Compressibility and Resource Bounded Measure

Harry Buhrman*[1] and Luc Longpré**[2]

[1] CWI. PO Box 94079, 1090 GB Amsterdam,
The Netherlands. E-mail: buhrman@cwi.nl
[2] Computer Science Department, University of Texas at El Paso,
El Paso TX 79968, USA. E-mail: longpre@cs.utep.edu.

Abstract. We give a new definition of resource bounded measure based on compressibility of infinite binary strings. We prove that the new definition is equivalent to the one commonly used. This new characterization offers us a different way to look at resource bounded measure, shedding more light on the meaning of measure zero results and providing one more tool to prove such results.
The main contribution of the paper is the new definition and the proofs leading to the equivalence result. We then show how this new characterization can be used to prove that the class of linear auto-reducible sets has p-measure 0. We also prove that the class of sets that are truth-table reducible to a p-selective set has p-measure 0 and that the class of sets that Turing reduce to a sub-polynomial dense set has p-measure 0. This strengthens various results.

1 Introduction

While Lebesgue measure has been used in mathematics since the previous century, and while it has been used earlier this century to study randomness in infinite strings [ML66], its notable appearance in complexity theory is in the formalization of "random oracles". Bennett and Gill [BG81] showed for example that relative to a random oracle A, $\mathrm{P}^A \neq \mathrm{NP}^A$. The statement about the random oracle is formalized as follows: the class of sets A such that $\mathrm{P}^A = \mathrm{NP}^A$ has Lebesgue measure zero.

When dealing with uncountable classes, measure zero is an intuitively appealing concept to formalize the idea that sets with a certain property are rare. But the concept seems to fall apart when dealing with countable classes, since all of these have Lebesgue measure zero. For example, how would one formalize the statement "most recursive oracles separate P from NP"?

Formalizing this kind of statement, Lutz introduced resource bounded measure [Lut90]. A more useful definition based on resource bounded martingales appeared in [Lut92]. With resource bounded measure, one is able to state formally results of the type "most languages in class $\mathcal{C}$ have property P". These

* Partially suported by the Dutch foundation for scientific research (NWO) through NFI Project ALADDIN, under contract number NF 62-376.
** Supported in part by NSF Grant CCR-9211174 and NSF Grant INT-9123551.

notions turned out to be quite useful in complexity theory, as witnessed by a stream of results in the last 3 years [JL95a, JL95b, Lut94, LM94b, LM94a, ASNT94, ASTZ94, AS94a]. In her thesis, Mayordomo gave a rather complete coverage of the topic [May94b].

In this paper, we offer a different definition of resource bounded measure equivalent to that of Lutz in [Lut92]. We say that a set is $t(n)$-compressible if its characteristic sequence can be compressed and uncompressed in time $t(n)$. The precise definition is given in section 4. A class of sets has p-measure zero if all the sets in the class are n^k-compressible for some fixed k. This new characterization has several advantages.

- This new characterization may allow more intuitive proofs of results about resource bounded measure.
- A result claiming that a class of sets does not have p-measure zero is usually seen as an abundance result. How should this abundance be interpreted? The new characterization explains in a precise way what is meant: the sets in the class cannot all be compressed with a fixed polynomial time bound.
- While the martingale characterization was not directly applicable to classes below E, Mayordomo [May94b] gave a definition applicable to PSPACE and Allender and Strauss [AS94a] gave a definition applicable to P and other subexponential classes. Although similar technical problems arise with our definition, it may offer other alternatives to define measure applicable to subexponential classes.

A corollary of the proof of equivalence is as follows: a class $\mathcal{C}$ has p-measure zero in E if and only if all the sets in $\mathcal{C}$ are n^k-compressible, for some fixed k. The n^k-compressible sets form a proper hierarchy, and E is not included in any fixed level of that hierarchy. Moreover, if we define $\mathrm{Comp}(n^k)$ as the class of sets in E that are n^k-compressible, then $E = \cup_k \mathrm{Comp}(n^k)$. So the meaning of abundance can be interpreted as follows: a class of sets X has p-measure 0 in E if and only if $X \cap \mathrm{E}$ is included in a fixed level of that hierarchy, while the hierarchy is itself infinite.

It should be noted that equivalence between a classical constructive measure and a definition based on Kolmogorov complexity has been studied in the context of random sequences. Martin-Löf's definition of random sequences [ML66] based on constructive measure is equivalent to a subsequent definition using incompressibility in the sense of some version of prefix Kolmogorov complexity due to Levin [Lev73]. See also Schnorr [Sch73] for a similar theorem. We refer the reader to the book by Li and Vitányi [LV93] for a more detailed account.

There is also a way to define compressibility in the non-uniform context and one can prove that plain Lebesgue measure zero is equivalent to that kind of compressibility (Kreinovich, personal communication).

Next we will use the new characterization to prove the following results:

- The class of $c \cdot n$ auto-reducible sets has p-measure 0.
- The class of sets that are truth-table reducible to a p-selective set has p-measure 0. It follows immediately that E does not have a truth-table hard p-selective set.

- The class of sets that are Turing reducible to a set with subpolynomial density has p-measure 0. This strengthens the results in [BH95].

2 Preliminaries

Let $\Sigma = \{0,1\}$. Strings are elements of Σ^*, and are denoted by lower case letters $x, y, u, v, \ldots$. Infinite strings are elements of Σ^∞, and are denoted by lower case Greek letters. The empty string is λ. For any string x, the length of a string is denoted by $|x|$, $x[i..j]$ is the substring of x from index i to index j inclusively, $x[i]$ stands for $x[i..i]$ and if $j < i$, then $x[i..j]$ is λ. For two strings x and y, $x \sqsubseteq y$ if y is an extension of x. Subsets of Σ^* are denoted by capital letters $A, B, C, S, \cdots$. The set $\Sigma^* - A$ is denoted by $\overline{A}$. The complement of a class of sets X is $X^c = \{A \subseteq \Sigma^* | A \notin X\}$. For a set A we use $A^{=n}(A^{\leq n})$ to denote the subset of A consisting of all strings of length $n(\leq n)$. For any set A the cardinality of A is denoted by $\|A\|$. We define C_w, the cylinder generated by w, as the class of languages $\{x \in \Sigma^\infty | w \sqsubseteq x\}$. We fix a pairing function $\lambda xy.\langle x,y\rangle$ computable in polynomial time from $\Sigma^* \times \Sigma^*$ to Σ^*. Without loss of generality we assume the pairing function respects the length of its arguments (i.e. $|x|+|y| \leq |\langle x,y\rangle| \leq 2(|x|+|y|)$.) We assume that the reader is familiar with the standard Turing machine model.

3 Resource bounded measure

We use here the definition of resource bounded measure based on martingales.

Let $D = \{m2^{-n} | m, n \in N\}$ be the set of nonnegative dyadic rationals.

Definition 3.1 *A* martingale *is a function* $d : \Sigma^* \to D$ *with the property that, for all* $w \in \Sigma^*$,

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

Definition 3.2 *A martingale* succeeds *on language* $A \subseteq \Sigma^*$ *if*

$$\limsup_{n\to\infty} d(\chi_A[0..n-1]) = \infty.$$

A martingale d is p-computable, and we call it a p-martingale, if $d(w)$ can be computed in time polynomial in $|w|$.

The intuition behind this definition is a game where a player is trying to predict the next bit by looking at all the bits that have been revealed so far. The player starts with an initial capital $d(\lambda)$ and can decide to bet an amount of money which is at most the current capital. If the predicted bit is correct, the capital increases by the amount bet, and if it's incorrect, the capital decreases by that same amount. The function d models the current capital of the betting strategy after having seen a finite binary string.

Lutz [Lut92] defined a martingale to give a real value and required computations of the martingale to approximate the real value by using dyadic rationals. But as shown indepedently by Mayordomo [May94b] and by Juedes and

Lutz [JL95b], defining the martingale directly with the dyadic rationals provides an equivalent definition of resource bounded measure. Moreover, without loss of generality, we may assume that $d(w)$ is a dyadic number $m2^{-n}$ such that $n < |w|$.

Definition 3.3 *A class X of languages has p-measure 0, and we write $\mu_p(X) = 0$, if there is a p-martingale d that succeeds on every element of X.*

Definition 3.4 *A class X of languages has p-measure 1, and we write $\mu_p(X) = 1$, if $\mu_p(X^c) = 0$.*

4 Compressibility

Compressibility of finite strings is usually defined using Kolmogorov complexity. There are various problems in defining compressibility of infinite strings in terms of the Kolmogorov compressibility of its prefixes, mainly because no string has all of its prefixes completely incompressible. We define compressibility of an infinite string by using another infinite string that can generate it, with the compressibility calculated as how much of the prefix of the compressed string is needed to reproduce a prefix of the uncompressed string. For the time bounded version, we also need that the compressed string can be computed, at least in some weak sense.

Everything in this paper is in terms of p-measure, which is appropriate for studying abundance with respect to E. The obvious extension to p_2, the class of functions of the form $2^{\log^c n}$ and EXP also holds, and we are still investigating the equivalence for other classes of functions.

Definition 4.1 *An infinite string $\omega \in \{0,1\}^\infty$ is f-compressible if $\exists \kappa \in \{0,1\}^\infty$ such that the following conditions hold.*

1. *(Decompression) There is a Turing machine M that, given $\kappa[0..j]$, outputs a prefix $\omega[0..i]$ of ω in time at most $f(i+j)$, and such that the value $i-j$ is not bounded by any constant.*
2. *(Compression) There is a Turing machine M' that, given $\omega[0..i]$, uses at most $f(i)$ time to output a finite number of strings, one of which is a prefix $\kappa[0..j']$ such that M, on input $\kappa[0..j']$, outputs a prefix of ω that is a proper extension of $\omega[0..i]$.*

5 Measure zero implies compressibility

We have defined measure zero using basic martingales. We will need a few more properties of the martingales to prove that measure zero implies compressibility. The following shows that such special properties can be assumed without loss of generality.

Lemma 5.1 *If $\mu_p(C) = 0$, then there exists a p-martingale d that succeeds on every element of C and satisfies:*

1. $d(\lambda) = 1$,
2. $d(x)$ *is a dyadic number* $m2^{-n}$ *such that* $n \leq |x| + 4$,
3. $(\forall x, y)\ d(x)/4 \leq d(xy)$, *where* y *is a finite binary string,*
4. $(\forall x, b)\ d(xb) \leq (7/4)d(x)$, *where* b *is a bit.*

Proof. Let d be a martingale witnessing that $\mu_p(\mathcal{C}) = 0$. Define a martingale d' in the following way. Given a finite string x, let $i_0 = -1$ and for $k \geq 1$, let i_k be the smallest integer, if it exists, such that

$$\frac{d(x[0..i_k])}{d(x[0..i_{k-1}])} \geq 2.$$

Define

$$\begin{aligned} d'(\lambda) &= 1 \\ d'(x) &= d'(x[0..i_k]) - 1/4 + (1/4)d(x)/d(x[0..i_k]), \\ &\qquad \text{where } k \text{ is the largest integer such that} \\ &\qquad x[0..i_k] \text{ is a proper prefix of } x \end{aligned}$$

Informally, d' starts with \$1.00 on λ, and its betting strategy on successively longer prefixes of the characteristic function of a set A is to keep part of its capital frozen and use the rest to bet in proportion with the d strategy. The amount of frozen capital is revised each time d has doubled its capital. At that time, all the capital is frozen, except for \$0.25 which is kept for betting. Each time d is doubled, d' earns \$0.25. If d is successful on A, the doubling occurs infinitely often, so d' is unbounded as well, thus successful.

Although d' does not meet Property 2 of the lemma, it meets stronger versions of Properties 3 and 4:

- $(\forall x, y)\ (3/5)d(x) \leq d(xy)$, where y is a finite binary string,
- $(\forall x, b)\ d(xb) \leq (7/5)d(x)$, where b is a bit.

To see this, first notice that the function $f(k) = d(x[0..i_k])$ is monotonically increasing, so $d(x[0..i_{k_1}]) \geq 1$. Let k_1 be the largest integer such that $xy[0..i_{k_1}]$ is a (not necessarily proper) prefix of xy, where indices i_k are defined as above. Similarly, let k_2 be the largest integer such that $x[0..i_{k_2}]$ is a prefix of x. Notice that $d'(xy[0..i_{k_2}]) \leq d'(xy[0..i_{k_1}])$ because $i_{k_2} \leq i_{k_1}$ and by monotonicity. Also notice that $d(x)/d(x[0..i_{k_2}]) < 2$, because if it were ≥ 2, then k_2 would not be largest by our definition of indices i_k.

$$\begin{aligned} d'(xy) &= d'(xy[0..i_{k_1}]) - 1/4 + (1/4)d(xy)/d(xy[0..i_{k_1}]) \\ d'(x) &= d'(xy[0..i_{k_2}]) - 1/4 + (1/4)d(x)/d(xy[0..i_{k_2}]) \\ d'(xy) &= d'(xy[0..i_{k_1}]) - 1/4 + (1/4)d(xy)/d(xy[0..i_{k_1}]) + \\ &\quad (3/5)d'(x) - (3/5)d'(xy[0..i_{k_2}]) + 3/20 - (3/20)d(x)/d(x[0..i_{k_2}]) \\ &\geq d'(xy[0..i_{k_2}]) - 1/4 + (1/4)d(xy)/d(xy[0..i_{k_1}]) + \\ &\quad (3/5)d'(x) - (3/5)d'(xy[0..i_{k_2}]) + 3/20 - (3/20)d(x)/d(x[0..i_{k_2}]) \\ &= (2/5)d'(xy[0..i_{k_2}]) - 1/4 + (1/4)d(xy)/d(xy[0..i_{k_1}]) + \end{aligned}$$

$$\begin{aligned}
&(3/5)d'(x) + 3/20 - (3/20)d(x)/d(x[0..i_{k_2}]) \\
&\geq 2/5 - 1/4 + 0 + (3/5)d'(x) + 3/20 - (3/20) * 2 \\
&= (3/5)d'(x).
\end{aligned}$$

The other property can be proved in a similar fashion:

$$d'(xb) = d'(xb[0..i_{k_1}]) - 1/4 + d(xb)/4d(x[0..i_{k_1}])$$

Case 1: $xb[0..i_{k_1}] = x$

$$\begin{aligned}
d'(xb) &= d'(x) - 1/4 + d(xb)/4d(x) \\
&\leq d'(x) - 1/4 + 1/2 \\
&= (5/4)d'(x) + (1 - d'(x))/4 \\
&\leq (5/4)d'(x) \leq (7/5)d'(x)
\end{aligned}$$

Case 2: otherwise

$$\begin{aligned}
d'(x) &= d'(xb[0..i_{k_1}]) - 1/4 + d(x)/4d(x[0..i_{k_1}]) \\
d'(xb) &= (7/5)d'(x) + d'(xb[0..i_{k_1}]) - (7/5)d'(xb[0..i_{k_1}]) - 1/4 + 7/20 + \\
&\quad d(xb)/4d(x[0..i_{k_1}]) - (7/20)d(x)/d(x[0..i_{k_1}]) \\
&\leq (7/5)d'(x) - (2/5)d'(xb[0..i_{k_1}]) + 1/10 + 2d(x)/4d(x[0..i_{k_1}]) - \\
&\quad (7/20)d(x)/d(x[0..i_{k_1}]) \\
&\leq (7/5)d'(x) - (2/5) + 1/10 + (3/20)d(x)/d(x[0..i_{k_1}]) \\
&\leq (7/5)d'(x) - (2/5) + 1/10 + 3/10 \\
&= (7/5)d'(x)
\end{aligned}$$

Now, define martingale d'' as follows. Define $d''(\lambda) = 1$, $d''(xb) = d''(x) + \text{round}((d'(xb) - d'(x\tilde{b}))/2)$, where $\tilde{b}$ is the bit b flipped, and $\text{round}((d'(xb) - d'(x\tilde{b}))/2)$ is $(d'(xb) - d'(x\tilde{b}))$ approximated towards 0 as the (possibly negative) dyadic number of the form $m2^{-n}$ where $n \leq |x| + 4$. The loss of capital for d'' compared to d' is at most $2^{-|x|-4}$ when betting on x, which totals to at most $\Sigma_{i=1}^{\infty} 1/2^{i+4} = 1/16$ cumulatively. Also, because $d'(\lambda) = 1$, for all y, $d'(y) \geq 3/5$. So, for all y, $d''(y) \geq d'(y) - 1/16 \geq d'(y) - (1/16)(5/3)d'(y) = (43/48)d'(y)$. Similarly, $d''(y) \leq (53/48)d'(y)$. We now have

$$d''(x)/4 \leq (\frac{53}{4 * 48})d'(x) \leq \frac{53 * 5}{4 * 48 * 3}d'(xy) \leq \frac{53 * 5 * 48}{4 * 48 * 3 * 43}d''(xy) < d''(xy).$$

Similarly,

$$d''(xb) \leq \frac{7 * 53 * 48}{5 * 48 * 43}d''(x) < (7/4)d''(x),$$

so we get all the properties of the lemma.

Theorem 5.2 $\mu_p(C) = 0 \Rightarrow (\exists f \in p)(\forall A \in C)$ χ_A *is* f*-compressible.*

Proof. Assume that $\mu_p(\mathcal{C}) = 0$, and let d be a p-martingale as in Lemma 5.1. Let $A \in \mathcal{C}$ and let $\omega = \chi_A$. We need an infinite string κ to encode ω. We can interpret ω as the encoding of a real in the interval $(0,1)$. In a standard encoding, all infinite strings starting with 0 encode reals in the first half of the interval and strings starting with 1 encode reals in the right half of the interval. We will create an encoding scheme that possibly moves this half-way border. If $d(0) > d(1)$, then the binary expansion of reals in an interval larger than $1/2$ will be used to encode reals between 0 and 0.5. Similarly, if $d(x0) > d(x1)$, then the size of the interval reserved to encode reals that extend $x0$ will be larger than that of extensions of $x1$. The idea is to keep an interval of reals to encode extensions such that the size of the interval is proportionally related to the current capital with the current betting strategy of the martingale. Intuitively, large intervals can be described with few bits, so a winning strategy will result in compression.

More formally, let g be a function from finite binary strings to intervals in $(0,1)$ defined as follows:

$$\begin{aligned} g(\lambda) &= (0,1) \\ g(x0) &= \text{the left part of } g(x) \text{ of size } d(x0)/2^{|x0|} \\ g(x1) &= g(x) - g(x0) \end{aligned}$$

The following lemma has a straightforward proof by induction on the length of the strings. The proof is omitted.

Lemma 5.3 $|g(x)| = d(x)/2^{|x|}$.

Since $g(xb)$ is a subinterval of $g(x)$ and since part 4 of Lemma 5.1 ensures that $\lim_{n\to\infty} |g(\omega[0..n])| = 0$, an infinite sequence can be associated with the real r defined by $r = \lim_{n\to\infty} g(\omega[0..n])$. Let κ be the binary expansion of r. We now have to show that κ is a valid compression of ω.

To generate $\omega[0..i]$ from $\kappa[0..j]$, simulate the martingale starting at λ on successively longer strings. Suppose we have generated the string x so far. If $g(x0)$ contains $C_{\kappa[0..j]}$, then append 0 to x. If $g(x1)$ contains $C_{\kappa[0..j]}$, then append 1 to x. Continue until $C_{\kappa[0..j]}$ is not contained in either of the intervals. Since $g(\omega[0..i])$ is an interval containing $C_{\kappa[0..j]}$, κ has to encode an extension of $\omega[0..i]$.

To generate $\kappa[0..j]$ from $\omega[0..i]$, we compute the list of all possible strings that encode extensions of $\omega[0..i]0$ and of $\omega[0..i]1$. Let $I_0 = g(\omega[0..i]0)$ and $I_1 = g(\omega[0..i]1)$. Since $|I_0|$ is $d(\omega[0..i]0)/2^{i+2}$ and $d(\omega[0..i]0)$ is a dyadic number $m2^{-n}$ where $n \leq i+6$, the borders of I_0 can be expressed by dyadic numbers $m2^{-n}$ where $n \leq 2i+8$. The interval I_0 can be covered exactly by a set of at most $2(2i+8)$ intervals. This is done by starting from a point inside the interval which is expressible by a dyadic number $m2^{-n}$ where n is the smallest, covering the part of I_0 on the left of that point with a set of at most $2i+8$ intervals and covering the right part of I_0 with another set of at most $2i+8$ intervals. Interval I_1 can be covered similarly. Output all of the intervals. Strings associated with intervals covering I_0 will extend $\omega[0..i]$ with a 0, and those associated with intervals covering I_1 will extend $\omega[0..i]$ with a 1.

It remains to show that $i-j$ is not bounded by any constant, where i is the index of the last bit produced by the uncompression algorithm M on $\kappa[0..j]$. Select

j such that M on $\kappa[0..j]$ produces $\omega[0..i]$, and such that M on $\kappa[0..j-1]$ produces a proper prefix of $\omega[0..i]$. Since M produces $\omega[0..i]$ on $\kappa[0..j]$, $g(\omega[0..i])$ includes $C_{\kappa[0..j]}$. Since M produces a proper prefix of $\omega[0..i]$ on $\kappa[0..j-1]$, $g(\omega[0..i])$ does not contain $C_{\kappa[0..j-1]}$. Interval $g(\omega[0..i])$ can be partitioned into 3 intervals: L, $\kappa[0..j]$ and R where L (resp. R) corresponds to the reals in $g(\omega[0..i])$ that are smaller (resp. greater) than those in $\kappa[0..j]$. Assume that $\kappa[j] = 0$. The case where $\kappa[j] = 1$ is similar. Then, R is an interval included in $\kappa[0..j-1]1$ because $g(\omega[0..i])$ does not include $\kappa[0..j-1]$. So, $|R| \leq |\kappa[0..j]|$. We also have the following upperbound for the size of L:

$$\begin{aligned}
|L| &\leq |g(\omega[0..i]0)| \\
&= d(\omega[0..i]0)/2^{i+2} \\
&\leq (7/4)d(\omega[0..i])/2^{i+2} \\
&\leq 7d(\omega[0..i]1)/2^{i+2} \\
&= 7|g(w[0..i]1)| \\
&\leq 7(|C_{\kappa[0..j]}| + |R|).
\end{aligned}$$

$$\begin{aligned}
|g(\omega[0..i])| &= |L| + |C_{\kappa[0..j]}| + |R| \\
&\leq 8(|C_{\kappa[0..j]}| + |R|) \\
&\leq 16|C_{\kappa[0..j]}|.
\end{aligned}$$

But $|g(\omega[0..i])| = d(\omega[0..i])/2^{i+1}$ and $|C_{\kappa[0..j]}| = 1/2^{j+1}$, so we get

$$\begin{aligned}
d(\omega[0..i])/2^{i+1} &\leq 16/2^{j+1} \\
d(\omega[0..i]) &\leq 2^{i-j+4} \\
\log(d(\omega[0..i])) - 4 &\leq i - j
\end{aligned}$$

Since the martingale is successful, $d(\omega[0..i])$ is unbounded and then so is $i - j$.

6 Compressibility implies measure zero

Theorem 6.1 $(\exists f \in p)(\forall A \in \mathcal{C})$ χ_A *is* f*-compressible* $\Rightarrow \mu_p(\mathcal{C}) = 0$.

Proof. Let f be computable in time n^k. For each $A \in \mathcal{C}$, we build below a martingale that succeeds on A and is n^k time computable. Since n^k time bounded martingales are easily enumerable for a fixed k, $\mathcal{C}$ is a p-union of p-measure 0 sets, so $\mu_p(\mathcal{C}) = 0$. (To build a p-union of p-measure 0 sets, we need to build a sequence of martingales $d_1, d_2, ...$ witnessing that the sets have p-measure 0, and a uniform algorithm that receives k and x as input and computes $d_k(x)$ in time polynomial in both k and x. See [Lut92] for more details and a proof of this.)

Let $A \in \mathcal{C}$ be a set such that χ_A is f-compressible. Assume we have algorithms to compress $\omega = \chi_A$. Let's say algorithm B_1 computes κ from ω and algorithm B_2 computes ω from κ. We build a p-martingale. Suppose we are given $\omega[0..i]$. For each $j < i$, simulate B_1 on $\omega[0..j]$. For increasing j, look at the strings output by B_1 and make sure they extend the previous ones. Look at the strings from the

simulation of B_1 on $\omega[0..j]$ that remain. Eliminate the strings that do not extend $\omega[0..j]$. Make the remaining strings minimal by simulating B_2 on their prefixes and taking the smallest prefix that does extend $\omega[0..j]$. Separate the remaining strings into two groups, those that predict a 0 and those that predict a 1. Call these groups G_0 and G_1. Let each string vote for the next bid. The relative weight of each vote depends on the length of the string: the shorter the string, the more weight. Let $s_i = \Sigma_{x_k \in G_0 \cup G_1} 2^{-|x_k|}$. Let $b_0 = \Sigma_{x_k \in G_0} 2^{-|x_k|}$. Let $b_1 = \Sigma_{x_k \in G_1} 2^{-|x_k|}$. Then, $d(\omega[0..i]0) = d(\omega[0..i])(1 + (b_0 - b_1)/s_i)$ and $d(\omega[0..i]1) = d(\omega[0..i])(1 + (b_1 - b_0)/s_i)$.

We claim that for any i, $d(\omega[0..i]) \geq \sum_k 2^{i-|x_k|}$. This can be proved by induction on i. The statement is true for $i = 0$, assuming initial capital of 1. Assume it's true up to i. At step $i+1$, wlog, suppose $\omega[i+1] = 0$. Suppose for the sake of analysis that each string is extended with all possible suffixes up to length $i+c$ for some appropriate constant c. Since the vote was weighted, extending the strings and giving each string an equal vote is equivalent. Suppose a ratio r of strings voted 0. Then the capital becomes $d(\omega[0..i+1]) = 2rd(\omega[0..i]) \geq 2r2^i s_i$ by induction hypothesis. At the same time, all the strings that voted for 1 got removed from the set, and the strings that voted for 0 get appended by both 0 and 1, doubling their number and increasing their length by one. Some of these strings may get removed by the B_1 algorithm. So, $s_{i+1} \leq rs_i$. So, we still have $d(\omega[0..i+1]) \geq 2^{i+1} s_{i+1}$.

Now, let k be an arbitrary value, and let j such that algorithm B_2 computes $\omega[0..i]$ from $\kappa[0..j]$ such that $i-j > k$. By the assumption on compressibility, such a j must exist. Algorithm B_1 on $\omega[0..i-1]$ will produce $\kappa[0..j]$ as one of the x_k. By the claim above, $d(\omega[0..i-1]) \geq \sum_k 2^{i-1-|x_k|} \geq 2^{i-1-|\kappa[0..j]|} \geq 2^{i-j-2} \geq 2^{k-2}$. This shows that the capital is unbounded, so the martingale succeeds.

7 Applications of the new characterization

In this section we give some examples of how the new characterization can be applied.

Theorem 7.1 *[May94a] The class of non-P-bi-immune sets has p-measure 0.*

Proof. Let A be a non-P-bi-immune set and suppose there is a infinite set $C \subseteq A$ where C is in P. (The case where $C \subseteq \overline{A}$ is analogous.) The compressed characteristic sequence of A is the concatenation of the bits $\chi_A(x)$ with $x \notin C$. To uncompress, given $\kappa[0..j]$, one simply re-computes the characteristic sequence of A, filling in the places where $x \in C$, until $j+1$ elements of $\overline{C}$ have been encountered, or until $2j$ elements of the sequence have been generated. The value $i-j$ is either j, which is unbounded, or it is the number of elements of C we have generated, so if C is infinite, $i-j$ is also unbounded. To compress $\chi_A[0..i]$, just remove the bits corresponding to elements of C. Then create two strings by adding a 0 and a 1. One of the two strings will uncompress into the correct extension of $\chi_A[0..i]$.

Next we will investigate the class of sets that are auto-reducible [AS94b]. A set A is auto-reducible if there is a polynomial time oracle Turing machine that accepts A with A as oracle provided that on input x, it never queries x to the oracle.

Theorem 7.2 *For any fixed constant c, the class of sets that are auto-reducible via oracle machines that query no more than $c \cdot |x|$ queries on input x has p-measure 0.*

Proof. Let $n_0 = 4$, and let $n_{i+1} = n_i^{\log n_i}$. For i large enough, an auto-reduction on 0^{n_i} will never query a string of size $\geq n_{i+1}$. To compress χ_A, substitute the bit for 0^{n_i} by a sequence of bits corresponding to the answers to all queries of strings y for $y > 0^{n_i}$ in the lexicographic ordering. This results in a local expansion of χ_A. Then, remove from χ_A all the bits corresponding to those large queries. Overall, we removed the bit for 0^{n_i}, and moved some other bits around in χ_A. This results in one bit of compression for each section of χ_A corresponding to strings between 0^{n_i} and $0^{n_{i+1}}$. Since this is done for each i, the number of bits of compression is unbounded. To uncompress, simulate the auto-reduction machine on strings of the form 0^{n_i} for successive i. This allows reordering the bits of χ_A and generating the missing bits. To compress, it is impossible to determine the answer to the queries. Instead, compute all the computations corresponding to possible answers to queries. Each computation provides a candidate compressed string. Since there are at most cn_i queries, there are at most 2^{cn_i} possible computations, so we have enough time to generate them all.

The next theorem deals with p-selective sets, introduced by Selman [Sel79]. A set A is p-selective if there is a polynomial time selector function f such that: 1) $f(x,y) \in \{x,y\}$ and 2) if $x \in A$ or $y \in A$ then $f(x,y) \in A$. Intuitively $f(x,y)$ hands back the most likely of x or y to be in A. We will use the following lemma:

Lemma 7.3 *[BvHT93] Let A be a p-selective set. Any finite set X can be ordered, using the p-selector, as follows: $\{x_1, \ldots, x_k\} = X$. Such that $x_i \in A$ implies $x_j \in A$ for all $j \geq i$.*

Theorem 7.4 *The class of sets that truth-table reduce to a p-selective set has p-measure 0.*

Proof. We show here that the p-selective sets do not have p-measure 0. It is not hard to generalize this proof to the class of sets that truth-table reduce to a p-selective set. Details will be in the final paper. Let A be a p-selective set. We will see how to compress $\chi_A[2^n - 1 \ldots 2^{n+1} - 2]$ (corresponding to strings of length n) into $n+1$ bits. The lemma provides us with an ordering of the strings of length n. The ordering can be computed in time polynomial in $\|X\| * \max\{|x| \in X\}$. This also implies that there are at most $2^n + 1$ settings of these strings consistent with A. The compression will be an index among those $2^n + 1$ consistent settings. Then it is easy to see that by generating all the $2^n + 1$ many of these codings, we have that χ_A is f-compressible for some $f \in p$.

Corollary 7.5 *There is no truth-table hard p-selective set for* E.

Proof. E does not have p-measure 0 [Lut94].

The next result shows that the class of sets that reduce by Turing reductions to a set that has sub-polynomial density has p-measure 0. A function is sub-polynomial if $\forall\epsilon\exists n_0\forall n > n_0 : f(n) < n^\epsilon$.

Theorem 7.6 *Let f be a sub-polynomial function. The class of sets that Turing reduce to a set S with density f (ie $\|S^{\leq n}\| \leq f(n)$) has p-measure 0.*

Proof. Let $A \leq_T^p$ to a set S with density $f(n)$ via machine M_T in time $p(n)$. Again we have to show that the characteristic sequence of A can be compressed and uncompressed. Consider $w = \chi_A[2^n - 1...2^{n+1} - 2]$, corresponding to strings of size n. We will compress the first n bits of w. Each string x of length n is mapped by M_T to at most $p(n)$ many different queries, provided that the answers to these queries are known. We replace the first n bits of w by $f(p(n))$ pointers in these n computations of M_T, meaning that the string queried is in S. Since there are only n computations of size $p(n)$, a pointer can be coded in $c \log n$ bits, for some c. There are only $f(p(n))$ pointers, so the number of bits is $cf(p(n)) \log n < n$ bits, achieving the wanted compression. To uncompress, simulate M_T on all those n strings of length n, maintaining a table for which strings are in S according to the compressed string and the computation so far. To compress, generate all possible codes. This results in $2^{cf(p(n))\log n}$ strings, keeping only the ones that are still consistent with χ_A so far. After the last bit of the n strings is known we pick the smallest among these still consistent strings and set that to be the prefix of κ.

Corollary 7.7 *[BH95] There is no Turing hard set for* E *with sub-polynomial density.*

Acknowledgements

We would like to thank Paul Vitányi for suggesting us to look at a Kolmogorov type of characterization of resource bounded measure. We would like to thank Martin Strauss for useful discussions that led us to improved results and David Martin and Jeroen van Maanen for useful comments. We also would like to thank A.M. Bobu for hosting a meeting that enabled us to initiate this research.

References

[AS94a] E. Allender and M. Strauss. Measure on small complexity classes, with applications for BPP. In *Proc. 35th IEEE Symposium on Foundations of Computer Science*, pages 807–818. IEEE Computer Society Press, 1994.

[AS94b] K. Ambos-Spies. p-mitotic sets. In *Logic and Machines, Lecture Notes in Computer Science*, volume 177, pages 1–23. Springer-Verlag, 1994.

[ASNT94] K. Ambos-Spies, C. Neis, and S. A. Terwijn. Genericity and measure for exponential time. In *Proceedings of the 19th Symposium on Mathematical Foundations of Computer Science*, pages 221–232. Springer-Verlag, 1994. To appear in Theoretical Computer Science.

[ASTZ94] K. Ambos-Spies, S.A. Terwijn, and X. Zheng. Genericity and measure for exponential time. In *Proc. ISAAC'94, Lecture Notes in Computer Science*, volume 834, pages 369–377. Springer-Verlag, 1994. To appear in Theoretical Computer Science.

[BG81] C. Bennett and J. Gill. Relative to a random oracle A, $P^A \neq NP^A \neq$ Co-NP^A with probability 1. *SIAM J. Comput.*, 10(1):96–113, February 1981.

[BH95] H. Buhrman and M. Hermo. On the sparse set conjecture for sets with low density. In Ernst W. Mayr and Claude Puech, editors, *STACS 95*, volume 900 of *Lecture Notes in Computer Science*, pages 609–618, Berlin, 1995. Springer-Verlag.

[BvHT93] H. Buhrman, P. van Helden, and L. Torenvliet. P-selective self-reducible sets: A new characterization of P. In *Proc. Structure in Complexity Theory eighth annual conference*, pages 44–51. IEEE Computer Society Press, 1993.

[JL95a] D. W. Juedes and J. H. Lutz. The complexity and distribution of hard problems. *SIAM Journal on Computing*, 24:279–295, 1995.

[JL95b] D. W. Juedes and J. H. Lutz. Weak compleness in E and E_2. *Theoretical Computer Science*, 143:149–158, 1995.

[Lev73] L. Levin. On the notion of a random sequence. *Soviet Math. Dokl.*, 14:1413–1416, 1973.

[LM94a] J. Lutz and E. Mayordomo. Cook versus Karp-Levin: Separating completeness notions if NP is not small. In *STACS 1994, Lectures Notes in Computer Science*, pages 415–426. Springer-Verlag, 1994.

[LM94b] J. Lutz and E. Mayordomo. Measure, stochasticity, and the density of hard languages. *SIAM Journal on Computing*, 23:762–779, 1994.

[Lut90] J. Lutz. Category and measure in complexity classes. *Siam J. Computing*, 19(6):1100–1131, December 1990.

[Lut92] J. Lutz. Almost everywhere high nonuniform complexity. *J. Computer and System Sciences*, 44:220–258, 1992.

[Lut94] J. Lutz. Weakly hard problems. In *Proc. Structure in Complexity Theory ninth annual conference*, pages 146–161. IEEE Computer Society Press, 1994. To appear in SIAM J. on Computing.

[LV93] M. Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Texts and Monographs in Computer Science. Springer-Verlag, 1993.

[May94a] E. Mayordomo. Almost every set in exponential time is p-bi-immune. *Theoretical Computer Science*, 136:487–506, 1994.

[May94b] E. Mayordomo. *Contributions to the Study of Resource-Bounded Measure*. PhD thesis, Universitat Politècnica de Catalunya, 1994.

[ML66] P. Martin-Löf. The definition of random sequences. *Information and Control*, 9:602–619, 1966.

[Sch73] C. Schnorr. Process complexity and effective random tests. *J. Comput. System Sci.*, 7:376–388, 1973.

[Sel79] A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Math. Systems Theory*, 13:55–65, 1979.