)$.

Definition. A language $A \subseteq \{0, 1\}^*$ is *weakly \leq_m^P -hard* for E (respectively, for E_2) if $\mu(P_m(A) \mid E) \neq 0$ (respectively, $\mu(P_m(A) \mid E_2) \neq 0$).

Thus a language A is weakly \leq_m^P -hard for E if a nonnegligible subset of the languages in E are \leq_m^P -reducible to A . The existence of languages that are weakly hard, but not hard, is an open question. (See section 7.) Also, although “ \leq_m^P -hard for E ” and “ \leq_m^P -hard for E_2 ” are equivalent, we do not know the relationship between “weakly \leq_m^P -hard for E ” and “weakly \leq_m^P -hard for E_2 .”

Recall that a language $D \subseteq \{0, 1\}^*$ is *dense* if there is a real number $\epsilon > 0$ such that $|D_{\leq n}| > 2^{n^\epsilon}$ a.e.

Theorem 4.9. Every language that is weakly \leq_m^P -hard for E or E_2 has a dense exponential complexity core.

Proof. We prove this for E . The proof for E_2 is identical.

Let H be a language that is weakly \leq_m^P -hard for E . Then $P_m(H)$ does not have measure 0 in E , so by Theorem 4.3, there is a language $A \in P_m(H)$ that is incompressible by $\leq_m^{\text{DTIME}(2^n)}$ -reductions. Let f be a \leq_m^P -reduction of A to H . Then f is also a $\leq_m^{\text{DTIME}(2^n)}$ -reduction of A , so the collision set C_f is finite. We will show that the language

$$K = f(\{0, 1\}^*)$$

is a dense $\text{DTIME}(2^{n^\epsilon})$ -complexity core of H .

Let q be a strictly increasing polynomial bound on the time required to compute the reduction f and let $\epsilon = \frac{1}{3 \cdot \deg(q)}$. For all sufficiently large n , $q(\lfloor n^{2^\epsilon} \rfloor) \leq n$, so

$$f(\{0, 1\}^{\lfloor n^{2^\epsilon} \rfloor}) \subseteq K_{\leq n},$$

so

$$\begin{aligned} |K_{\leq n}| &\geq |\{0, 1\}^{\lfloor n^{2^\epsilon} \rfloor}| - |C_f| \\ &= 2^{\lfloor n^{2^\epsilon} \rfloor} - |C_f| \\ &> 2^{n^\epsilon} \text{ a.e.} \end{aligned}$$

Thus K is dense.

To see that K is a $\text{DTIME}(2^{n^\epsilon})$ -complexity core of H , let $c \in \mathbb{N}$, let M be a machine that is consistent with H , and define the fast set

$$F = \{x \mid \text{time}_M(x) \leq c \cdot 2^{|x|^\epsilon} + c\}.$$

Let \hat{M} be a machine (designed in the obvious way) such that

$$\hat{M}(x) = M(f(x))$$

for all $x \in \{0, 1\}^*$. Since f reduces A to H and M is consistent with H , \hat{M} is consistent with A . By Corollary 4.2 (part 2), $\{0, 1\}^*$ is a $\text{DTIME}(2^n)$ -complexity core for A , so it follows that the fast set

$$\hat{F} = \{x \mid \text{time}_{\hat{M}}(x) \leq c \cdot 2^n + c\}$$

is finite. By our choice of ϵ , for all but finitely many y , $y \in F \cap K$ implies $y \in f(\hat{F})$. That is, the set $(F \cap K) - f(\hat{F})$ is finite. Since $|\hat{F}| < \infty$, it follows that $|F \cap K| < \infty$. Thus K is a dense $\text{DTIME}(2^{n^\epsilon})$ -complexity core of H . \square

Lutz has proposed the investigation of the consequences of the strong hypotheses $\mu(\text{NP} \mid E) \neq 0$ and $\mu(\text{NP} \mid E_2) \neq 0$. In this regard, we have the following.

Corollary 4.10. If $\mu(\text{NP} \mid E) \neq 0$ or $\mu(\text{NP} \mid E_2) \neq 0$, then every \leq_m^P -hard language for NP has a dense exponential complexity core. \square

Thus, for example, if NP is not small, then there is a dense set K of Boolean formulas in conjunctive normal form such that every machine that is consistent with SAT performs exponentially badly (either by running for more than $2^{|x|^\epsilon}$ steps or by failing to decide) on all but finitely many inputs $x \in K$.

Note that Theorem 4.9 extends Fact 4.7 and that Corollary 4.10 has a stronger hypothesis and stronger conclusion than Fact 4.8. Note also that Corollary 4.10 holds with NP replaced by PH, PP, PSPACE, or any class whatsoever.

The following result shows that the density bounds of Theorem 4.9 and Corollary 4.10 are tight.

Theorem 4.11. For every $\epsilon > 0$, each of the classes NP, E, and E_2 has a \leq_m^P -complete language, every P-complexity core K of which satisfies $|K_{\leq n}| < 2^{n^\epsilon}$ a.e.

Proof. Let $\epsilon > 0$, let \mathcal{C} be any one of the classes NP, E, E_2 , and let A be a language that is \leq_m^P -complete for \mathcal{C} . Let $k = \lceil \frac{2}{\epsilon} \rceil$ and define the language

$$B = \{x10^{|x|^k} \mid x \in A\}.$$

It is clear that B is \leq_m^P -complete for \mathcal{C} . Let K be a P-complexity core of B . To prove the theorem, it suffices to show that $|K_{\leq n}| < 2^{n^\epsilon}$ a.e.

Let

$$D = \{x10^{|x|^k} \mid x \in \{0, 1\}^*\}$$

and let M be a machine (designed in the obvious way) such that, for all $y \in \{0, 1\}^*$,

$$M(y) = \begin{cases} 0 & \text{if } y \notin D \\ \perp & \text{if } y \in D. \end{cases}$$

Then M is consistent with B (because $B \subseteq D$) and there is a polynomial t such that

$$D^c = \{x \mid \text{time}_M(x) \leq t(|x|)\}.$$

Since K is a P-complexity core of B , it follows that $|K \cap D^c| = c < \infty$ for some $c \in \mathbb{N}$. For all sufficiently large n , then,

$$\begin{aligned} |K_{\leq n}| &\leq c + |D_{\leq n}| \\ &= c + \sum_{m=0}^n |D_{=m}| \\ &\leq c + \sum_{m=0}^n 2^{m^{\frac{1}{k}}} \\ &\leq c + (n+1)2^{n^{\frac{1}{k}}} \\ &\leq c + (n+1)2^{n^{\frac{\epsilon}{2}}} \\ &< 2^{n^\epsilon}. \end{aligned}$$

This completes the proof. □

5 Measure of Degrees

In this section we prove that all \leq_m^P -degrees have measure 0 in the complexity classes E and E₂. This fact and more will follow from the Small Span Theorem, which we prove first.

Recall that the *lower* \leq_m^P -span of a language $A \subseteq \{0, 1\}^*$ is

$$P_m(A) = \{B \subseteq \{0, 1\}^* \mid B \leq_m^P A\}.$$

Similarly, define the *upper* \leq_m^P -span of A to be

$$P_m^{-1}(A) = \{B \subseteq \{0, 1\}^* \mid A \leq_m^P B\}.$$

The \leq_m^P -degree of A is then

$$\deg_m^P(A) = P_m(A) \cap P_m^{-1}(A),$$

the intersection of the upper and lower spans.

Intuitively, in the \leq_m^P -reducibility structure of the set of all languages, we think of $P_m(A)$ as lying “below” A , while $P_m^{-1}(A)$ lies “above” A . (See Figure 1.) We will be especially concerned with the size, i.e., the resource-bounded measure, of the upper and lower spans of various languages. If neither of those spans is small (i.e., neither has resource-bounded measure 0), then we have the configuration depicted schematically in Figure 1. On the other hand, if one or both of these spans is small, then we have one of the “small-span” configurations depicted schematically in Figure 2. The main result of this section is that, if A is in E or E₂, *then at least one of the sets* $P_m(A)$, $P_m^{-1}(A)$ *is small*. That is, only small-span configurations can occur in E or E₂:

Theorem 5.1.(Small Span Theorem)

1. For every $A \in E$,

$$\mu(P_m(A) \mid E) = 0$$

or

$$\mu_p(P_m^{-1}(A)) = \mu(P_m^{-1}(A) \mid E) = 0.$$

2. For every $A \in E_2$,

$$\mu(P_m(A) \mid E_2) = 0$$

or

$$\mu_{p_2}(P_m^{-1}(A)) = \mu(P_m^{-1}(A) \mid E_2) = 0.$$

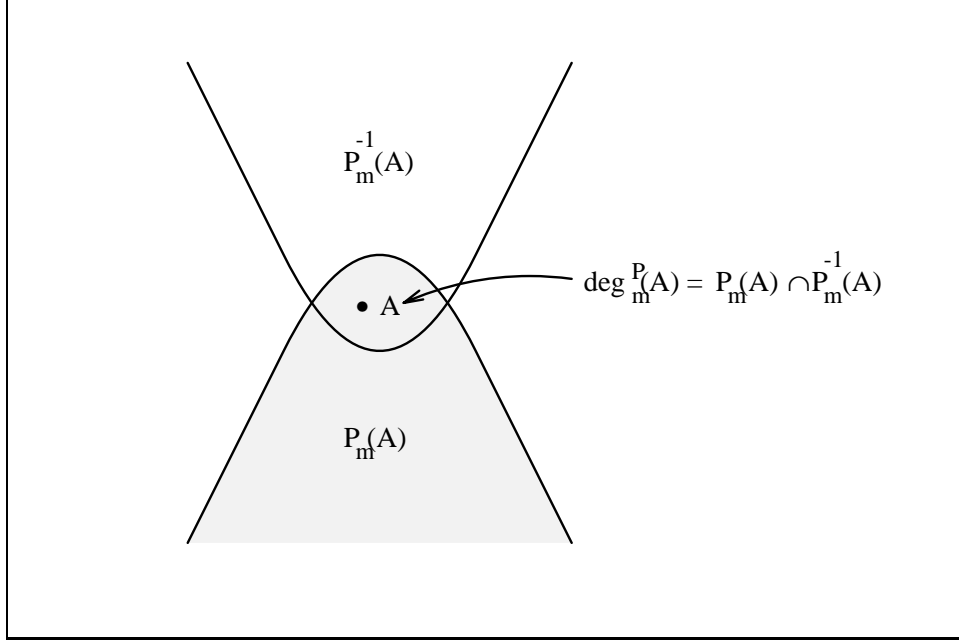


Figure 1: The upper span, lower span (shaded), and degree of A

We first use the following lemma to prove Theorem 5.1 We then prove the lemma.

Lemma 5.2. Let A be a language that is incompressible by \leq_m^P -reductions.

1. If $A \in E$, then $\mu_P(P_m^{-1}(A)) = \mu(P_m^{-1}(A)|E) = 0$.
2. If $A \in E_2$, then $\mu_{P_2}(P_m^{-1}(A)) = \mu(P_m^{-1}(A)|E_2) = 0$.

Proof of Theorem 5.1.

To prove 1, let $A \in E$ and let X be the set of all languages that are incompressible by \leq_m^P -reductions. We have two cases.

Case I. If $P_m(A) \cap E \cap X = \emptyset$, then Corollary 4.4 tells us that $\mu(P_m(A) | E) = 0$.

Case II. If $P_m(A) \cap E \cap X \neq \emptyset$, then fix a language $B \in P_m(A) \cap E \cap X$. Since $B \in E \cap X$, Lemma 5.2 tells us that

$$\mu_P(P_m^{-1}(B)) = \mu(P_m^{-1}(B) | E) = 0.$$

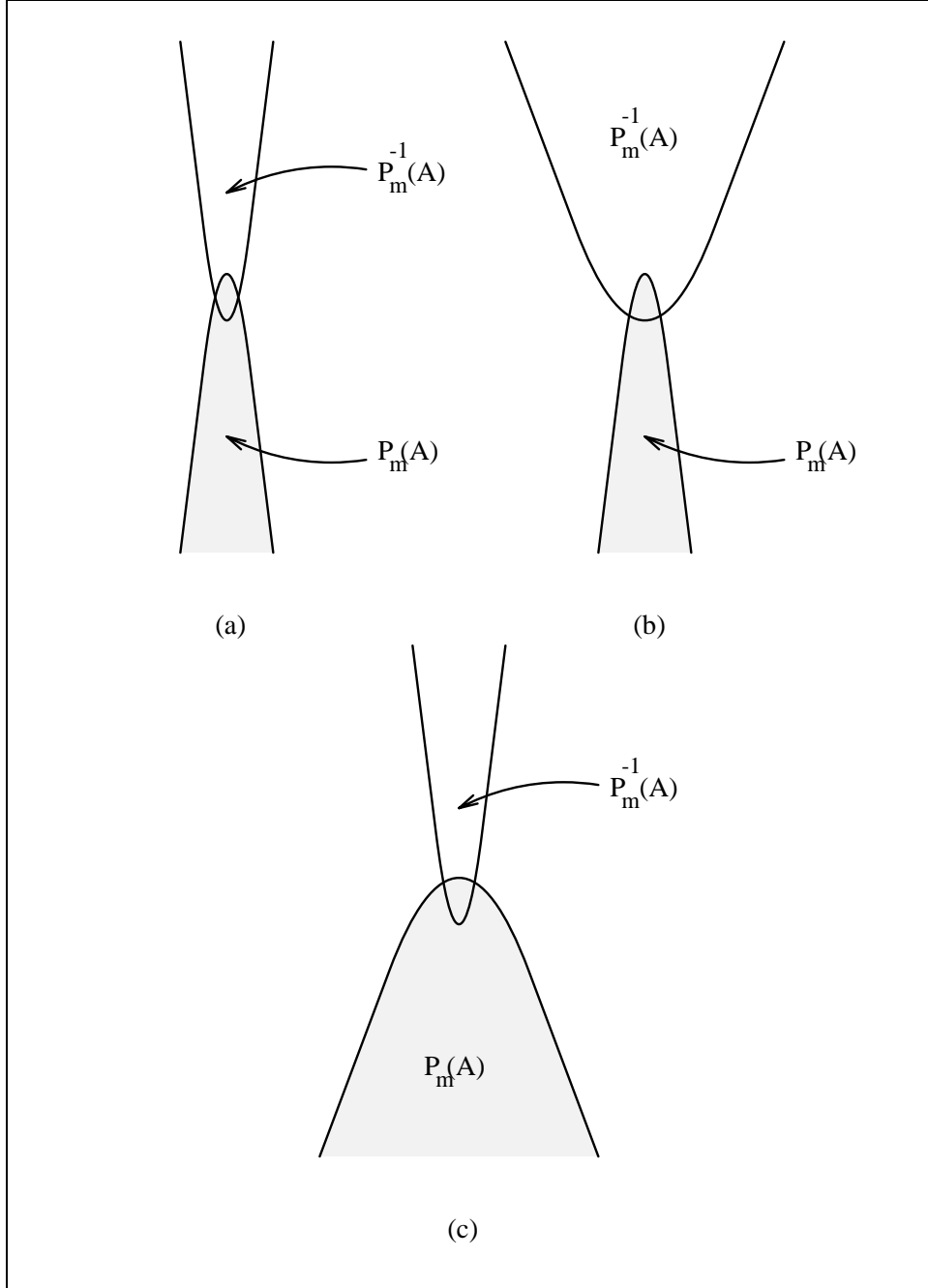


Figure 2: Small-span configurations (narrow regions depict measure 0 spans)

Since $P_m^{-1}(A) \subseteq P_m^{-1}(B)$, it follows that

$$\mu_p(P_m^{-1}(A)) = \mu(P_m^{-1}(A) \mid E) = 0.$$

This proves 1. The proof of 2 is identical. \square

Proof of Lemma 5.2.

To prove 1, let $A \in E$ be incompressible by \leq_m^P -reductions. Let $f \in \text{DTIMEF}(2^n)$ be a function that is universal for PF, in the sense that

$$\text{PF} = \{f_i \mid i \in \mathbf{N}\}.$$

For each $i \in \mathbf{N}$, define the set Z_i of languages as follows: If the collision set C_{f_i} is infinite, then $Z_i = \emptyset$. Otherwise, if C_{f_i} is finite, then

$$Z_i = \{B \subseteq \{0,1\}^* \mid A \leq_m^P B \text{ via } f_i\}.$$

Note that

$$P_m^{-1}(A) = \bigcup_{i=0}^{\infty} Z_i$$

because A is incompressible by \leq_m^P -reductions.

Define a function $d : \mathbf{N} \times \mathbf{N} \times \{0,1\}^* \rightarrow [0, \infty)$ as follows: Let $i, k \in \mathbf{N}$ be arbitrary, let $w \in \{0,1\}^*$, and let $b \in \{0,1\}$.

(i) $d_{i,k}(\lambda) = 2^{-k}$.

(ii) If there is no $j \leq 2|w|$ such that $f_i(s_j) = s_{|w|}$, then $d_{i,k}(wb) = d_{i,k}(w)$.

(iii) If there exists $j \leq 2|w|$ such that $f_i(s_j) = s_{|w|}$, then fix the least such j and set

$$d_{i,k}(wb) = 2 \cdot d_{i,k}(w) \cdot \llbracket b = \llbracket s_j \in A \rrbracket \rrbracket.$$

It is clear that d is a 2-DS. Also, since $f \in \text{DTIMEF}(2^n)$ and $A \in E$, it is easy to see that $d \in p$, whence d is a p-computable 2-DS.

We now show that $Z_i \subseteq S[d_{i,k}]$ for all $i, k \in \mathbf{N}$. If C_{f_i} is infinite, then this is clear (because $Z_i = \emptyset$), so assume that $|C_{f_i}| = c < \infty$ and let $B \in Z_i$, i.e., $A \leq_m^P B$ via f_i . Let v be the string consisting of the first l bits of the characteristic sequence of B , where l is large enough that

$$f_i(\{s_0, \dots, s_{2k+4c-1}\}) \subseteq \{s_0, \dots, s_{l-1}\}.$$

Consider the computation of $d_{i,k}(v)$ by clauses (i), (ii), and (iii) above. Since $A \leq_m^P B$ via f_i , clause (iii) does not cause $d_{i,k}(w)$ to be 0 for any prefix w of v . Let

$$S = \{s_n \mid 0 \leq n < 2k + 4c \text{ and } f_i(s_n) \notin \{s_0, \dots, s_{\lceil \frac{n}{2} \rceil - 1}\}\}$$

and

$$T = f_i(S).$$

Then clause (iii) doubles the density whenever $s_{|w|} \in T$, so

$$d_{i,k}(v) \geq 2^{|T|} d_{i,k}(\lambda) = 2^{|T|-k} \geq 2^{|S|-k-c}.$$

Also, if

$$S' = \{s_n \mid 0 \leq n < 2k + 4c \text{ and } f_i(s_n) \notin \{s_0, \dots, s_{k+2c-1}\}\},$$

then $S \subseteq S'$ and

$$|S'| \geq (2k + 4c) - (k + 2c) - c = k + c.$$

Putting this all together, we have

$$d_{i,k}(v) \geq 2^{|S|-k-c} \geq 2^{|S'|-k-c} \geq 1,$$

whence $B \in C_v \subseteq S[d_{i,k}]$. This shows that $Z_i \subseteq S[d_{i,k}]$ for all $i, k \in \mathbf{N}$.

Since d is p-computable and $d_{i,k}(\lambda) = 2^{-k}$ for all $i, k \in \mathbf{N}$, it follows that, for all $i \in \mathbf{N}$, d_i is p-null cover of Z_i . This implies that $P_m^{-1}(A)$ is a p-union of the p-measure 0 sets Z_i . It follows by Lemma 3.1 that $\mu_p(P_m^{-1}(A)) = \mu(P_m^{-1}(A) \mid E) = 0$. This completes the proof of 1.

The proof of 2 is identical. One need only note that, if $A \in E_2$, then $d \in p_2$. \square

Remark. Ambos-Spies [Amb86] has shown that $P_m(A)$ has Lebesgue measure 0 whenever $A \notin P$. Lemma 5.2 obtains a stronger conclusion (resource-bounded measure 0) from a stronger hypothesis on A .

It is now straightforward to derive consequences of these results for the structure of E and E_2 . We first note that \leq_m^P -hard languages for E are extremely rare:

Theorem 5.3. Let \mathcal{H}_E be the set of all languages that are \leq_m^P -hard for E . Then $\mu_P(\mathcal{H}_E) = 0$.

Proof. Let A be as in Corollary 4.5. Then $\mathcal{H}_E \subseteq P_m^{-1}(A)$, so Lemma 5.2 tells us that

$$\mu_P(\mathcal{H}_E) = \mu_P(P_m^{-1}(A)) = 0.$$

□

Theorem 5.3 immediately yields an alternate proof of the following result.

Corollary 5.4. (Mayordomo[May92]). Let $\mathcal{C}_E, \mathcal{C}_{E_2}$ be the sets of languages that are \leq_m^P -complete for E, E_2 , respectively. Then $\mu(\mathcal{C}_E|E) = \mu(\mathcal{C}_{E_2}|E_2) = 0$. □

(Mayordomo's proof of Corollary 5.4 used Berman's result [Ber76], that no \leq_m^P -complete language for E is P -immune.)

As it turns out, Corollary 5.4 is only a special case of the following general result. All \leq_m^P -degrees have measure 0 in E and in E_2 :

Theorem 5.5. For all $A \subseteq \{0, 1\}^*$,

$$\mu(\deg_m^P(A) \mid E) = \mu(\deg_m^P(A) \mid E_2) = 0.$$

Proof. Let $A \subseteq \{0, 1\}^*$. We prove that $\mu(\deg_m^P(A) \mid E) = 0$. The proof that $\mu(\deg_m^P(A) \mid E_2) = 0$ is identical (in fact easier, because E_2 is closed under \leq_m^P).

If $\deg_m^P(A) \cap E = \emptyset$, then $\mu(\deg_m^P(A) \mid E) = 0$ holds trivially, so assume that $\deg_m^P(A) \cap E \neq \emptyset$. Fix $B \in \deg_m^P(A) \cap E$. Then, by Theorem 5.1,

$$\mu(\deg_m^P(B) \mid E) = \mu(P_m(B) \mid E) = 0$$

or

$$\mu(\deg_m^P(B) \mid E) = \mu(P_m^{-1}(B) \mid E) = 0.$$

Since $\deg_m^P(A) = \deg_m^P(B)$, it follows that $\mu(\deg_m^P(A) \mid E) = 0$. □

We now have the following two corollaries for NP.

Corollary 5.6. Let \mathcal{H}_{NP} be the set of languages that are \leq_m^P -hard for NP.

1. If $\mu(NP \mid E) \neq 0$, then $\mu(\mathcal{H}_{NP} \mid E) = 0$.
2. If $\mu(NP \mid E_2) \neq 0$, then $\mu(\mathcal{H}_{NP} \mid E_2) = 0$.

Proof. This follows immediately from Theorem 5.1, with $A = \text{SAT}$. \square

Corollary 5.7. Let \mathcal{C}_{NP} be the set of languages that are \leq_m^{P} -complete for NP. Then $\mu(\mathcal{C}_{\text{NP}} \mid E) = \mu(\mathcal{C}_{\text{NP}} \mid E_2) = 0$.

Proof. Since $\mathcal{C}_{\text{NP}} = \deg_m^{\text{P}}(\text{SAT})$, this follows immediately from Theorem 5.5. \square

It is interesting to note that Corollary 5.7, unlike Corollary 5.6, is an absolute result, requiring no unproven hypothesis. The price we pay for this is that we do not know *why* it holds! For example, the Small Span Theorem tells us that $\mathcal{C}_{\text{NP}} = \mathcal{H}_{\text{NP}} \cap \text{NP}$ has measure 0 in E because $\mu(\mathcal{H}_{\text{NP}} \mid E) = 0$ or $\mu(\text{NP} \mid E) = 0$, but it does *not* tell us which of these two very different situations occurs.

Note that Corollaries 5.6 and 5.7 also hold with NP replaced by *any other class whatsoever*.

We conclude this section by noting two respects in which the Small Span Theorem cannot be improved. First, the hypotheses $A \in E$ and $A \in E_2$ are essential for parts 1 and 2, respectively. For example, if A is p-random [Lut92b], then $\mu_p(\{A\}) \neq 0$, so none of $\deg_m^{\text{P}}(A)$, $\text{P}_m(A)$, $\text{P}_m^{-1}(A)$ can have p-measure 0.

The second respect in which the Small Span Theorem cannot be improved involves the variety of small-span configurations: In both E and E_2 , all the small-span configurations depicted in Figure 2 (a, b, c) do in fact occur. We give examples for E .

- (a) It is well known [Mey77] that there is a language $A \in E$ that is both sparse and incompressible by \leq_m^{P} -reductions. Fix such a language A . By Lemma 5.2, $\mu_p(\text{P}_m^{-1}(A)) = 0$. Also, since A is sparse, the main result of [LM92] implies that $\mu_p(\text{P}_m(A)) = 0$.
- (b) If $A \in \text{P} - \{\emptyset, \{0, 1\}^*\}$, then $\mu(\text{P}_m(A) \mid E) = \mu_p(\text{P}_m(A)) = 0$, but $\mu_p(\text{P}_m^{-1}(A)) \neq 0$ and $\mu(\text{P}_m^{-1}(A) \mid E) \neq 0$.
- (c) If A is \leq_m^{P} -complete for E , then $\mu(\text{P}_m^{-1}(A) \mid E) = \mu_p(\text{P}_m^{-1}(A)) = 0$ by Theorem 5.3, but $\mu(\text{P}_m(A) \mid E) = \mu(E \mid E) \neq 0$.

Similar examples can be given for E_2 .

6 Complexity Cores: Upper Bounds

In this section we give an explicit *upper* bound on the sizes of complexity cores of languages that are \leq_m^P -hard for E. This will imply that \leq_m^P -complete languages for E have *unusually small* complexity cores, for languages in E.

Theorem 6.1. For every \leq_m^P -hard language H for E, there exists $B, D \in \text{DTIME}(2^{4n})$ such that D is dense and $B = H \cap D$.

Proof. By Corollary 4.5, there is a language in E that is incompressible by \leq_m^P -reductions. In fact, Meyer's construction shows that there is a language $A \in \text{DTIME}(5^n)$ that is incompressible by \leq_m^P -reductions. For the sake of completeness, we review Meyer's construction at the end of this proof. First, however, we use this A to prove Theorem 6.1.

The following simple notation will be useful. The *nonreduced image* of a language $S \subseteq \{0, 1\}^*$ under a function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ is

$$f^{\geq}(S) = \{f(x) \mid x \in S \text{ and } |f(x)| \geq |x|\}.$$

Note that

$$f^{\geq}(f^{-1}(S)) = S \cap f^{\geq}(\{0, 1\}^*)$$

for all f and S .

Let H be \leq_m^P -hard for E. Then there is a \leq_m^P -reduction f of A to H . Let $B = f^{\geq}(A)$, $D = f^{\geq}(\{0, 1\}^*)$. Since $A \in \text{DTIME}(5^n)$ and $f \in PF$, it is clear that $B, D \in \text{DTIME}(10^n) \subseteq \text{DTIME}(2^{4n})$.

Fix a polynomial q and a real number $\epsilon > 0$ such that $|f(x)| \leq q(|x|)$ for all $x \in \{0, 1\}^*$ and $q(n^{2\epsilon}) < n$ a.e. Let $W = \{x \mid |f(x)| < |x|\}$. Then, for all sufficiently large $n \in \mathbf{N}$, writing $m = \lfloor n^{2\epsilon} \rfloor$, we have

$$\begin{aligned} f(\{0, 1\}^{\leq m}) - \{0, 1\}^{< m} &\subseteq f(\{0, 1\}^{\leq m}) - f(W_{\leq m}) \\ &\subseteq f^{\geq}(\{0, 1\}^{\leq m}) \\ &\subseteq D_{\leq q(m)} \\ &\subseteq D_{\leq n}, \end{aligned}$$

whence

$$\begin{aligned} |D_{\leq n}| &\geq |f(\{0, 1\}^{\leq m})| - |\{0, 1\}^{< m}| \\ &\geq |\{0, 1\}^{\leq m}| - |C_f| - |\{0, 1\}^{< m}| \\ &= 2^m - |C_f|. \end{aligned}$$

```

begin
  input  $x$  ;
   $R := \emptyset$ ;  $S := \emptyset$ ;
  for  $n := 0$  to  $|x|$  do
    begin
       $R := R \cup \{n\}$ ;
      if there exists  $(k, y, z) \in R \times \{0, 1\}^n \times \{0, 1\}^{\leq n}$ 
        such that  $z < y$  and  $g_k(y) = g_k(z)$  then
        begin
          find the first such  $(k, y, z)$ ;
          if  $z \notin S$  then  $S := S \cup \{y\}$ ;
           $R := R - \{k\}$ 
        end
      end
    end;
    if  $x \in S$  then accept else reject
  end  $M$ .

```

Figure 3: Meyer's construction (for proof of Theorem 6.1).

Since $|C_f| < \infty$, it follows that $|D_{\leq n}| > 2^{n^\epsilon}$ for all sufficiently large n . Thus D is dense.

Finally, note that $B = f^{\geq}(A) = f^{\geq}(f^{-1}(H)) = H \cap f^{\geq}(\{0, 1\}^*) = H \cap D$. This completes the proof of Theorem 6.1.

We now describe Meyer's construction of the language A . It is well-known that there is a function $g \in \text{DTIMEF}(n^{\log n})$ that is universal for PF in the sense that

$$\text{PF} = \{g_k \mid k \in \mathbf{N}\}.$$

(Recall that g_k is defined by $g_k(x) = g(\langle 0^k, x \rangle)$ for all $x \in \{0, 1\}^*$.) Fix such a function g . Let $A = L(M)$, where M is a machine that implements the algorithm in Figure 3. It is clear by inspection that $A \in \text{DTIME}(5^n)$. To see that A is incompressible by \leq_m^P -reductions, suppose that $f \in \text{PF}$ and $|C_f| = \infty$. It suffices to show that f is not a \leq_m^P -reduction of A . Fix $k \in \mathbf{N}$ such that $f = g_k$. Then there is some $n \in \mathbf{N}$ such that, on input $x = 0^n$, M finds a triple (k, y, z) on cycle n of the for-loop. We then have $f(y) = g_k(y) = g_k(z) = f(z)$ and $y \in A \iff z \notin A$, so $f^{-1}(f(A)) \neq A$, so f is not a \leq_m^P -reduction of A . \square

We now use Theorem 6.1 to prove our upper bound on the size of complexity cores for hard languages.

Theorem 6.2. Every $\text{DTIME}(2^{4n})$ -complexity core of every \leq_m^P -hard language for E has a dense complement.

Proof. Let H be \leq_m^P -hard for E and let K be a $\text{DTIME}(2^{4n})$ -complexity core of H . Choose B, D for H as in Theorem 6.1. Fix machines M_B and M_D that decide B and D , respectively, with $\text{time}_{M_B}(x) = O(2^{4|x|})$ and $\text{time}_{M_D}(x) = O(2^{4|x|})$. Let M be a machine that implements the following algorithm.

```

begin
  input  $x$ ;
  if  $M_D(x)$  accepts
    then simulate  $M_B(x)$ 
    else run forever
end  $M$ .

```

Then $x \in D \Rightarrow M(x) = \llbracket x \in B \rrbracket = \llbracket x \in H \cap D \rrbracket = \llbracket x \in H \rrbracket$ and $x \notin D \Rightarrow M(x) = \perp \leq \llbracket x \in H \rrbracket$, so M is consistent with H . Also, there is a constant $c \in \mathbf{N}$ such that for all $x \in D$,

$$\text{time}_M(x) \leq c \cdot 2^{4n} + c.$$

Since K is a $\text{DTIME}(2^{4n})$ -complexity core of H , it follows that $K \cap D$ is finite. But D is dense, so this implies that $D - K$ is dense, whence K^c is dense. \square

Note that Theorem 5.3 follows from Corollary 4.6 and Theorem 6.2, but that Theorem 6.2 tells us more.

Finally, we note that the upper bound given by Theorem 6.2 is tight.

Theorem 6.3. Let $c \in \mathbf{N}$ and $0 < \epsilon \in \mathbf{R}$.

1. E has a \leq_m^P -complete language with a $\text{DTIME}(2^{cn})$ -complexity core K that satisfies $|K_{\leq n}| > 2^{n+1} - 2^{n^\epsilon}$ a.e.
2. E_2 has a \leq_m^P -complete language with a $\text{DTIME}(2^{n^c})$ -complexity core K that satisfies $|K_{\leq n}| > 2^{n+1} - 2^{n^\epsilon}$ a.e.

Proof. We prove the result for E . The proof for E_2 is similar.

Let A be a language that is \leq_m^P -complete for E and let $k = \lceil \frac{2}{\epsilon} \rceil$. By Corollary 4.6, fix a language $B \in E$ that has $\{0, 1\}^*$ as a $\text{DTIME}(2^{cn})$ -complexity core. Let

$$D = \{x10^{|x|^k} \mid x \in \{0, 1\}^*\}$$

and define the languages

$$C = (B - D) \cup \{x10^{|x|^k} \mid x \in A\}$$

and

$$K = D^c.$$

It is clear that C is \leq_m^P -complete for E . Also, for all sufficiently large n ,

$$\begin{aligned} |D_{\leq n}| &= \sum_{m=0}^n |D_{=m}| \\ &\leq \sum_{m=0}^n 2^{m^{\frac{1}{k}}} \\ &\leq (n+1)2^{n^{\frac{1}{k}}} \\ &\leq (n+1)2^{n^{\frac{\epsilon}{2}}} \\ &< 2^{n^\epsilon} - 1, \end{aligned}$$

so

$$|K_{\leq n}| = 2^{n+1} - 1 - |D_{\leq n}| > 2^{n+1} - 2^{n^\epsilon} \text{ a.e.}$$

We complete the proof by showing that K is a $\text{DTIME}(2^{cn})$ -complexity core for C . For this, let $s \in \mathbf{N}$, let M be a machine that is consistent with C , and define the fast set

$$F = \{x \mid \text{time}_M(x) \leq a \cdot 2^{c|x|} + a\}.$$

It suffices to prove that $|K \cap F| < \infty$.

Let \hat{M} be a machine (designed in the obvious way) such that, for all $y \in \{0, 1\}^*$,

$$\hat{M}(y) = \begin{cases} M(y) & \text{if } y \notin D \\ \perp & \text{if } y \in D. \end{cases}$$

Then \hat{M} is consistent with B (because $B - D = C - D$ and M is consistent with C) and $\{0, 1\}^*$ is a $\text{DTIME}(2^{cn})$ -complexity core for B , so the fast set

$$\hat{F} = \{x \mid \text{time}_{\hat{M}}(x) \leq (a+1)2^{c|x|} + a\}$$

is finite. Since $K \cap F = F - D$ and $(F - D) - \hat{F}$ is finite, it follows that $|K \cap F| < \infty$, completing the proof. \square

7 Conclusion

In this paper we have investigated measure-theoretic aspects of the \leq_m^P -reducibility structure of the exponential time complexity classes E and E_2 . Among other things, we have proven the following. (For simplicity we only consider the class E.)

- (i) Every weakly \leq_m^P -hard for E has a dense exponential complexity core (Theorem 4.9).
- (ii) For every language $A \in E$, at least one of the spans $P_m(A)$, $P_m^{-1}(A)$ has resource-bounded measure 0 (Theorem 5.1, the Small Span Theorem). Thus the \leq_m^P -hard languages for E form a p-measure 0 set (Theorem 5.3), *every* \leq_m^P -degree has measure 0 in E (Theorem 5.5), and the \leq_m^P -complete languages for NP form a set of measure 0 in E (Corollary 5.7).
- (iii) Every $\text{DTIME}(2^{4n})$ -complexity core of every \leq_m^P -hard language for E has a dense complement (Theorem 6.2). Since almost every language in E has $\{0, 1\}^*$ as a $\text{DTIME}(2^{4n})$ -complexity core (Corollary 4.6), this says that, in E, the \leq_m^P -complete languages are *unusually simple*, in the sense that they have *unusually small* complexity cores.

Item (i) above highlights the importance of resolving the following open problem.

Conjecture.(Lutz). There exist languages that are weakly \leq_m^P -hard, but not \leq_m^P -hard, for E.

In fact, Lutz has conjectured that SAT has this property, but this may be very hard to prove, since it implies that $P \neq NP$. It may be much easier to prove the Conjecture by direct diagonalization.

It is reasonable to conjecture that most of our results holds with \leq_m^P replaced by \leq_T^P . New techniques may be required to investigate this, since most of our proofs make essential use of languages that are incompressible by \leq_m^P -reductions. Is there an analogous, useful notion of incompressibility by \leq_T^P -reductions?

References

- [Amb86] K. Ambos-Spies. Randomness, relativizations, and polynomial reducibilities. In *Proceedings of the First Structure in Complexity Theory Conference*, pages 23–34, 1986.
- [BD87] R. Book and D.-Z. Du. The existence and density of generalized complexity cores. *Journal of the ACM*, 34:718–730, 1987.
- [BDG88] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I*. Springer-Verlag, 1988.
- [BDG90] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity II*. Springer-Verlag, 1990.
- [BDR88] R. Book, D.-Z. Du, and D. Russo. On polynomial and generalized complexity cores. In *Proceedings of the Third Structure in Complexity Theory Conference*, pages 236–250, 1988.
- [Ber76] L. Berman. On the structure of complete sets: Almost everywhere complexity and infinitely often speedup. In *Proceedings of the Seventeenth Annual Conference on Foundations of Computer Science*, pages 76–80, 1976.
- [BH77] L. Berman and J. Hartmanis. On isomorphism and density of NP and other complete sets. *SIAM Journal on Computing*, 6:305–322, 1977.
- [BS85] J. L. Balcázar and U. Schöning. Bi-immune sets for complexity classes. *Mathematical Systems Theory*, 18:1–10, 1985.
- [DB89] D.-Z. Du and R. Book. On inefficient special cases of NP-complete problems. *Theoretical Computer Science*, 63:239–252, 1989.
- [Du85] D.-Z. Du. *Generalized complexity cores and levelability of intractable sets*. PhD thesis, University of California, Santa Barbara, 1985.
- [ESY85] S. Even, A. Selman, and Y. Yacobi. Hard core theorems for complexity classes. *Journal of the ACM*, 35:205–217, 1985.
- [GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman and Company, 1979.

- [HS65] J. Hartmanis and R. E. Stearns. On the computational complexity of algorithms. *Transactions of the American Mathematical Society*, 117:285–306, 1965.
- [Huy87] D. T. Huynh. On solving hard problems by polynomial-size circuits. *Information Processing Letters*, 24:171–176, 1987.
- [Kar72] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–104. Plenum Press, 1972.
- [Lev73] L. A. Levin. On the notion of a random sequence. *Soviet Mathematics Doklady*, 14:1413–1416, 1973.
- [LM92] J. H. Lutz and E. Mayordomo. Measure, stochasticity, and the density of hard languages. Technical Report 92–11, Department of Computer Science, Iowa State University, 1992.
- [Lut92a] J. H. Lutz. Almost everywhere high nonuniform complexity. *Journal of Computer and System Sciences*, 44:220–258, 1992.
- [Lut92b] J. H. Lutz. Intrinsically pseudorandom sequences, 1992. in preparation.
- [Lut92c] J. H. Lutz. Resource-bounded measure, 1992. in preparation.
- [Lyn75] N. Lynch. On reducibility to complex or sparse sets. *Journal of the ACM*, 22:341–345, 1975.
- [May92] E. Mayordomo. Almost every set in exponential time is P-bi-immune. In *Seventeenth International Symposium on Mathematical Foundations of Computer Science*. Springer-Verlag, 1992. to appear.
- [Mey77] A. R. Meyer, 1977. reported in [BH77].
- [Orp86] P. Orponen. A classification of complexity core lattices. *Theoretical Computer Science*, 70:121–130, 1986.
- [OS86] P. Orponen and U. Schöning. The density and complexity of polynomial cores for intractable sets. *Information and Control*, 70:54–68, 1986.

- [RO87] D. A. Russo and P. Orponen. On P-subset structures. *Mathematical Systems Theory*, 20:129–136, 1987.
- [SC79] L. Stockmeyer and A. K. Chandra. Provably difficult combinatorial games. *SIAM Journal on Computing*, 8:151–174, 1979.
- [Ye90] H. Ye. Complexity cores for P/poly, 1990. submitted.



IOWA STATE UNIVERSITY

OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE

SCIENCE
with
PRACTICE

Tech Report: TR 92-23
Submission Date: August 13, 1992