M. Tech (Computer Science) Dissertation Report

Resource Bounded Measure and P versus NP problem - A Critical Study

A dissertation submitted towards partial fulfilment of the
requirements for the **M.Tech. (Computer Science)** degree of
Indian Statistical Institute

By
Chinmay Mukhopadhyay

Under the supervision of
Prof. K. Sikdar

Stat-Math Unit
INDIAN STATISTICAL INSTITUTE
203, Barrackpore Trunk Road
Calcutta-700035

July 27 , 2000

Certificate of Approval

This is to certify that the thesis entitled *Resource Bounded Measure and P versus NP problem - A Critical Study* submitted by *Chinmay Mukhopadhyay*, towards partial fulfilment of the requirement for *M.Tech.* in *Computer Science* degree of the *Indian Statistical Institute, Calcutta*, is an acceptable work for the award of the degree.

Date : July 27 , 2000 (Supervisor)

(External Examiner)

Acknowledgement

My sincerest gratitude goes to *Prof. K. Sikdar* for his guidance, advice, enthusiasm and criticisms throughout the course of this dissertation.

I would also like to take this oppurtunity to thank *Dr. Elvira Mayordoma* for providing me a helping hand.

Author

Chinmay Mukhopadhyay

Abstract

This reports deals with the concept of resource-bounded measure that provides a quantative approach to deal with many questions of Structural Complexity Theory. In particular , we examine the consequences of the hypothesis that NP does not have p-measure 0 which is is a stronger hypothesis than that of NP $\neq$ P.

In the first chapter , we introduce the concept of resource-bounded measure and survey some basic results about it . We also give examples of classes with p-measure 0 or p-measure 1 . In particular , (a) we give a sufficient condition for a class to have p-measure 0 in E and (b) we define , for each infinite language L $\in$ P , a class of languages X^L that has p-measure 0 . In the second chapter , we define a measure inside the class PSPACE . and survey some basic results about it . In the third chapter , we study nice properties like P-immunity and Incompressibility of languages in relation to resource bounded measure . Then we have made a small observation that if Bertman-Hartmanis conjecture is true then the class NPC has p-measure 0 in E. Also we have shown that if Berman-Hartmanis conjecture holds and for any two language L_1 and $L_2 \in$ NPI we have $L_1 \Delta L_2 \notin$ NPC then NP can have either measure 0 or it is non-measurable in E. In the fourth chapter , we present a notion of resource-bounded measure for P and other subexponential-time classes and examine it's basic properties . In Chapter 5 , we study the reasonableness and consequnces of the hypothesis " NP does not have p-measure 0 " .

PART I

Contents

1 Introduction and preliminaries
1.1 Introduction
1.2 Preliminaries
1.3 Resource bounded measure
1.4 Some technical lemmas
1.5 Γ-measurability and the Kolmogorov 0-1 Law

2 Measuring in PSPACE
2.1 Introduction
2.2 Measure in PSPACE
2.3 Γ-additivity in PSPACE

3 Quantative Structure of Exponential time
3.1 Introduction
3.2 Most is all
3.3 Incompressibility and Bi-immunity
3.4 Complexity Cores

4 Measure of nonuniform classes
4.1 Introduction
4.2 Formal Definitions of the Measure
4.3 Elementary Facts Concerning the Measure
4.4 Robustness, Alternative Formulations and Auxiliary Axioms
4.5 Hard Sets for BPP

5 If NP is not small
5.1 Introduction
5.2 If NP does not have p-measure 0
5.3 Separating completeness notions in NP
5.4 Separating reducibilities in NP
5.5 Further results and open problems

References

Chapter 1: Introduction and preliminaries

1.1 Introduction

Resource bounded measure , introduced by Lutz ([Lutz90],[Lutz92]) , is a generalization of the Lebesgue meaure restricted to the closed interval [0,1] , which can be identified with the set of all languages over {0,1} . It's formulation is basically motivated by the desire to provide quantative means for differnetiating various complexity classes such as P , NP etc .

In this survey , following [Lutz92] and [Mayo94] , we will introduce resource-bounded measure from the scratch and examine how this concept is useful in discussing various questions of structural complexity theory . In particular , we examine how it is useful to extend existence results of the form "There is a langauge L in C that is no in X " to abundance result of the form " most languages in C are not in X " , where C , X are complexity classes , e.g C=E , EXP , X=P etc.

Intutively , a class X has measure 0 in C when $X \bigcap C$ is negligibly small compared to C. We will be basically interested in classes with measure 0 or 1 because all those classes in which we are interested , are closed under finite variation and from the resource-bounded version of Kolmogorov 0-1 law it follows that these classes can have either measure 0 or 1 , if at all they are measurable .

We will also study here the reasonabality and consequences of the hypothesis "NP does not have measure 0 in E " .

In the course of study we have made a small contribution as described below.

(i) In Chapter 1 , (Proposition 1.1) , we have given a sufficient condition for a class to have p-measure 0 in E . Also for each infinite language in P we have defined , (Example 1.4) , a class of languages which has measure 0 in E .

(ii) In Chapter 3 , we have proved that (a) (Proposition 3.1) the class of NP-complete languages has p-measure 0 in E assuming Berman-Hartmanis isomorphism conjecture is true , and (b) (Proposition 3.2) if

(i) $L_1 \in NPI$, $L_2 \in NPI \Rightarrow L_1 \Delta L_2 \notin NPC$ and

(ii) Berman-Hartmanis conjecture holds , then either NP has p-measure 0 or it is not p-measurable .

1.2 Preliminaries

Let Σ={0,1} . A string is a finite sequence x $\in\{0,1\}^*$. We write $|x|$ for the length of x. The unique string of length 0 is λ, the empty string. If x and y are two strings, then $x < y$ if $|x| < |y|$ or $|x| = |y|$ and x precedes y in alphabetical order. We call this order relation on strings lexicographical order. Let s_0, s_1 , s_2 ... be the standard enumeration of the strings in $\{0,1\}^*$ in lexicographical order. A sequence is an element of $\{0,1\}^\infty$. If x is a string and y is a string or sequence, then xy is the concatenation of x and y. If x is a string and k $\in N \cup \infty$, then x^k is the k-fold concatenation of x with itself. If x is a string and y is a string or sequence, then x $\sqsubseteq$y iff there exists a string or sequence z such that y = xz, and x $\not\subseteq$ y if x $\subseteq$ y and x $\neq$ y. If w is a string or sequence and 0 $\leq$ i ¡ |w| then w[i] denotes the ith bit of w.

A language is a set of strings. A class is a set of languages. For each language A and n $\in$ N we denote as $A^{=n}$ the set of all strings in A of length n, and as $A^{\leq n}$ the set of all strings in A of length less or equal to n. Given a set A, we denote as $\wp(A)$ the power set of A, that is, the set of all subsets of A.

We will use the characteristic sequence χ_L of a language L, defined as follows: $\chi_L \in$ {0,1}$^\infty$ and χ_L[i] = 1 iff s_i belongs to L.

We identify through characteristic sequences the class $\wp(\{0,1\}^*)$ of all languages over {0,1} with the set $\{0,1\}^\infty$ of all sequences. Let w $\in \{0,1\}^*$. We define C_w , the cylinder generated by w, as the class of languages {x$\in\{0,1\}^\infty$: w$\subseteq$x} The complement of a class of languages X is X ^{c} = $\{0,1\}^\infty$-X. The complement of a language L is $\bar{L}=\{0,1\}^*$-L . For a class X $\subseteq \{0,1\}^\infty$, we define the class of complements as co-X = $\{\bar{L} \mid$ L $\in$ X} .

The symmetric difference of two sets A and B , denoted AΔB , is defined by AΔB =(A$\bigcup$B)-(A$\bigcap$B) .

Let X be a class of languages. We say that X is closed under finite variations , if A $\in$ X and $|A\Delta B| < \infty$, then B $\in$ X. We say that X is closed under finite translations if B $\in$ X and there exists w $\in \{0,1\}^*$ such that A= w·B , then A $\in$ X .

Next we fix some notation on complexity classes. For a complete introduction to Turing machines and complexity classes see for instance [BalcDig]. Our computation model is the multitape oracle Turing machine, with a read-only input tape and a write-only oracle tape. We will work with oracle Turing machines that halt on every oracle and every input. For a turing machine and a language A, L(M) denotes the set accepted by M with the empty oracle, and L(M,A) stands for the set accepted by machine M with oracle A. Given t: N $\rightarrow N$, we say that a Turing machine M recognizes a language L in time t when on each input x, M halts with output L(x) in time less or equal than t($|x|$). Analogously, M recognizes a language L in space t when on each input x, M halts with output L(x) using memory space less or equal than t($|x|$) . Here L(x)=1 if x $\in$ L and L(x)=0 if x$\notin$L .

For each nondecreasing function t: N $\rightarrow$ N, we denote by DTIME(t) the class of all languages that can be recognized by a deterministic machine in time t, and DSPACE(t) the class of all languages that can be recognized by a deterministic machine in space t.

Let NTIME(t) be the class of languages than can be recognized by a nondeterministic machine in time t, and let NSPACE(t) be the class of languages that can be recognized by a nondeterministic machine in space t. DTIMEF(t) and DSPACEF(t) are the corresponding classes of functions that can be computed in time t and space t, respectively. Unless indicated otherwise, when we bound the space used in the computation of a function we are also bounding the output space. For each language A, let DTIMEF^A(t) be the class of all fuctions that can be computed by a deterministic machine in time t when having access to oracle A ; and analogously we define DSPACEF^A (t). For each class F of functions from N to N, we write DTIME(F) for $\bigcup_{t\in F}$DTIME(t) and analogously for NTIME(F), DSPACE(F), NSPACE(F), DTIMEF(F) and DSPACEF(F). For each language A, DTIMEF^A(F) denotes $\bigcup_{t\in F}$DTIMEF^A(t) ,and in the same way we have DSPACEF^A (F). Let C be a class of languages. Then DTIMEF^C (F) =$\bigcup_{A\in C}$DTIMEF^A(F) and with a similar meaning DSPACEF^C(F) is defined.Let RE be the class of recursively enumerable languages, and REC be the class of recursive languages. We use the following notation for classes of languages

$$\mathrm{P} =\bigcup_{k\in N}\mathrm{DTIME}(\mathrm{n}^k) \qquad \mathrm{E}=\bigcup_{c>0}\mathrm{DTIME}(2^{cn})$$

$$\mathrm{E}_2 =\bigcup_{k\in N}\mathrm{DTIME}(2^{n^k}) \qquad \mathrm{NP} =\bigcup_{k\in N}\mathrm{NTIME}(\mathrm{n}^k)$$

$$\mathrm{NE}=\bigcup_{c>0}\mathrm{NTIME}(2^{cn}) \qquad \mathrm{LINSPACE} =\bigcup_{c\geq 0}\mathrm{DSPACE}(\mathrm{cn})$$

$$\mathrm{ESPACE} =\bigcup_{c>0}\mathrm{DSPACE}(2^{cn}) \qquad \mathrm{PSPACE} =\bigcup_{k\in N}\mathrm{DSPACE}(\mathrm{n}^k)$$

$$\mathrm{E_2SPACE} =\bigcup_{k\in N}\mathrm{DSPACE}(2^{n^k})$$

Let all be the class of all functions f : $\{0,1\}^* \to\{0,1\}^*$, and rec be the class of recursive functions in all. We will denote different classes of functions as follows.

$$\mathrm{p} =\bigcup_{k\in N}\mathrm{DTIMEF}(\mathrm{n}^k) \qquad \mathrm{pspace} =\bigcup_{k\in N}\mathrm{DSPACEF}(\mathrm{n}^k)$$

$$\mathrm{p}_2=\bigcup_{k\in N}\mathrm{DTIMEF}((2^{logn})^k) \qquad \mathrm{p_2space}=\bigcup_{k\in N}\mathrm{DSPACEF}((2^{logn})^k)$$

1.3 Resource-bounded measure

Our goal is to define a measure in C, where C can be one of the following E , E$_2$, ESPACE , E$_2$SPACE and REC. Intuitively , a measure in C is a function μ:$\wp$(C)$\to$ [0, 1] with some additivity properties . Now given a class C we can very well define a measure μ in C as a restriction of Lebesgue measure to C . But since Lebesgue measure of any countable class has Lebesgue measure 0 and recursive classes are always countable , thus $\mu \equiv 0$. Now to define a nontrivial measure on countable classes Lutz considered an alternative but equivalent formulation of the concept of Lebesgue measure in terms of the concept of a martingale and then obtaining nontrivial measures on such classes by suitably restricting the martingales . We next give an exposition of this following [AmboMay] .

To introduce the concept of a martingale , let us consider a game in which there is a player with starting capital $0 < c_0 \in$ R and a hidden language L. The player bets part of his money on the successive bits of χ_L , making money on a double or nothing fashion. (We can imagine an infinite number of boxes marked with 0 ,1 , 2 ... so that ith box contain 0 if $s_i \notin$ L and it contains 1 if $s_i \in$ L . Content of the ith box can be checked only after the player bets an amount of his money either on $s_i \in$ L or on $s_i \notin$ L .) The game goes as follows.

Step 0: The player bets a_0 , a part of c_0 , either that $s_0 \in$ or that $s_0 \notin$ L. If he wins, he gets double, that is $2a_0$ and his capital is now $c_1 = c_0 + a_0$. If he loses, he gets nothing and his capital is now $c_1 = c_0$-a_0.

Step n, n > 0: Now the player has got the information on $[[s_0 \in L]] \dots [[s_{n-1} \in L]]$. Using this information the player bets a_n , a part of c_n, either that $s_n \in L$ or $s_n \notin L$ .If he wins he gets double ,that is If he loses, he gets nothing and his capital is now $2a_n$ and his capital is now $c_{n+1} = c_n + a_n$ .If he loses , he get nothing and his capital is now $c_{n+1} = c_n$ - a_n.

The game goes on eternally,and we say that the player succeeds if

$$\lim \sup_n c_n = \infty$$

The player tries to find a betting strategy that is always useful. A strategy for this game is a function a: $\{0,1\}^* \to \{0,1\} \times [0,\infty)$ that tells the player how much to bet, depending on the information the player has. That is, if $[[s_0 \in L]] \dots [[s_{n-1} \in L]] = w$, $w \in \{0,1\}^*$,and a(w)=(b,u) ,the player should bet an amount a_n= u that $[[s_n \in L]]$=b,according to strategy a.

We can now compute the capital a player has when using this strategy a and represent it via a function $d_a:\{0,1\}^* \to [0,\infty)$ with the meaning that, $[[s_0 \in L]] \dots [[s_{n-1} \in L]] = w$, $w \in \{0,1\}^*$, then the player's capital , after having bet on $s_0 \dots s_{n-1}$according to a ,is $c_n = d_a(w)$,we also have $c_0 = d_a(\lambda)$.

From a we can compute d_a and vice versa:

$$a(w) = \begin{cases} (0, d_a(w0) - d_a(w)) & \text{if } d_a(w0) \geq d_a(w), \\ (1, d_a(w1) - d_a(w)) & \text{if } d_a(w1) \geq d_a(w). \end{cases}$$

let b∈ {0,1}

$$d_a(wb) = \begin{cases} d_a(w) + u & \text{if } a(w)=(b,u), \\ d_a(w) - u & \text{if } a(w)=(1-b,u). \end{cases}$$

From now on we represent a strategy a by its capital function d_a,which we call **a martingale**

Definition 1.1. A martingale is a function d: $\{0,1\}^* \to [0,\infty)$ satisfying

$$d(w) = \frac{d(w0)+d(w1)}{2}$$

for all $w \in \{0,1\}^*$

Notice that if d is a martingale then for each $w \in \{0,1\}^*$ $d(w) \leq 2^{|w|} \cdot d(\lambda)$.

Since $d(wb) \leq 2d(w)$ where $b \in \{0,1\} \Rightarrow d(w) \leq 2^{|w|} d(\lambda)$.

Definition 1.2. A martingale d is said to be successful for a language x $\in \{0,1\}^\infty$ iff $\lim_{n\to\infty} \sup d(x[0\dots n]) = \infty$.

We observe that the player can succed on a language x iff ∃ a martingale d such that

$$\lim_{n\to\infty} \sup d(x[0\dots n]) = \infty.$$

For each martingale d, we denote the set of all languages for which d is successful as $S^{\infty}[d]$, that is $S^{\infty}[d]= \{x \mid \lim_{n\to\infty} \sup d(x[0\ldots n]) = \infty\}$. We next give a characterization of the classes with Lebesgue measure 0 in terms of the martingale.

Now , Lebesgue measure in the interval [0,1] can be defined in terms of the basic open sets $B_x = \{ y : y \sqsubseteq x \}$ and $\mu(B_x) = 2^{-x}$, for $x \in \{0,1\}^*$. From following theorem , due to Ville , we get a nice characterization of sets with Lebesgue measure 0 in terms of martingale .

Theorem 1.1. (Ville[Vi39])

Let C be a class then $\mu(C)=0$ iff there exists a martingale which suceeds on C.

Proof. See [AmboMay]

Definition 1.3. A class $X \subseteq \{0,1\}^{\infty}$ has Lebesgue-measure 0 iff there exists a martingale d such that X , that $X \subseteq S^{\infty}[d]$,that is for any $L \in X$, d is successful for L.

Intuitively, a class X has measure 0 when there exists a single strategy that is good for predicting any language in the class X.

Definition 1.4. A class $X \subseteq \{0,1\}^{\infty}$ has Lebesgue-measure 1 iff X^c (the complement of X) has Lebesgue measure 0.

We only define measure 0 and measure 1 because , as mentioned earlier , we are always interested in classes that are closed under finite variations, and from resource bounded version of Kolmogorov 0-1 law it follows that these classes can only have measure 0 or measure 1, if at all they are measurable .

Going back to the initial problem of defining a non trivial measure inside REC, E, E_2, ESPACE or E_2SPACE, what we do next is to restrict the martingales that can witness that a class has measure 0. We will require the martingales to be recursive and computable within certain time and space bounds, depending on the class where we are defining a measure. Further we see that it is enough to consider dyadic rational valued martingales as the lemma states below .

Lemma 1.1. (Mayordoma [Mayo94]) For each martingale d, there exists a martingale $\hat{d}:\{0,1\}^* \to D$ such that $S^{\infty}[d] = S^{\infty}[\hat{d}]$.

Here , $D = \{m2^{-n} : m,n \in N \}$ is the set of non-negative dyadic rational numbers.

We now define the concept of resource-bounds that are classes of recursive functions. By requiring the martingales to be in a certain measure resource-bound we will define measures for different classes.

We say that a set F of functions from N to N is a family of bounds if all functions in F are nondecreasing and for each f ,g $\in$F , f$\circ$g is also in F .

Definition 1.5. A class $\Gamma \subseteq$ all is a measure resource-bound if $p \subsetneq \Gamma$ and Γ is one of the following cases :

a) Γ= **all**

b) Γ=DTIMEF^C(F) for F a family of bounds and C a family of language,

c) Γ=DSPACEF^C(F) for F a family of bounds and C a family of language.

Where **all** is the class of all functions f:$\{0,1\}^* \to \{0,1\}^*$.

We are specially interested in the following measure resource-bounds: p, p_2, pspace, $p_2$space and rec, as we will see below.

As in the case of Lebesgue measure, there exists a resource-bounded generalization of the Kolmogorov 0-1 law by which classes that are closed under finite variations can only be in one of three cases, namely being Γ-measure 0,being Γ-measure 1 and being non-Γ-measurable. For this reason we only define to find the appropriate Γ-measure 0 and Γ-measure 1.

Now for each measure resource-bound Γ,we define μ_Γ as a restriction of Lebesgue measure to martingales in Γ.We then use μ_Γ to define a nontrivial measure on a suitable recursive class C.

Definition 1.6. A class X$\subseteq\{0,1\}^\infty$has Γ-measure 0 (and we denote it μ_Γ(X)=0) iff there exists a martingale d$\in$ Γ such that X$\subseteq$S$^\infty$[d].

Definition 1.7. A class X$\subseteq\{0,1\}^\infty$ has Γ-measure 1 (and we denote it μ_Γ(X)=1) iff X^c has Γ-measure 0.

Notice that taking Γ= **all** we get back the definition of Lebesgue measure 0 and Lebesgue measure 1 sets (which follows from the Theorem 1.1).

Throughout this chapter we will take Γ to be a measure resource bound .

Definition 1.8. $f \in \Gamma$ is said to be a constructor iff $\forall$w $\in\{0,1\}^*$ w $\subset f$ (w) and w $\neq f$(w).

Definition 1.9. If h is a constructor in Γ ,then R(h) is the unique element in $\{0,1\}^\infty$ such that $\forall$i h$^i(\lambda) \subseteq$ R(h).

Observation R(h)=$\lim_{i\to\infty}$ **h**$^i(\lambda)$.

Definition 1.10. R(Γ) is the class of all languages {R(h) | h is a constructor in Γ }.

Lemma 1.2. [Lutz90].

R(all) =$\{0,1\}^\infty$, R(p_2) = E_2 ,
R(rec) = REC, R(pspace) = ESPACE,
R(p) = E, R($p_2$space) = E_2SPACE. (Mayordoma [Mayo94])

We will now use Γ-measure to define a nontrivial measure on the class R(Γ) . The justification of why the defined measure is nontrivial is given by the following theorem which states that R(Γ) does not have Γ-measure 0 .

Theorem 1.2. [Lutz92] (Measure Conservation Theorem) For every martingale d$\in \Gamma$ there exist a language L $\in$R(Γ) such that d is not successful on L.

Definition 1.11. A set X$\subseteq\{0,1\}^\infty$ has measure 0 in R(Γ) iff X $\bigcap$ R(Γ) has Γ-measure 0.This is denoted as μ_Γ(X | R(Γ))=0.

Definition 1.12: A set X$\subseteq\{0,1\}^\infty$ has measure 1 in R(Γ) iff X^c $\bigcap$ R(Γ) has Γ-measure 0.This is denoted as μ_Γ(X | R(Γ))=1.

We note that for each martingale d $\in \Gamma$ the class S$^\infty$[d]$\bigcap$R(Γ) has Γ-measure 0 , where as R(Γ)-S$^\infty$[d] has Γ-measure 1 .

We now state a few known results about Γ-measure .

Lemma 1.3.

Let X,Y$\subseteq\{0,1\}^\infty$,

a) If Y $\subseteq$ X and X has Γ-measure 0 Then Y has Γ-measure 0.

b) If Y $\subseteq$ X and X has measure 0 in R(Γ) Then Y has measure 0 in R(Γ).

c) if X has Γ-measure 0 Then X has measure 0 in R(Γ).

Proof. Follows from definition.

Lemma 1.4. (Mayordoma [Mayo94])

Let n $\in$N. If X$_1$ ... X$_n$ have measure 0 in R(Γ) then $\cup_{1\leq i\leq n}$ X$_i$ has measure 0 in R(Γ).

Thus we note that the measure that we are dealing with is finitely additive but we see that it is not countably additive as example for each A $\in$ R(Γ) {A} has Γ-measure 0 in R(Γ) where as R(Γ) = $\bigcup_{A\in R(\Gamma)}$ {A} does not have Γ-measure 0 .

Lemma 1.5. (Mayordoma [Mayo94])

Let X $\subseteq\{0,1\}^\infty$. Let Γ, Γ' , be two resource-boundeds such that $\Gamma \subseteq \Gamma'$ if X has Γ-measure 0 then X has Γ'-measure 0 , and If X has Γ-measure 1 then X has Γ'-measure 1.

Proof. If X has Γ-measure 0 then clearly X has Γ'-measure 0 . Now X has Γ-measure 1 implies that X^c has Γ-measure 0 so that X^c has Γ'-measure 0 and hence X has Γ'-measure 1. Hence the Proof.

We now give some examples of classes that are of measure 0 and of measure 1 in E=R(p).

Example 1.1 Let C ={ L$_1$, L$_2$ ... } be a countable class of langauges Let $\{n_k\}_{k\in N}$ be such that for each i , either $\chi_{L_i}[n_k]$=1 $\forall$ k or $\chi_{L_i}[n_k]$=0 $\forall$ k . Note that such a sequence exists by digonalization arguement . Further assume that the language M = { s_{n_k} : k $\geq$ 1 } $\in$ P . Then C has p-measure 0.

Proof. To see this note that , as C = C$_1\bigcup$C$_2$ where

$$C_1 = \{ L : \chi_{L_i}[n_k]=1 \ \forall k \} .$$
$$C_2 = \{ L : \chi_{L_i}[n_k]=0 \ \forall k \} .$$

Now the betting strategy for C_1 will be , for w $\in \{0,1\}^*$ if w $\in$ M bet all the money on w$\in$ L othewise do not bet on that particular bit.

And the betting strategy for C_2 will be , for w $\in \{0,1\}^*$ if w $\in$ M bet all the money on w$\notin$ L othewise do not bet on that particular bit.

Clearly C_1 and C_2 has p-measure 0 and so C = $C_1 \bigcup C_2$ has p-measure 0.

Thus we can state that

Proposition 1.1 Given a class of a language C if we can find a subsequence n_k of natural number such that for each L in C either $\chi_L[n_k]=1\ \forall$ k or $\chi_L[n_k]=0$ $\forall$ k and if the language consisting L = { s_{n_k} $k \geq 1$ } is in P then the class C has p-measure 0.

Example 1.2. [Mayo94] The class X = { A | there exist n such that $|A^{\leq n}|$ is not a multiple of 3} has measure 1 in E.

Proof . We will show that the class Y = X^c has measure 0 in E.

Let A be a language in Y . For every n $\in$N, $|A^{\leq n}|$ is a multiple of 3. If we know the value of $|A^{\leq n}$-$\{1^n\}|$ we can guess whether 1^n is in A or not since if $|A^{\leq n}$-$\{1^n\}|$ is a multiple of 3 then 1^n must be out of A (if not $|A^{\leq n}|$=3·k+1 for some k which would be a contradiction) also if $|$ $A^{\leq n}$-$\{1^n\}|$ is a multiple of 3 plus two ,then 1^n must be in A

Thus a succesful strategy for Y will be to bet only on bits corresponding to strings of the form 1^n if $|A\leq^n$-$\{1^n\}|$ is a multiple of 3 we bet all our money to $1^n \notin$ A, else we bet all our money to $1^n \in$ A.

Notice that s_i is of the form 1^n if and only if i is of the form 2^{m-2}.

We define a martingale d that corresponds to the described strategy.Let d(λ)=1. For each w$\in\{0,1\}^*$ We define d(w0) and d(w1) be as follows ,

if $|w|$ $=2^{m-2}$ for some m then

$$d(w0) = \begin{cases} 2 \cdot d(w) & \text{if } \sum_{0\leq i\leq |w|-1} \text{w[i] is a multiple of 3,} \\ 0 & \text{otherwise.} \end{cases}$$

$$d(w1) = \begin{cases} 2 \cdot d(w) & \text{if } \sum_{0\leq i\leq |w|-1} \text{w[i] is not a multiple of 3,} \\ 0 & \text{otherwise.} \end{cases}$$

Else,if |w| is not of the form 2^m-2 then d(w0)=d(w1)=d(w).

Let us see that d is successful on all languages in Y . Let A $\in$ Y, n $\in$ N. Then $|A\leq^n|$ $=\sum_{0\leq i\leq 2^{n+1}-2}$A[i] is a multiple of 3.if $|A\leq^n$-$\{1^n\}|$=$\sum_{0\leq i\leq 2^{n+1}-3}$A[i] is a multiple of 3,then $1^n \notin$ A and A[2^{n+1}-2]=0. By the definition of d then d(A[0.... 2^{n+1}-2])=2. d(A[0.... 2^{n+1}-3]), if $|A^{\leq n}$-$\{1^n\}|$ is not a multiple of 3,then $1^{n\in}$ A and d(A[0.... 2^{n+1}-2])=2. d(A[0.... 2^{n+1}-3]).

Since we only bet on bits of the form 2^m-2,then for each n$\geq$1 d(A[0.... 2^{n+1}-3])= d(A[0.... 2^n-2]).Thus d(A[0.... 2^{n+1}-2]) =2. d(A[0.... 2^n-2]), $\lim_{m\to\infty}$ sup d(A[0. . .m])=∞ and Y$\subseteq S^\infty$[d]

Also, d is a martingale in p, because for each input w we can compute d(w) from d(w[0. . . |w|-2]) just by checking whether $|w|$ is of the from 2^m-2,and the computing $\sum_{0\leq i\leq |w|-1}$w[i],all of which can be done in time linear in

$|w|$,computing d(w) requires computing d(u) for each u prefix of w, and can thus be done in time quadratic in $|w|$. This proves that Y has p-measure 0 and hence we have that X = Y^c has measure 1 in E.

Example 1.3. [Mayo94]

The class X = {A | for every n$\in \mathcal{N}$ such that $|A^{=n}| \geq \frac{2}{3}2^n$ } has measure 0 in E

Notice that for every A, $|A^{=n}| \leq 2^n$ for every n.

Proof . The only information about the language in X is that they have more strings Our strategy is to bet half of our money on $s_i \in$ A for any language A $\in$ X .

We define a martingale d as follows. d(λ) = 1 For each w $\in$ {0,1}* ,

$$d(w0) = \frac{1}{2}d(w) ,\ d(w1) = \frac{3}{2}d(w)$$

Let us see that d is successful on every language in X. If A $\in$X then for each n$\in$N we have that

d(A[0...2^{n+1}-2])= $(\frac{3}{2})^a(\frac{1}{2})^b$d(A[0...$2^n$-2])where a=$|A^{=n}|$ and b=2^n-a

$\geq(3/2)^c(1/2)^d$d(A[0...2^n-2])where c=$\frac{2}{3}2^n$ and d=$\frac{1}{3}2^n$

$\geq(\frac{9}{8})^d$d(A[0...2^n-2])

This implies that $\lim_{m\to\infty}$ sup d(A[0. . .m])=∞ .

Clearly , d is computable in linear time, thus X has p-measure 0. Therefore , X has measure 0 in E.

We can make a slight generalization of Example 1.3 as

Example 1.4

The class X^L = {A | for every n$\in \mathcal{N}$ such that $|(A\bigcap)L^{=n}| \geq \frac{2}{3}\ |L^{=n}|$ } has measure 0 in E for any L $\in$ P such that $|L| = \infty$.

Proof. We define a martingale d^L as follows. $d^L(\lambda)$ = 1 For each w $\in$ {0,1}*, if w $\in$ L then

$$d^L(w0) = \frac{1}{2}d^L(w) ,\ d^L(w1) = \frac{3}{2}d^L(w)$$

else

$$d^L(w0)=d^L(w1)=d^L(w).$$

Clearly , d^L is a martingale and $d^L \in$ p.

Now . similiar calculation as that of Example 1.3 , we see that d^L succeeds for all languages in X^L whenever L $\in$ P.

1.4 Some technical lemmas

In this section we summarize some technical results obtained by Lutz [Lutz92] which are useful in checking whether a given class has measure 0 or measure 1 in R(Γ) .

Notation: Given two sets X,Y , we consider each function f: N$\times$X $\to$Y as an enumeration of the functions f_k, k $\in$ N where for each k$\in$N, f_k : X $\to$Y is defined as $f_k(x) = f$ (k,x) for every x∈X. In the same way we considea each function f : $N^n \times X \rightarrow Y$ as an enumeration of the functions $f_{\vec{k}}$ for $\vec{k} \in N^n$.

Definition 1.14. Let X be the cartesian product of a finite number of factors of the form N and $\{0,1\}^*$. A function $f \in \Gamma, f : N \times X \rightarrow D$ is a Γ-computation of a computation of a function g:X→[0,∞) iff $|f_k(w)$- $g(w)| \leq 2^{-k}$ for all w∈X and k∈ N.

Notice that we are trying to approxiamate the function g by the sequence of functions f_k where $f_k \in \Gamma \ \forall$ k , however g may not be in Γ

Definition 1.15. A function g:X→[0,∞) is Γ-computable iff there exists a Γ-computation of g.

Lemma 1.7. Exact Computation Lemma. (Mayordoma [Mayo94])

For each Γ-computable martingale d there exists a martingale $\hat{d}$ in Γ such that $S^\infty[d] = S^\infty$ $[\hat{d}]$.

So if we can compute a good approxiamation of a martingale in bound specified by Γ , that would be enough to serve our purpose.

Corollary. Let X be a class of languages , X has Γ-measure 0 iff there exists a Γ-computable martingale d such that X ⊆ $S^\infty[d]$.

Now , as we have already noticed , a countable union of Γ-measure 0 classes is not necessarily of Γ-measure 0 . We find a weaker version of countable additivity .

Defintion 1.16. An n-dimensional martingale system (n-MS) is a funcion d : $N^n \times \{0,1\}^* \rightarrow [0,\infty)$

such that $d_{\vec{k}}$ is a martingale for every $\vec{k} \in N^n$.

We now define a restricted notion of countable union , that is called Γ-union. This concept is only defined for Γ-measure 0 sets.

Defintion 1.17. A set X is a Γ -union of the Γ-measure sets X_0 ,X_1,X_2... iff $X = \bigcup_{j \geq 0} X_j$ and there exists a Γ-computable 1-MS d such that for every j,

$$X_j \subseteq S^\infty[d_j].$$

Lemma 1.8. (Γ-additivity Lemma.)

Proof. See (Mayordoma [Mayo94])

If X is a Γ-union of the Γ-measure sets 0,then X has Γ-measure 0.

Thus we have finally got a weaker notion of countable additivity for Γ-measure.

Definition 1.18. A set X is a Γ-union of measure 0 in R(Γ) sets X_0, X_1, $X_2 \ldots$ iff $X \bigcap R(\Gamma)$ is a Γ-union of the Γ-measure 0 sets $X_0 \bigcap R(\Gamma)$, $X_1 \bigcap R(\Gamma)$, $X_2 \bigcap R(\Gamma)$,...

Corollary . If X is a Γ-union of measure 0 in R(Γ) sets,then X has measure 0 in R(Γ).

Now we give one example to show how the concept of Γ-union is useful to find out Γ-measure of a given class .

Example 1.5. [Mayo94]

The class X=$\{A \mid$ for all but finitely many n$\in N$, $|A^{=n}| \geq \frac{2}{3}2^n\}$ has p-measure 0 .

Proof . We start by writing X as a countable union of classes. For each i$\in$N let

$X_i = \{A \mid$ for every n$\geq$i ,$|A^{=n}| \geq \frac{2}{3}2^n$ }

Clearly X=$\cup_i X_i$.

We want to show that X has p-measure 0 by proving that X is a p-union of the measure 0 sets X_i.Therefore we have to define a p-computable 1-MS d such that for each i, $X_i \subseteq S^\infty[d_i]$.

For each i$\in N$, $d_i(\lambda)$=1 and for each w$\in\{0,1\}^*$,

If $|w| \geq 2^i$ -1 then

$$d_i(w0)=\frac{1}{2}d_i(w) \ d_i(w1)=\frac{3}{2}d_i(w)$$
$$\text{else if } |w| < 2^i \text{ -1 then } d_i(w0)= d_i(w1)= d_i(w).$$

(Where $d_i(w)$=d(i,w))

For each i$\in N$ we note that $X_i \subseteq S^\infty[d_i]$.

To check whether $|w| < 2^i$-1 we just need to write $|w|$ in binary and count the number of bits used ,comparing it with i.Thus d can be computed in time linea in $|w|$+i,is trivially p-computable and X has p-measure 0.

The next theorem has a number of interesting corollaries .

Theorem 1.3. (Mayordoma [Mayo94]) Let Γ,Γ'be two measure resource-bounds such that Γ'contains a universal function for Γ,that is $\exists f \in \Gamma'$with $\Gamma=\{f_i \mid i \in N\}$ .Then the class X=$\cup_{\mu_\Gamma(Y)=0}$ Y has Γ' -measure 0.

Corollary . Let Γ,Γ'be two measure resource-bounds such that Γ'contains a universal function for Γ.Then R(Γ) has Γ'-measure 0.

We recall that a language is said to be Γ-random if it belongs to all classes which has Γ-measure 1 .

Corollary . E has measure 0 in E_2 . ESPACE has measure 0 in E_2SPACE. The class of p-random languages has measure 1 in E_2 . The class of pspace-random languages has measure 1 in E_2SPACE.

We see that most of the languages in E_2 are not in E .

Lemma 1.9. (Mayordoma [Mayo94])

For every c > 0,

$$\mu(\text{DTIME}(2^{cn}) \mid \text{E}) = 0 \text{ ,}$$

and

$$\mu(\mathrm{DSPACE}(2^{cn}) \mid \mathrm{ESPACE}) = 0.$$

This shows that , for each $c > 0$, DTIME(2^{cn}) is a small class in comparison to E , and DSPACE(2^{cn}) is a small class in comparison to ESPACE .

Lemma 1.10. (Classical first Borel-Cantelli Lemma.)

Let $\{X_j \subseteq \{0,1\}^\infty \mid j \in N\}$ sequence of Lebesgue-measurable sets such that $\sum_{j=0}^{\infty} \Pr(X_j\ )$ is convergent, then $\Pr(\bigcap_{t=0}^{\infty} \bigcup_{j=t}^{\infty} X_j)=0$

We notice that the class $\bigcap_{t=0}^{\infty} \bigcup_{j=t}^{\infty} X_j$ consists exactly of those x that belong to X_n for an infinitely many n.

We are interested in classes of languages that can be represented with this kind of expressions. To study their measure , an appropriate resource-bounded version has been formulated . For a translation of the classical Borel-Cantelli Lemma to Γ-measure we need resource-bounded version of idea of a family of classes with Lebesgue measure decreasing quickly to 0. To do this, a way of saying " X has Γ-measure smaller than μ" has been introduced for a class X and $\mu > 0$.

For each martingale d and $r > 0$, we define the class

$$S^r\ [d] = \{A \mid lim_{m \to \infty} d(A[0 \ldots m]) \geq d(\lambda) \cdot r\}$$

We interpret $X \subseteq S^r[d]$ as X has measure smaller than $\frac{1}{r}$.

For resource-bounded version of the Borel-Cantelli Lemma a restrictive notion of convergence of series is used.

Definition 1.19. Let $\{a_n \mid n \in N\}$ be a sequence of nonnegative real numbers. A modulus for the series $\sum_{n=0}^{\infty} a_n$ is a function $m: N \to N$ such that $\sum_{n=m(j)}^{\infty} a_n \leq 2^{-j}$ for all $j \in N$. A series is Γ-convergent if it has modulus that is in Γ.

Definition 1.20. Let $\{a_{j,k} \mid j,k \in N\}$ be a sequence of non-negtive real numbers.A Sequence $\sum_{k=j}^{\infty} a_{j,k}$ (j=0,1,2,...) of series is uniformly Γ-convergent if there exists a function $m: N^2 \to N$ such that $m \in \Gamma$ and for each $j \in N, m_j$ is a modulus for the given series.

Finally we state the resource-bounded generalization of the classical first Borel-Cantelli Lemma

Lemma 1.11. [Lutz92]. Let $\{X_{i,j} \subseteq \{0,1\}^\infty \mid i,j \in N\ \}$.If there exists d a Γ-computable 2-MS such that

(i) $\forall\ i,j \in N\ , X_{i,j} \subseteq S^{\frac{1}{d_{i,j}}(\lambda)}\ [d_{i,j}]$

(ii) the series $\sum_{j=0}^{\infty} d_{i,j}(\lambda)$ (i=0,1,2,...)

are uniformly Γ-convergent,

then

$$\mu_{\Gamma}(\bigcup_{i=0}^{\infty} \bigcap_{t=0}^{\infty} \bigcup_{j=t}^{\infty} X_{i,j}) = 0$$

Proof. See [Mayo94] .

In order to take full advantage of this lemma the following sufficient condition for uniform Γ-convergence is useful .

Lemma 1.12. (Mayordoma [Mayo94]) Let $a_{j,k} \in [0, \infty)$ for all j, k ∈ N. If there exist a real $\epsilon > 0$ and a polynomial g: N → N such that $a_{j,k} \leq \epsilon^{-k}$ for all j, k ∈ N with $k \geq g(j)$,then the series

$$\sum_{k=0}^{\infty} a_{j,k} \qquad\qquad (j=0,1,2 \ldots)$$

are uniformly Γ-convergent.

Let us see an application of the resource-bounded Borel-Cantelli Lemma

Example 1.6. [Mayo94] The class X ={A | $|A^{=n}| \geq 2^n(\frac{1}{2}+\frac{1}{n})$ for infinitely many n} has measure 0 in E .

Proof . For each n ∈ N, let

X_{2^n} ={A | $|A^{=n}| \geq 2^n(\frac{1}{2}+\frac{1}{n})$}

and let $X_j=\emptyset$; if j is not a power of 2. Then by definition of X,

$X = \bigcap_{t=0}^{\infty}\bigcup_{j=t}^{\infty} X_j$.

We want to apply Γ-additivity lemma to this expression of X. Notice that we do not have the outermost union. It is enough to define a 1-MS d such that for each j, $X_j \subseteq S^{\frac{1}{d_j(\lambda)}}[d_j]$ for each j ∈ N such that j is a power of 2, w ∈ $\{0,1\}^*$ d(j,w) = Pr[x ∈X_j | x ∈ C_w], for the rest of j let $d_j \equiv 2^{-j}$. By definition of conditional probability , d is a 1-MS . We have to show that d is p-computable and that condition (i) and (ii) of the above lemma hold . To see that condition (i) holds, fix j ∈ N a power of 2. and x∈ X_j.Since this condition x ∈ X_j is only based on the prefix x[0. . .2j-2], and y ∈ $\{0,1\}^\infty$ such that y ∈ $C_{x[0\ldots 2j-2]}$ is also in X_j and d(j,x[0. . .2j-2])=1,thus x ∈ $S^{\frac{1}{d_j(\lambda)}}[d_j]$. To see that condition (ii) holds, we have to look at the series $\sum_{j=0}^{\infty} d_j(\lambda)= \sum_{j=0}^{\infty}$ Pr[x∈X_j]. for each n∈ N and $j=2^n$ Pr[x∈X_j] $\leq e^{-\frac{j}{6n^2}}$ thus there exists c > 0 such that j > c then Pr[x∈X_j] $\leq e^{-j^{.5}}$ which shows that $\sum_{j=0}^{\infty} d_j(\lambda)$ is p-convergent and condition (ii) holds .

We need to check that d is p-computable .We can use binomial coeffiecients to exactly compute Pr[x∈ X_j | x∈C_w] in time polynomial in |w| +j,thus d ∈ p and we have the result.

1.5 Γ-measurability and the Kolmogorov 0-1 law

In this section we define the concept of Γ-measurability and consider the classes that have Γ-measure μ, for μ any value between 0 and 1. Then we see that these new concept are not useful for classes that are closed under finite variations, rather we see that classes which are closed under finite variation can have either measure 0 or measure 1 if at all the are measureable . This is stated as the resource-bounded version of the Kolmogorov 0-1 law, which is a consequence of the classical Kolmogorov 0-1 law.

Definition 1.21. Let μ^*_Γ:$\{0,1\}^\infty \rightarrow \wp([0,1])$ be the function that for each X$\in\{0,1\}^\infty$ is defined as follows

μ^*_Γ(X=$\{\gamma \geq 0$ | there exists 1-MS d$\in \Gamma$ such that for each k $\in$ N ,X$\subseteq$ $S^{\frac{1}{\gamma+2^{-k}}}[d_k]\}$.

Note that μ^*_Γ is a function mapping a langauge to a subset of [0,1] .

The next lemma states some basic properties of μ^*_Γ

Lemma 1.13.

(i) If X $\subseteq$ Y then $\mu^*_\Gamma(Y) \subseteq \mu^*_\Gamma(X)$

(ii) Let $\{Y_0,\ldots,Y_n\}$ be a finite sequence of pairwise disjoint classes If $\gamma_i \in \mu^*_\Gamma(Y_i)$ for each i then $\sum_{i=1}^n \gamma_i \in \mu^*_\Gamma$ $(\bigcup_{i=1}^n Y_i)$

(iii)for every X , $\mu^*_\Gamma(X)+\mu^*_\Gamma(X^c)\geq 1$

(iv)Let $\{Y_i \mid i \in N\}$ be a sequence of pair-wise disjoint classes .If $\gamma_i \in \mu^*_{all}$ then $\sum_{i\in N} \gamma_i \in \mu^*_{all}(\bigcup_{i\in N} Y_i)$

(v)If $\{a_n \mid n \in N\}$ is a decreasing sequence such that $\{a_n \mid n \in N\} \subseteq \mu^*_{all}(X)$ and $l=lim_{n\rightarrow\infty}a_n$ then $l \in \mu^*_{all}(X)$.

Proof . See Mayordoma [Mayo94] .

Definition 1.22. Let X be a class of languages.We say that X is Γ - measurable if there exists $\gamma \in \mu^*_\Gamma(X)$,

$\gamma' \in \mu^*_\Gamma(X^c)$ such that $\gamma+\gamma'=1$.

Notice that if X is Γ-measurable then there exists a unique $\gamma \in \mu^*_\Gamma(X)$, $\gamma' \in \mu^*_\Gamma(X^c)$ such that $\gamma+\gamma'=1$. Since let γ_1 , $\gamma_2 \in \mu^*_\Gamma(X)$ and δ_1 , $\delta_2 \in \mu^*_\Gamma(X^c)$ such that

$\gamma_1+\delta_1=1$

$\gamma_2+\delta_2=1$

Then by part(iii) of above lemma $\gamma_1+\delta_2 \geq 1$ and $\gamma_2+\delta_1 \Leftarrow \gamma_1+\delta_2=1$ and $\gamma_2+\delta_1=1$ (Since $\gamma_1+\delta_1+\gamma_2+\delta_2 = 2$.

We denote $\mu_\Gamma(X)=\gamma$.

For Γ=**all** this definition corresponds to classical Lebesgue measurability , which has been shown by Lutz [Lutz92] .

Lemma 1.14. Let Γ, Γ' be two measure resource-bounds such that $\Gamma \subseteq \Gamma'$ If X is Γ-measurable then X is Γ'-measurable and $\mu_\Gamma(X)=\mu_{\Gamma'}(X)$.

Proof. See Mayordoma [Mayo94]

We notice that if one can prove that a class of language has Γ-measure 0 for some resource bound Γ then X will have Lebesgue measure 0 , as example all classes defined in example 1.4 have Lebesgue measure 0 , and there is no obvious way to find out their lebesgue measure , thus some times we could even try resource bounded measure for finding out whether a class has lebesgue measure 0 .

Lemma 1.15. X has Γ-measure 0 iff X is Γ-measureable and $\mu_\Gamma(X)$=0

Proof. See Mayordoma [Mayo94] .

This shows that Definition 1.22 is consistent with our earlier definition of Γ-measure 0 and Γ-measure 1 sets. Thus we can use any one of the two definition for Γ-measure 0 , 1 sets .

Theorem 1.4 (Resource bounded version of Kolmogorv 0-1 law). Let X be a class of languages that is closed under finite variations. If X is Γ-measurable then either X has Γ-measure 0 or Γ -measure 1

Proof. See Mayordoma [Mayo94] .

Chapter 2: Measuring in PSPACE

2.1 Introduction

In Chapter 1 , we have defined Lutz's resource-bounded measure for classes such as E, E_2, ESPACE and E_2SPACE . However, there are interesting problems that can be formulated dealing with the estimation of the size of subclasses of P or PSPACE. For instance, we may want to know whether most languages in P are efficiently parallelizable, or whether self-reducibility is a typical property for the languages in PSPACE. In this chapter , we describe how to extend Lutz's measure to the class PSPACE following the work of Mayordoma [Mayo94] .

2.2 Measure in PSPACE

Our definition of resource-bounded measure in Chapter 1 was restricted to classes of the form R(Γ),for Γ a measure resource-bound. In particular since each measure resource-bound contains p, for any resource-bound Γ we have E $\subseteq$ R(Γ).

In order to define a measure inside PSPACE we have to find a solution to the equation R(Γ) = PSPACE and check that the corresponding Γ-measure fulfils the Measure Conservation Theorem, that is, PSPACE does not have Γ-measure 0. This time we can not require that Γ is a measure resource-bound.

Let us look at the solution of the equation R(Γ)=DTIME(F) and R(Γ)=DSPACE(F) for differnt families F that we used in Chapter 1, e.g for F=E the solution is p.

By analogy ,the class of polylogarithmic space computable functions is the natural candidate to define a measure in PSPACE .

Lemma 2.1. (Mayordoma [Mayo94]) PSPACE $\subseteq$ R(polylogspace).

We see in the next theorem that R(polylogspace) corresponds to a class of self-reducible languages that is expected to be different from PSPACE.

Definition 2.1. A language A is PSPACE-wdq-self-reducible (where wdq stands for word-decreasing-queries) if A = L(M,A), where M is a PSPACE Oracle Turing Machinc that makes only queries strictly smaller than the input (in lexicographical order).

Balcazar has defined two types of self-reducibility, namely wdq-self-reducibility and ldq-self-reducibility, ldq standing for length decreasing queries. The most restrictive one is ldq-self-reducibility, where all the queries must be strictly shorter than the input. Notice that wdq-self-reducibility allows exponentially long decreasing chains to exist, while only linearly long chains can appear for the ldq type.

Theorem 2.1. (Mayordoma [Mayo94]) R(polylogspace) is exactly the class of PSPACE-wdq-self-reducible languages.

In [Balc] it is proven that E_2 has $\leq^p_m$-complete languages that are P-wdq-self-reducible.

Since every P-wdq-self-reducible language is clearly PSPACE-wdq-self-reducible, E_2 has $\leq^p_m$-complete languages that are PSPACE-wdq-self-reducible, and we have the following result.

Theorem 2.2. (Mayordoma [Mayo94]) If PSPACE = R(polylogspace) then E_2 = PSPACE.

And we know that E_2 = PSPACE is not likely to hold .

Now we consider only functions that are computable by on-line polylogspace machines

Our model of on-line machine is based on that of Hartmanis, Immerman and Mahaney in [HartImM].

Definition 2.2. An on-line Turing Machine is a machine that on input of length n

(a) starts with log n blank spaces marked on one of the working tapes,

(b) reads the input tape once from left to right, and

(c) writes the output from left to right on a write-only tape.

Definition 2.3. Let plogon be the class of functions that are computable by online machines with working and output space polylogarithmic in the size of the input. In this case and for constructor functions only, we do not bound the output space.

Theorem 2.3. (Mayordoma [Mayo94]) PSPACE = R(plogon).

Thus we can define a measure in PSPACE from plogon-measure.

2.3 Γ-additivity in PSPACE

Definition 2.4. A set X is a Γ-union of the Γ-measure 0 sets X_j , j $\in \mathrm{N}\bigcup\{0\}$ iff X=$\bigcup_{j=0}^{\infty}$ and there exists a Γ-computable 1-MS d such that , for every j,X_j $\subseteq S^{\infty}[d_j]$. (Notice that here Γ is not necessarily a measure resource-bound)

Lemma 2.2. (Mayordoma [Mayo94]) If X is a plogon-union of plogon-measure 0 sets , then X has plogon-measure 0.

We have studied PSPACE-wdq-self-reducibility in section 2.2, showing that there are languages out of PSPACE that are PSPACE-wdq-self-reducible, unless PSPACE = E_2 . We now look at a more restrictive form of wdq-self-reducibility, where the machine used has a linear bound on the space and a restriction on the order the queries are made.

Definition 2.5. A language A is LINSPACE-oq-self-reducible (where oq stands for ordered queries) if A = L(M,A), where M is a LINSPACE-oracle-machine that for each input makes the queries in lexicographical ascending order, and all of them are strictly smaller than the input (in lexicographical order).

Note that if A is LINSPFACE-wdq-self-reducible via a O(n)-truth-table LINSPACE-machine, then A is LINSPACE-oq-self-reducible, because we can order the queries before making them. In particular, most of the known selfreductions for natural problems in PSPACE are O(n)-tt (even O(1)-tt) and computable in LINSPACE.

Theorem 2.4. (Mayordoma [Mayo94]) The class of LINSPACE-oq-self-reducible languages has measure 0 in PSPACE.

Chapter 3
Quantive Structure of Exponential time

3.1 Introduction. In this chapter we summarize some devolopments concerning resource bounded measure of classes having certain nice properties like P-bi-immunity , Incompressibility etc , within E and E_2 , and using these it will be shown that "NPC is a small class in E " assuming that the Berman-Hartmanis isomorphism conjecture is true . Also we will show that , if (i) Berman-Hartmanis conjecture is true and (ii) for L_1 , $L_2 \in$ NPI , $L_1 \Delta L_2 \notin$ NPC , then either NP has p-measure 0 or NP is not p-measurable .

3.2 "Most is all"

To begin with we state a result , due to Regan , Sivakumar , and Cai [RSC95] which states that any "reasonable" complexity class that contains almost every element of E (respectively E_2) , then it contains every element of E(respectively E_2) .

Theorem 3.1 (Regan , Sivakumar , and Cai [RSC95]). Let C be a class of languages which is either closed under symmetric difference or closed under (finite) union and intersection

1. If $\mu_p(C \mid E) = 1$ then $E \subseteq C$.
2. If $\mu_p(C \mid E_2) = 1$ then $E_2 \subseteq C$.

As an example if NP has p-measure 1 in E then E $\subseteq$ NP , that is any language that can be recognized in exponential time can also be recognized by some non-deterministic polynomial time bounded turing machine . This would be a very interesting relationship between deterministic and non-deterministic time .

3.3 Incompressibility and Bi-immunity

First we introduce some basic concepts .

Definition 3.1. The collision set of a function f : $\{0,1\}^* \rightarrow \{0,1\}^*$ is

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

Note that if f is one to one then $C_f = \emptyset$.

Definition 3.2. A function f : $\{0,1\}^* \rightarrow \{0,1\}^*$ is said to be one-to-one almost everywhere if it's collision set C_f is finite .

Definition 3.3. Let A , B $\subseteq \{0,1\}^*$ and let t: N $\rightarrow$ N , A $\le_m^{DTIME(t)}$-reduction of A to B is a function f $\in$ DTIME(t) such that for all x $\in \{0,1\}^*$, x $\in$ A iff f(x) $\in$ B A $\le_m^{DTIME(t)}$-reduction of A is a function f that is a $\le_m^{DTIME(t)}$-reduction of A to f(A).

Definition 3.4 Let t : N → N be a time constructible function . A langauge A is said to be incompressible by $\leq_m^{DTIME(t)}$-reductions if every $\leq_m^{DTIME(t)}$-reductions of A is one to one almost everywhere . A langauge A is said to be incompressible by $\leq_m^p$-reductions if it is incompressible by $\leq_m^{DTIME(q)}$-reductions for all polynomial q .

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

Theorem 3.2 (Juedes and Lutz) [JL95a] . Let c ∈ Z^+ and define the sets

$$X = \{ A \subsetneq \{0,1\}^* \mid A \text{ is incompressible by } \leq_m^{DTIME(2^{cn}}) \}$$
$$Y = \{ A \subsetneq \{0,1\}^* \mid A \text{ is incompressible by } \leq_m^{DTIME(2^{n^c})} \}$$

Then $\mu_p(X)=\mu_{p_2}(Y)=1$.

Thus almost every language in E is incompressible by $\leq_m^{DTIME(2^{cn})}$-reductions, and almost every langauge in E_2 is incompressible by $\leq_m^{DTIME(2^{n^c})}$-reductions

Corollary. (Juedes and Lutz) [JL95a]. Almost every langauge in E and almost every language in E_2 is incompressible $\leq_m^p$-reductions .

It had been known that there is a language A ∈ E that is incompressible by $\leq_m^p$-reductions [Meyer77] . Thus the above corollary strengthens this result .

Next we recall the definition of P-immune languages .

Definition 3.5. A language A $\subsetneq$ $\{0,1\}^*$ is P-immune if no infinite subset of A is in P . A language A $\subsetneq$ $\{0,1\}^*$ is P-bi-immune if A and A^c are both P-immune.

Intuitively , a language that is P-immune "cannot be non-trivially approxiamated from inside or outside " by any language in P.

The following theorem relates incompressibility by $\leq_m^p$-reduction to P-bi-immunity .

Theorem 3.3 (Ko and Moore[KM75]). Every language that is incompressible by $\leq_m^p$-reductions is P-bi-immune.

The following results shows that almost every language in E is P-bi-immune.

Theorem 3.4 (Mayordomo [May94]) . The class of P-bi-immune language has p-measure 1 , and hence p_2-measure 1 .

Next we recall the notion of p-isomorphism and state the Berman-Hartmanis conjecture .

Two languages L_1 and L_2 are p-isomorphic if ∃ a function f : $\{0,1\}^* \rightarrow \{0,1\}^*$ such that

1. x ∈ L_1 ⇔ f(x) ∈ L_2 .

2. f is bijective .
3. Both f and f^{-1} are p-computable .

Berman-Hartmanis Conjecture . Any two NP-complete languages are p-isomorphic .

Proposition 3.1 If Berman-Hartmanis isomorphism conjecture holds then the class NPC has p-measure 0 .

Proof. We know that the language B = { $\alpha(A)©\alpha(b)$: all entries of A are in {0,1} and all entries of b are 1 , and Ax=b has a binary solution } is NP-complete . (See [Lepad] for a proof)

(Here $\alpha(A)$ the encoded form of matrix A and © is a special symbol which does not occur in the encoding of A and b .)

We define a subset C of B such that

C={ $\alpha(A)©\alpha(b)$: A= I_n , for some $n \geq 2$, I_n being the identity matrix of order n , for some $n \geq 2$ and all entries of b are 1. }

Clearly $C \subseteq B$ and C is infinite which is in P . Thus B is not P-immune as it contains an infinite subset which is in P.

Now take any other language L which is NP-complete . If Berman-Hartmanis isomorphism conjecture holds , then there exists a polynomial time computable function h : C → L which is 1-1 and onto and whose inverse is also polynomial time computable . So the set h(C)= { h(w) | w ∈ C } ⊆ L and is infinite .

Now x ∈ h(C) iff $h^{-1}(x) \in C$. Thus we see that h(C) ∈ P so that L is also not P-immune. Thus no NP-complete language is P-immune and hence not P-bi-Immune . Hence , by Theorem 3.4 , the class NPC has p-measure 0 in E . Hence the proof.

Remark. We also note that p_2-measure of NPC in E_2 is 0 assuming Berman-Hartmanis conjecture is true . To see this note that the class PBI= set of all P-bi-immune languages , has p_2-measure 1 in E_2 . Also no language in NPC is P-immune as seen in the proof of Proposition 3.1 .

So NPC $\subseteq E_2 \bigcap (PBI)^c$ and hence NPC has p_2-measure 0 in E_2 .

Proposition 3.2 . if

(i) $L_1 \in$ NPI , $L_2 \in$ NPI ⇒ $L_1 \Delta L_2 \notin$ NPC , and

(ii) Berman-Hartmanis conjecture holds , then either NP has p-measure 0 or it is not p-measurable .

Proof. Consider the class D = NPI ⋃ P . D is closed under symmetric difference since

L ∈ P , M ∈ P ⇒ LΔM ∈ P

L ∈ P , M ∈ NPI ⇒ LΔM ∈ NPI

L ∈ NPI , M ∈ NPI ⇒ LΔM ∉ NPC

Also we know that , if Berman-Hartmanis isomorphism conjecture is true , then NPC has p-measure 0 (Proposition 3.1) . Thus if NP has to have p-measure 1 , then NPI must have p-measure 1 , that is D has p-measure 1 . Then by

Theorem 3.1 we would have $E \subseteq D$ which is a contradiction to the fact that the language B considered in Propsition 3.1 is in NPC$\bigcap$E .

To see that $B \in$ NPC$\bigcap$E , first note that $B \in$ NPC [Lepad] . Next we show that $B \in E$.

(Recall B = { $\alpha(A)©\alpha(b)$: all entries of A are in {0,1} and all entries of b are 1 , and Ax=b has a binary solution })

For this ,note that given a string w , we can find out whether w is of the form $\alpha(A)©\ \alpha(b)$ in linear time , and , if it is so , then we can extract A and b from w in linear time . Now we have to check for atmost all $x \in \{0,1\}^n$, where n is the column dimension of the matrix A , whether x is a solution of Ax=b . Roughly speaking this can be done in $O(mn2^n)$ time , m being the row dimension of A , that is $O(2^{c|w|})$ time for some $c > 0$. Hence $B \in E$.

Thus NP can have p-measure 0 or NP is non-measurable . Hence the proof.

3.4 Complexity Cores

Now we look at the complexity cores of languages in E . Roughly speaking , a complexity core for a language L is a set of input strings for which every algorithm that recognizes L requires running time which is more than a specified bound .

Definition 3.6. Let t: N → N be a time bound and let A , $K \subsetneq \{0,1\}^*$. Then K is a DTIME(t(n))-complexity core of A , if , for every $c \in N$ and every machine that accepts A , the fast set F , defined as ,

F={ x | Number of steps used in the computation of $M(x) \leq c \cdot\ t(|x|) + c$ }

satisfies $|F \bigcap K| < \infty$.

Definition 3.7. Let A , $K \subsetneq \{0,1\}^*$

1. K is a polynomial complexity core of A if K is a DTIME(n^k)-complexity core of A for all $k \in N$.

2. K is a exponential complexity core of A if there is a real number $\epsilon > 0$ such that K is DTIME(2^{n^ϵ})-complexity core of A.

The following result , due to Juedes and Lutz , shows that almost every langauge in E (or E_2) are $\leq_m^P$-hard .

Theorem 3.5 (Juedes and Lutz [JL95a]) . If t: N → N is time constructible then every langauge that is incompressible by $\leq_m^{DTIME(t)}$-reductions has $\{0,1\}^*$ as a DTIME(t)-complexity core

Corollary. (Juedes and Lutz [JL95a]) Let $c \in Z^+$

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

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

Measure on Small Complexity Classes and application to BPP

4.1 Introduction .

Here , following Allender and Strauss [AlleSt94] , we present a notion of resource-bounded measure for P and other subexponential-time classes. This generalization is based on Lutz's notion of measure, but overcomes the limitations that cause Lutz's definitions to apply only to classes at least as large as E. We present many of the basic properties of this measure, and which are used to explore the class of sets that are hard for BPP. Bennett and Gill showed that almost all sets are hard for BPP with respect to Lebesgue measure . Lutz improved this from Lebesgue measure to measure on ESPACE. Using this , an improved result is obtained which shows that , for all $\epsilon > 0$, almost every set in E_ϵ is hard for BPP, where $E_\epsilon = \bigcup_{\delta<\epsilon} DTIME(2^{n^\delta})$. This is the best that can be achieved without showing that BPP is properly contained in E. A number of related results are also obtained in this way.

First we present the following important definition .

Definition 4.1. A density-system is a function $d_{i_1,\ldots,i_r} : \{0,1\}^* \to [0, \infty)$, where

1. The subscripts are natural numbers.
2. The argument is considered a partial characteristic sequence of a language.
3. $d.(w) = \frac{d.(w0)+d.(w1)}{2}$

A density-system is called an n-DS according to the number of subscripts. A 0-DS is a density function. A density function covers X if $X \subseteq \bigcup_{w|d(w)\geq 1} C_w$. A null cover of X is a 1-DS such that for all k, d_k covers X and $d_k(\lambda) \leq 2^{-k}$.

When faced with the task of defining measure on classes smaller than E, it is natural to try to modify Lutz's definitions, merely using smaller resource bounds C. For instance, to define a measure on P, one would consider density functions computed in DTIME($\log^{O(1)}$ n). This fails to work, because the usual convention for having sublinear-time Turing machines compute some function f is to have them recognize the language { x,i,b : bit i of f(x) is b} .

However , it is easy to see that the class of functions computed in this way by, for instance, polylog-time-bounded machines, is not closed under addition and subtraction. which seems to be necessary for many constructions. Similar problems arise when one uses the usual binary notation or scientific notation for the numeric values of the density functions. Our solution is to have the run time bound the length of the output, and to express numbers as the difference of two formal sums of powers of two, which allows us to perform the necessary arithmetic operations in the restricted time available. Using this representation for numeric values, one obtains a system that quite possibly does define a measure on P. Unfortunately , it seems quite difficult to verify that P itself does not have measure zero .

The central problem lies with an observation made previously that the binary sequences that are constructible in DTIME($\log^{O(1)}$ n) correspond not to sets in

P, but rather to "word-decreasing self-reducible" sets, some of which are hard for E [Ba]. This motivates the notion of limiting the dependency set size, which allows us to obtain subexponential bounds on the complexity.

4.2 Formal Definitions of the Measure .

The preceding paragraphs motivate some detailed definitions of a class of functions computed by sublinear-time Turing machines. In order for sublinear-time machines to perform interesting computation, we follow the usual convention of providing these machines with random access to their input; that is, the machines have an "address tape," and if i is written in binary on the address tape, then the machine can in unit time move its read/write head to bit position i of the input. Among other things, such machines can, in logarithmic time, compute the length of their input ([Bu]).

We adopt the convention that a machine computing a k-ary function is provided with k input tapes (with an "address tape" for each input tape). The running time of such a machine must be polylogarithmic in m, where m is the sum of the lengths of its k inputs. (Note that by choosing a suitable encoding, such a machine can be simulated by a machine with a single input tape.) The machines we consider will write their output on a write-only output tape; thus the output is restricted to be of length bounded by the running time.

Given a machine M and natural number n, define a dependency set $G_{M,n} \subseteq \{0 \ldots n\}$ to be a set such that for each $i \in G_{M,n}$ and each word w of length n, M can compute $M(w[0 \ldots i])$ querying only input bits in $G_{M,n} \bigcap \{0 \ldots n\}$. Note that for all M and n, there is a unique minimal dependency set for M and n, which can easily be computed by expanding the tree of queries that one obtains by assuming both possible values for each queried bit. In what follows, we let $G_{M,n}$ denote this minimal dependency set. Given a function f computed by machine M , we may use notation and speak of $G_{f,n}$ instead of $G_{M,n}$,

Note that our convention about paired inputs to M requires that if M computes a subscripted function $d_{k,r}$, G gets a cadre of subscripts matching d's: $G_{M,|w|,k,r}$ for $d_{k,r}(w)$: Often in practice $G_{M,n,k,r}$ is independent of k and r. In that case these subscripts will be suppressed.

Given a complexity class C of the form DTIME(F) (where we will always assume that F is a set of time-constructible functions such that $f(n) \in F$ $(f(n))^2 \in F$), let $\Delta(C)$ be the class of functions computed by Turing machines running in time f(log n) for some $f \in F$, and let $\Gamma(C)$ be the class of functions $d_{i_1 \ldots i_n}(w)$ computed by machines whose runtime and dependency set size are both bounded by functions of the form $f(\log(i_1 + \ldots + i_l + |w|))$ for some $f \in F$.

Note that if the functions in F are at least exponential, then $\Delta(C) = \Gamma(C$ If f is a function in $\Gamma(C)$, where f: $\{0,1\}^* \rightarrow \{0,1\}$, then f defines a constructor δ, where $\delta(w) = wf(w)$. A constructor specifies the sequence that is the limit as $j \rightarrow \infty$ of $\delta^j(\lambda)$. This gives rise to the class $R(\Gamma(C))$, which is the class whose characteristic sequence are given by some constructor in $\Gamma(C)$.

Theorem 4.1 (Allender , Strauss [AlleSt]) R(Γ(C))=C.

Thus we can think of defining Γ(C)-measure in C.

We now make precise the notion of a density system (DS) being easy to compute. A Γ(C)-computation of an n-DS $d_{i_1,\ldots,i_n}$ is an (n+1)-subscripted function $\hat{d}_{i_1,\ldots,i_n,r}$ satisfying

$\hat{d}_{i_1,\ldots,i_n,r}(w)$ is computable in Γ(C) , (numeric output is represented as a pair (a , b) representing the dyadic rational a-b , where each of a and b are represented as a formal sum of powers of 2) , and

$|\hat{d}_{i_1,\ldots,i_n,r}(w)- d_{i_1,\ldots,i_n}(w) \mid \leq 2^{-|w|}$.

For any complexity class C , we write X is Γ(C)-null if there is a Γ(C)-computation of a 1-DS $d_k(w)$ covering X such that $d_k(\lambda) \leq 2^{-k}$ for all k .

If a class X has nonzero measure , we can talk about another set Y having "measure 0 in X " and we write $\mu_{\Gamma(C)}(Y \mid X) = 0$, if $\mu_{\Gamma(C)}(Y \bigcap X) = 0$ or measure 1 if $\mu_{\Gamma(C)}(Y \bigcap X^c) = 0$.

A set X is a C-union of Γ(C)-null sets if $X = \bigcup_{j=0}^{\infty} X_j$ and there is a 2-DS $d_{j,k}$ so that $d_{j,k}$ covers X_j with value 2^{-k} and $d_{j,k}$ has a Γ(C)-computation .

Some useful results are :

Theorem 4.2. (Alleneder , Strauss [AlleSt]) For each L ∈ C the singleton set Γ(C)-null.

Theorem 4.3 (Alleneder , Strauss [AlleSt]) If X is a C-union of Γ(C)-null sets , then X is Γ(C)-null.

Theorem 4.4 (Alleneder , Strauss [AlleSt]) $\mu_{\Gamma(C)}(C) \neq 0$.

Thus , on C we can define a Γ(C)-measure which is nontrivial .

Note that one can also define a measure analogous to our measure on time-bounded classes, using space bounds, as opposed to time bounds. In this way, one can obtain a measure $\mu_{\Gamma(PSPACE)}$

4.3 Elementary Facts Concerning the Measure

Theorem 4.5. (Alleneder , Strauss [AlleSt]) Let C = DTIME(F). If f ∈ F , then $\mu_{\Gamma(C)}$(DTIME(f)=0.

That is DTIME(f) is a small class in DTIME(F) for any f ∈ F.

Corollary. (Alleneder , Strauss [AlleSt]) For all k, $\mu_{\Gamma(P)}(\mathrm{DTIME}(n^k)) = 0$.

Corollary. (Alleneder , Strauss [AlleSt]) Let $0 < \eta < \epsilon$, and E_ϵ denote $\bigcap_{\delta<\epsilon}$ DTIME(2^{n^δ}) . then $\mu_{\Gamma(E_\epsilon)}(E_\eta) = 0$.

Theorem 4.6. (Alleneder , Strauss [AlleSt]) Let C=DTIME(F) , where F contains no superexponential function . Let { X_j } be a collection of sets ,where X_j is covered by d_j for d ∈ Γ(C). Suppose m (a "modulus of convergence") is an increasing function of the form f(logn) for some f ∈ F and for all k we have

$$\sum_{j=m(k)}^{\infty} d_j(\lambda) < 2^{-k} .$$

Then

$$\mu_{\Gamma(C)}(\bigcap_{t=0}^{\infty}\bigcup_{j=t}^{\infty}X_j) = 0.$$

Theorem 4.7 (Alleneder , Strauss [AlleSt]) The set SPARSE is not Γ(P)-null in P.

That is the set SPARSE is not small in P , so that we can even think of a nontrivial measure in SPARSE⋂P .

4.4 Robustness, Alternative Formulations and Auxiliary Axioms

There are many choices that must be made in making the notion of a measure precise. The definitions in the preceding subsections reflect one set of choices, but it is instructive to consider other ways a definition could have been formulated, to see if the class of measure-zero sets varies under these changes. Juedes, Lutz and Mayordomo have previously shown that their notion of resource-bounded measure is robust in the face of many modifications of the definition of covers. As a practical matter, when trying to show that a class does not have measure zero in E or some larger complexity class, it is very useful to know that, in that setting, a null cover can be assumed without loss of generality to satisfy all of the following "niceness" conditions

A density system d_k is exactly computable if $d_k = d_{k,r}$.

A density function is conservative if it satisfies the following "conservation" property: $d(w) = \frac{d(w0)+d(w1)}{2}$.

If the density system d k is of the form $d_k = 2^{-k}d$ for some density function d then we say that d_k is derived from the martingale d condition that d_k be a null cover of

A set A is covered in the limit if there is a martingale d such that, for all w ∈ A, the sequence d(w[0. . .n]) has a limit of infinity, instead of merely an infinite lim sup.

A density system is regular if $d_k(z) = 1$ and $z \sqsubseteq w$ imply $d_k(w) = 1$.

When considering measure on subexponential complexity classes, there are additional choices involved in the definition, concerning how (or if) the length of the input is provided, questions concerning how dependency sets should be defined, etc , which raises the spectre that each of the 2^5 combinations of the niceness conditions listed above (not counting additional choices concerning providing the input length, etc.) would give rise to a different notion of measure.

It turns out that any null set can be covered by an exactly-computable martingale, but surprisingly, assuming any of the other niceness conditions is equivalent to assuming all of them.

That is, it can be shown that the notion of measure defined here is equivalent to the definition that results from having the measure-zero sets be covered in the limit by exactly-computable conservative martingales, where the machines that compute the martingales are even more limited than the machines that are considered here.

4.5 Hard Sets for BPP

It was shown in [BG] that for almost every A, $\mathrm{BPP}^A = \mathrm{P}^A$. Lutz showed that almost every set in ESPACE has this property, and thus , in particular , almost every such set is hard for BPP.

Theorem 4.8 (Alleneder , Strauss [AlleSt]) For almost every $A \in E_\epsilon$ we have $\mathrm{BPP} \subseteq \mathrm{P}^A$.

That is almost every language in E_ϵ is hard for BPP.

Theorem 4.9 (Alleneder , Strauss [AlleSt]) For almost every $A \in$ PSPACE we have $\mathrm{BPP} \subseteq \mathrm{P}^A$.

That is almost every language in PSPACE is hard for BPP.

Corollary . Let C be any of the classes E, EXP, PSPACE or E_ϵ : If $\mu_{\Gamma(C)}(\mathrm{NP} \mid C) \neq 0$ then $\mathrm{BPP} \subseteq \mathrm{P}^{NP}$.

We also see that almost every set A in E satisfies $\mathrm{BPP}^A = \mathrm{P}^A$.

Theorem 4.10 (Alleneder , Strauss [AlleSt]) For almost every $A \in E$ we have $\mathrm{BPP}^A = \mathrm{P}^A$.

Chapter 5: If NP is not small

5.1 Introduction

Many of the main open problems in Structural Complexity, such as whether the class NP coincides with one of the classes P or E_2 , are instances of a more general problem: the relationship between deterministic and nondeterministic time.

In this chapter we summarize some devolopments concerning the reasonableness and consequences of the hypothesis "NP does not have p-measure 0", which is a stronger hypothesis than that of the classical complexity theory hypothesis $P \neq NP$. Since , even if we assume that $P \neq NP$ most of the open questions in complexity theory still remain open , that is , we do not know how to use the fact $P \neq NP$ to solve other problems in this field . However , if we assume " NP does not have p-measure 0" we can settle many open problems as for example , we can prove the famous CvKL conjecture("Cook versus Karp-Lavin") to be true which is not known to follow from the fact that $P \neq NP$. Also we see that if we assume "NP does not have p-measure 0 " , then there is an NP search problem which does not reduce to the corresponding decision problem .

Let us review the various concepts relevant to this chapter .

We know that , if we can recognize a language A with an oracle B , then B is at least as hard as A , since an algorithm for B would produce an algorithm for A . This defines a partial preorder denoted as $\leq_T$ and called Turing reducibility .

We say that a language L $\leq^p_r$ reduces to a langauge A when L can be recognized in polynomial time using A as oracle and with the access restriction indicated by r . We say that a language A is $\leq^p_r$ -hard for a class C , when every language in C reduces to A and A is said to be $\leq^p_r$-complete if A is $\leq^p_r$-hard and $A \in C$. The most common polynomial time reducibilities are $\leq^p_T$, which is obtained when we do not give any restriction on oracle access , $\leq^p_m$ -completeness arises when we allow only one query per input and with the additional restriction that the input is accepted iff the query is in the oracle . A $\leq^p_{tt}$ B iff A $\leq^p_T$ via a machine which writes down all the queries to be made before the first word is queried .

$\leq^p_{q(n)-tt}$-reducibility is $\leq^p_{tt}$-reducibility where maximum number of queries allowed on an input of length of n is $q(n)$ $\leq^p_{q(n)-T}$-reducibility is $\leq^p_T$-reducibility where maximum number of queries allowed on an input of length of n is $q(n)$

The polynomial time truth-table reducibility is defined as follows . A $\leq^p_{btt}$ B if and only if A $\leq^p_{tt}$ B via a machine that on each input queries the oracle at most k times where k is a constant independent of the inputs . A $\leq^p_{ctt}$ B if and only if A $\leq^p_{tt}$ B via a machine that accepts the input exactly when all the queries have been answered positively by the oracle .

An oracle is positive iff $A \subseteq B \Rightarrow L(M,A) \subseteq L(M,B)$. The positive Turing reducibility is defined as follows A $\leq_{pos}$ B iff A = L(M,B) for some positive oracle machine .

A language A is said to be polynomial time nondeterministic Turing reducible to B (denoted as A $\leq^{NP}_T$ iff A=L(M,B) for some nondeterministic polynomial time oracle machine M .

For a language A we define P(A)= { L : L $\leq_T^p$ A } and NP(A) = { L : L $\leq_T^{NP}$ A } ,

Intuitively , Aadecision problem can be described as a set of valid inputs (graphs , numbers , mathematical expressions , etc) and a certain property that some of the inputs may satisfy (being connected , containing a hamiltonian path , being a prime , satisfying certain equations , etc). The problem consists in deciding whether a given valid input satisfies that property . But a search problem consists in finding a set of valid inputs for which given property may hold .

Finally , we recall

CvKL Conjecture ("Cook versus Karp-Levin"). There exists a language that is $\leq_T^p$-complete, but not $\leq_m^p$-complete, for NP.

5.2 If NP does not have p-measure 0

We note that P = NP $\Rightarrow$ ($\exists$c)NP $\subseteq$ DTIME(2^{cn}) $\Rightarrow$ μ_p(NP) = 0, μ_p(NP) = 0 $\Rightarrow$ μ_{p_2}(NP) = 0 $\Leftrightarrow$ μ(NP | E$_2$) = 0 $\Rightarrow$ μ(NP | E) = 0.

Lutz has conjectured that NP does not have measure 0 in E (denoted μ(NP | E) $\neq$ 0) and that NP does not have measure 0 in E$_2$ (denoted μ(NP | E$_2$) $\neq$ 0).

Now let us see the reasonableness of the hypothesis "NP does not have p-measure 0 " By the definition of p-measure, we know that NP has p-measure 0 if and only if there is a single martingale d $\in$ p that succeeds on every language A $\in$ NP. Since d $\in$ p, when betting on the condition "x $\in$ A" d requires only $2^{c|x|}$ time for some fixed constant c. On the other hand, for all k $\in$ N, there exist languages A $\in$ NP with the property that the apparent search space (space of witnesses) for each input x has $2^{|x|^k}$ elements. Since c is fixed, we have $2^{cn} \ll 2^{n^k}$ for large values of k. Such a martingale d would thus be a very remarkable algorithm! It would bet successfully on all NP languages, using far less than enough time to examine the search spaces of most such languages. It is reasonable to conjecture that no such martingale exists, i.e., that NP does not have p-measure 0.

Next we describe some consequences of the hypothesis that NP does not have p-measure 0.

Theorem 5.1. (Mayordoma [Mayo94]) If NP does not have p-measure 0 then NP contains a P-bi-immune set. If NP does not have measure 0 in E$_2$ then NP contains an E-bi-immune set.

Definition 5.1. An infinite set K $\subseteq$ {0,1}* is an exponential complexity core for a language A if there is a real number $\epsilon > 0$ such that for every machine M that accepts A there are at most finitely many x $\in$ K such that the time of machine M on input x is smaller than $2^{|x|^\epsilon}$.

(Intuitively, an exponential complexity core for a language L is a set of 'very infeasible' inputs for every algorithm that correctly recognizes L.)

Theorem 5.2. [JuedLu94a]. If NP does not have p-measure 0, then every $\leq_m^p$-complete language A for NP has a dense exponential complexity core.

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 giving no output) on all but finitely many inputs x ∈ K. (however it was known that hypothesis P $\neq$ NP implies the weaker conclusion that every $\leq_m^p$-complete language for NP has a nonsparse polynomial complexity core.)

The third consequence of $\mu_p(\mathrm{NP}) \neq 0$ to be mentioned here concerns the density of hard languages for NP. Let us consider the usual polynomial-time reducibilities ranging from $\leq_m^p$ to $\leq_T^p$. If $\leq_r^p$ is any of these reducibilities, all known $\leq_r^p$-hard languages for NP are dense. Efforts to explain this observation (and similar observations for other classes and reducibilities) have yielded many results. Berman and Hartmanis conjectured that no sparse language is $\leq_m^p$-hard for NP, unless P = NP. This conjecture was subsequently proven correct.

Theorem 5.3. [Maha]. If P $\neq$ NP, then no sparse language is $\leq_m^p$ -hard for NP. That is, P] $\neq$ NP $\Rightarrow$ NP $\not\subseteq \mathrm{P}_m$(SPARSE):

Theorem 5.3 was extended much later to truthtable reducibility with a bounded number of queries:

Theorem 5.4. (Ogihara and Watanabe [OgihWa]). If P $\neq$ NP, then no sparse language is $\leq_{btt}^p$-hard for NP. That is, P $\neq$ NP $\Rightarrow$ NP $\not\subseteq \mathrm{P}_{btt}$(SPARSE).

One is thus led to ask whether there is a reasonable hypothesis θ such that we can prove results of the form $\theta \Rightarrow$ NP $\not\subseteq \mathrm{P}_r(\mathrm{DENSE}^c)$, for various choices of the reducibility $\leq_r^p$. (Such a result is much stronger than the corresponding result $\theta \Rightarrow$ NP $\not\subseteq \mathrm{P}_r$(SPARSE). because there is an enormous gap between polynomial and 2^{n^ϵ} growth rates.)

Now we define

$$\mathrm{EE} = \bigcup_{c\in N} \bigcup_{n\in N} \mathrm{DTIME}(2^{2^{n+c}}) .$$
$$\mathrm{NEE} = \bigcup_{c\in N} \bigcup_{n\in N} \mathrm{NTIME}(2^{2^{n+c}}) .$$

Lemma 5.1. [Mayo94]

1. If NP contains a P-bi-immune language, then E $\neq$ NE and EE $\neq$ NEE. 2. If NP $\bigcap$ co-NP contains a P-bi-immune language, then E $\neq$ NE $\bigcap$ co-NE and EE $\neq$ NEE $\bigcap$ co-NEE.

Theorem 5.5

1. If NP does not have p-measure 0, then E $\neq$ NE and EE $\neq$ NEE. .

2. If NP $\bigcap$ co-NP does not have p-measure 0, then E $\neq$ NE $\bigcap$ co-NE and EE $\neq$ NEE $\bigcap$ co-NEE.

Proof . This follows immediately from Theorem 5.1 and Lemma 5.1.

Corollary . If NP does not have p-measure 0, then there is an NP search problem that does not reduce to the corresponding decision problem.

Proof . Bellare and Goldwasser [BellGo] have shown that, if EE $\neq$ NEE, then there is an NP search problem that does not reduce to the corresponding decision problem. The present corollary follows immediately from this and Theorem 5.5.

5.3 Separating completeness notions in NP

In this section we present main consequence of μ_p(NP) $\neq$0, that is:

Theorem 5.6. If NP does not have pmeasure 0, then there is a language C that is $\leq^p_{2-T}$-complete,but not $\leq^p_{2-tt}$-complete,for NP.

This theorem implies that if μ_p(NP) $\neq 0$ then CVKL conjecture holds.

5.4 Separating reducibilities in NP

In this section, assuming that NP is not small, we establish the distinctness of many polynomial-time reducibilities in NP , that is we will see completeness notion under different reducibility are different .

Theorem 5.7. (Selman[Selm82]) Assume that NP does not have p-measure 0. There exist A, B $\in$ NP$\bigcup$co-NP such that A$\leq^p_T$ B, but A$\not\leq^p_{pos-T}$

Proof. Selman [Selm82]

Similarly, we have the following.

Theorem 5.8. Assume that NP $\bigcap$ co-NP does not have p-measure 0. There exist A ,B $\in$ NP such that A $\leq^p_T$B but A $\not\leq^p_{pos-T}$B. There exist A ,B $\in$ NP such that A $\leq^p_{tt}$B but A $\not\leq^p_{pos-tt}$B.

Proof. See Selman[Selm82]

The rest of our results concern the separation of various polynomial-time truth-table reducibilities in NP, according to the number of queries.

Theorem 5.9. If NP does not have p-measure 0, then for all k $\in$ N there exist A,B $\in$ NP such that A$\leq^p_{k+1-tt}$ B but A$\leq^p_{k-tt}$ B.

Proof. See Mayordoma [Mayo94] .

Theorem 5.10. If NP does not have p-measure 0 and q,r: N $\rightarrow$ N are polynomial-time computable query-counting functions satisfying the conditions q(n) $\in$ o($\sqrt{(r(n))}$) and r(n) $\in$ O(n), then there exist A ,B $\in$ NP such that A $\leq^p_{r(n)-tt}$ but A $\not\leq^p_{q(n)-tt}$

Proof. See Mayordoma [Mayo94]

Theorem 5.11. If μ(NP | E$_2$) $\neq$ 0 and q is a polynomialtime computable querycounting function such that q(n) $\in$ O(log n), then there exist A, B $\in$ NP such that A$\leq^{p}_{q(n)+1-tt}$ but A$\not\leq^{p}_{q(n)-tt}$ r(n)), then there exist A,B $\in$ NP such

Proof. See Mayordoma [Mayo94]

Theorem 5.12. If μ(NP | E$_2$) $\neq 0$ and q,r :N $\to$ N are polynomial-time computable query-counting functions satisfying q(n) $\in o(\sqrt{r(n)})$,then there exists A,B $\in$ NP such that A$\leq^{p}_{r(n)-tt}$B but A$\not\leq^{p}_{q(n)-tt}$.

Proof. See Mayordoma [Mayo94]

5.5 Further results and open problems

Now we summarizes the consequences of the hypothesis "NP does not have measure 0 in PSPACE " known so far. Notice that if μ(NP | E$_2$) $\neq 0$ then μ(NP | PSPACE) $\neq 0$.

Theorem 5.13. If NP does not have measure 0 in PSPACE then

(i) NP contains a DLOG-bi-immune language.

(ii) NP contains a language that is not LINSPACE-oq-self reducible.

We have seen that for each of the treated questions, the hypothesis "NP does not have p-measure 0" gives the answer that seems most likely, relative to our current knowledge. Further investigation of this hypothesis and its power to resolve other questions is clearly indicated.

Now we review polynomial time hierarchy . The polynomial time hierarchy is the structure formed by the classes Σ^P_k , c and Δ^P_k for each k ≥ 0 , where

1. $\Sigma^P_0 = \Pi^P_0 = \Delta^P_0 = P$.
2. $\Sigma^P_{k+1} = NP(\Sigma^P_k)$ for k ≥ 0 .
3. $\Delta^P_{k+1} = P(\Delta^P_k)$ for k ≥ 0 .
4. $\Sigma^P_{k+1} = NP(\Sigma^P_k)$ for k ≥ 0 .

As noted in section 5.2, all known $\leq^p_T$-hard languages for NP are dense, i.e., our experience suggests that NP $\not\subseteq$ P(DENSE^c). This suggests two open questions. Karp and Lipton [KarpLi] have shown that

$$\Sigma^P_2 \neq \Pi^P_2 \Rightarrow \text{NP} \not\subseteq \text{P(SPARSE)}.$$

The first question posed by Selman , is whether the strong hypothesis μ($\Sigma^P_2 \backslash \Pi^P_2$ | E$_2$) $\neq 0$ can be used to combine these ideas to get a conclusion that NP $\not\subseteq$ P(SPARSE).

The second question is suggested by the first. A wellknown downward separation principle [Stoc77] says that , if the polynomial time hierarchy separates at some level , then it separates at all lower levels. Thus , for example , $\Sigma^P_2 \neq \Pi^P_2$ implies that P $\neq$ NP. Is there a "downward measure separation principle " stating that $\mu(\Sigma^P_{k+1} \backslash \Pi^P_{k+1}$ | E$_2$) $\neq 0 \Rightarrow \mu(\Sigma^P_k \backslash \Pi^P_k$ | E$_2$) $\neq 0$? In particular , does $\mu(\Sigma^P_2 \backslash \Pi^P_2$ | E$_2$) $\neq 0$ imply that μ(NP | E$_2$) $\neq 0$.

The next immediate open problem involves the further separation of completeness notions in NP. We have seen that the hypothesis μ_p(NP) $\neq 0$ separates $\leq^p_{2-T}$-completeness from $\leq^p_{2-tt}$-completeness in NP . However there is a large spectrum of completeness notations between $\leq_T^p$ and $\leq_m^p$. Watanabe([Wata87a] , [Wata87b]) and Buhrman , Homer , and Torenvliet [BuhrHoT] have shown that nearly all these completeness notions are distinct in E and in NE , respectively. In light of the results of sections 5.3 and 5.4 above , it is reasonable to conjecture that the hypothesis "NP does not have p-measure 0" yields a similarly detailed separation of completeness notions in NP. Investigation of this conjecture may shed new light on NP-completeness phenomena.

We finish by looking at the Berman-Hartmanis isomorphism conjecture formulated in 1977 , namely that all NP $\leq_m^p$-complete sets are polynomial time isomorphic [BermHa]. Most researchers now believe that the isomorphism conjecture as stated by Berman and Hartmanis is false. It would be very interesting to obtain results of the form "If NP does not have p-measure 0 then the isomorphism conjecture is false for $\leq_r^p$-complete sets" , for different reducibilities $\leq_r^p$.

Conclusion.

Resource bounded measure has been shown to interact in quantative ways with polynomial time reductions , bi-immunity , complexity cores , completeness and many other studied structural aspects of various complexity classes .

Resource-bounded measure is a powerful generalization of Lebesgue measure . There is reason to hope that it will be as fruitful in complexity theroy as Lebesgue measure has been in analysis and mathematical physics . Many investigators will have to ask and answer many question in order for resource bounded measure to achieve its full potential.

References

[AmboMay97] K. Ambos-Spies and E. Mayordomo. Resource-bounded measure and randomness. In A. Sorbi, editor, Complexity, Logic and Recursion Theory, Lecture Notes in Pure and Applied Mathematics, pages 1–47. Marcel Dekker, New York, N.Y., 1997.

[AlleSt94] E. Allender and M. Strauss. Measure on small complexity classes with applications for BPP. In Proceedings of the 35th Symposium on Foundations of Computer Science, pages 807–818, Piscataway, NJ, 1994. IEEE Computer Society Press.

[ArviKöm] V.Arvind , J.Köbler , M.Mundhenk: On Bounded Truth-Table Conjunctive and Randomised reductions to Sparse sets . Proceedings of the 12th Conference FSTTCS, Lecture Notes in Computer Science Vol.52 (1992) 140-151 .

[AS] E. Allender and M. Strauss. Towards a measure for P , DIMACS technical report , 1994 .

[BG] C. Bennet and J. Gill , Relative to random oracle P(A) $\neq$ NP(A) $\neq$ CO-NP(A) with probability 1 . SIAM J ,computation (1981) 96-113 .

[Bu] S.R Buss The Boolean formula value problem is in ALOGTIME , Prc 19th ACM symposium on thoery of computing 1987 , 123-131.

[Balc] J.L Balcazar: Self Reducibility . Journal of computer and Sysetm

Sciences 41(1990).

[BalcDig] J.L Balcazar , J .Diaz , J . Gabarro: Structural Complexity L EATCS Monograph on Computer Science , Vol 11, Springer-Verlag 1988.

[BellGo] M.Bellare , S.Goldwasser : The complexity of Decision Versus Search . SIAM Journal on Computing.

[BuhrHoT] H.Buhrman , S. Homer , L.Torenvliet : Completeness for Nondeterministic Complexity Classes . Mathematical Systems Theory 24 (1991) . 177-200.

[Bermha] L.Berman , J. Hartmanis : On Isomorphism and Density of NP and Other Complete sets . SIAM Journal on Computing (1977), 305-332.

[Gasaho] W.I Gasarch, S.Homer : Relativizations Comparing NP and Exponential Time , Information and Control (1983) , 88-100.

[HartImM] J.Hartmanis , N.Immerman , S.mahaney : One-way LOng-Tape Reductions .Proceedings of the 19th Symposium on Foundations of Computer Science (1978) , 65-72.

[JuedLu94a] D.W Juedes , J.H Lutz : The Complexity and Distribution of Hard Problems Proceedings of the 34th Symposium on Foundation of Computer Science (1993) , 177-185 SIAM Journal of Computing.

[Kautmi] S.Kautz , P.B Milltersen : Relative to random oracle NP is not small .Complexity Theory Retrospective :Springer-Verlag 1990, 108-146.

[Lepad] "Elements of the theory of computation " by Lewis and Papadimitriu

[Lutz90] J.H Lutz : Category and Measure in Complexity Classes. SIAM Journal of computing 1990 , 1100-1131.

[Lutz97] 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.

[Maha] S.R Mahaney : Sparse complete sets for NP . Journal of Computer and System Sciences 1982 , 130-143.

[Mayo94] E. Mayordomo. Contributions to the study of resource-bounded measure. PhD thesis, Universitat Politecnica de Catalunya, 1994.

[Ogihwa] M.Ogihara, O.Watanabe : On Polynomial Bounded Truth-Table Reducibility of NP Sets to Sparse Sets . SIAM Journal on Computing (1991), 471-483.

[Stoc77] L.J StockMeyer : The polynomial time Hierarchy , Theoretical Computer science (1977) 1-22.

[Wata87a] O.Watanabe : On Structure of Intractable Complexity Classes . PhD Dissertation

[Wata87b] O.Watanabe : A Comparison of Polynomial Time Completeness Notions . Theoritical Computer Science (1987) , 249-265.