Resource Bounded Randomness and Its Applications

D. M. Stull

1 Introduction

Martin-Löf gave the first meaningful definition of a *random* infinite binary sequence [51]. From a classical standpoint, such a notion seems meaningless, as every sequence has probability zero. Martin-Löf's insight was to use computability theory to give an *effective* version of measure theory. With this effectivization, a sequence A is (Martin-Löf) random if the singleton $\{A\}$ is not of effective measure zero. The field of algorithmic randomness has since grown to include many different notions of randomness, all making essential use of computability [18, 58].

With the prominence of complexity theory in computation, a natural step is to impose resource bounds on the computation in algorithmic randomness. Resource bounded randomness studies the different notions of what it means for a sequence to be "random" relative to a *resource bounded* observer. Schnorr gave the first definition of resource bounded randomness by imposing time bounds on martingales [65]. However, this work did not have an immediate impact. Instead, resource bounded randomness was largely unexplored until Lutz's, independent, development of *resource bounded measure* [46], a resource bounded effectivization of Lebesgue measure theory. As noted by Ambos-Spies, et al. [5], resource bounded measure implicitly redefines Schnorr's notion of resource bounded randomness. Lutz showed that resource bounded measure, and therefore resource bounded randomness, is a valuable tool in complexity theory. This application ignited interest in resource bounded randomness, resulting in significant growth in the area.

Nevertheless, many fundamental problems remain to be explored. One of the great achievements in algorithmic randomness, in the computable setting, is the variety of characterizations of its principal definitions. There are three prominent viewpoints of randomness: statistical tests (Martin-Löf [51]), martingales (Schnorr [65]) and compressibility (Levin [44] and Chaitin [15]). The richness of algorithmic randomness is, in part, due to the fact that most notions can be defined using each paradigm. Unfortunately, this has not yet extended to resource bounded randomness. Thus far, only the martingale approach of Schnorr and Lutz has cemented itself as fundamental. An important goal of resource bounded randomness is to be able to pass freely between the different

viewpoints. This is valuable for several reasons. The first is that having varied characterizations is indicative of a concept being fundamental. Due to its relative youth, it is not yet clear (with the exception of the notion of Schnorr and Lutz) which definitions of resource bounded randomness are fundamental. The second is that, while being formally equivalent, the three viewpoints are psychologically very different. Thinking in terms of martingales brings clarity to certain problems, while null covers are better suited for others. Resource bounded randomness would benefit by having this richness of viewpoints.

Another direction is to discover the resource bounded analogs of the most common notions of randomness in the computable setting. For example, what is the resource bounded version of Martin-Löf randomness or of Schnorr randomness? Part of the solution to this problem relates to the ability to define randomness in a variety of ways, as described above. However, more is needed for a satisfactory answer to this problem. Specifically, it will require showing that theorems characterizing a notion of randomness in the computable setting have clear resource bounded analogs. Interestingly, recent theorems characterizing randomness using results from classical analysis provide one of the most promising avenues for the solution to this problem (see Section 5.1.2).

The purpose of this paper is to give an overview of resource bounded randomness. We will introduce several notions, and pay particular attention to their relations with one another. We will also highlight several recent applications of resource bounded randomness to analysis and number theory. While the most fruitful applications of resource bounded randomness has been to complexity theory, the recent applications to other areas of mathematics suggest promising avenues for future research.

2 Preliminaries

Unless otherwise stated, we will be using the binary alphabet $\Sigma = \{0, 1\}$. We will use the standard lexicographical ordering of Σ^*, $s_0 = \lambda$, $s_1 = 0, \ldots$. A *language* $L \subseteq \Sigma^*$ is a set of finite strings. The *characteristic sequence* of a language L, $\chi_L \in \mathbf{C}$, is the infinite binary sequence

$$\chi_L = [s_1 \in L][s_2 \in L][s_3 \in L] \ldots,$$

where

$$[s_i \in L] = \begin{cases} 0 & \text{if } s_i \in L \\ 1 & \text{if } s_i \notin L. \end{cases}$$

We will commonly omit the notation χ_L, and consider L as both a set of strings and as an infinite binary sequence. For a language $S \subseteq \Sigma^*$, denote the subset of $\mathbf{C}$ consisting of all cylinders generated by S by

$$[S] = \bigcup_{w \in S} C_w.$$

Let M be a Turing machine which halts on all inputs. The *time complexity of* M is the function $f : \mathbb{N} \to \mathbb{N}$, where $f(n)$ is the maximum number of steps that M uses on any input of length n. The *space complexity of* M is the function $f : \mathbb{N} \to \mathbb{N}$, where $f(n)$ is the maximum number of tape cells that M scans on any input of length n. For this survey, we will always include the output tape in determining the space complexity of a Turing machine. Let $f : \mathbb{N} \to \mathbb{N}$ be a function. A language L is *decidable in time (space)* $f(n)$ if there is a TM M with time (resp. space) complexity $f(n)$ deciding L. The *time complexity class* $\mathsf{TIME}(\mathsf{f}(\mathsf{n}))$ is the set of languages

$$\mathsf{TIME}(\mathsf{f}(\mathsf{n})) = \{L \subseteq \Sigma^* \mid L \text{ is decidable by an } O(f(n)) \text{ time TM}\}.$$

Similarly, the *space complexity class* $\mathsf{SPACE}(\mathsf{f}(\mathsf{n}))$ is the set of languages

$$\mathsf{SPACE}(\mathsf{f}(\mathsf{n})) = \{L \subseteq \Sigma^* \mid L \text{ is decidable by an } O(f(n)) \text{ space TM}\}.$$

We will primarily be interested in the following classes:

$$\begin{aligned} \mathsf{P} &= \cup_k \mathsf{TIME}(\mathsf{n}^\mathsf{k}) & \mathsf{PSPACE} &= \cup_k \mathsf{SPACE}(\mathsf{n}^\mathsf{k}) \\ \mathsf{E} &= \cup_k \mathsf{TIME}(2^{\mathsf{kn}}) & \mathsf{ESPACE} &= \cup_k \mathsf{SPACE}(2^{\mathsf{kn}}) \\ \mathsf{EXP} &= \cup_k \mathsf{TIME}(2^{\mathsf{n}^\mathsf{k}}) & \mathsf{EXPSPACE} &= \cup_k \mathsf{SPACE}(2^{\mathsf{n}^\mathsf{k}}) \end{aligned}$$

A function $g : \Sigma^* \to \Sigma^*$ is *computable in time (space)* $f(n)$ if there is a TM M computing g with time (resp. space) complexity $f(n)$. The *time complexity function class* $\mathsf{FTIME}(\mathsf{f}(\mathsf{n}))$ is the set of functions

$$\mathsf{FTIME}(\mathsf{f}(\mathsf{n})) = \{g : \Sigma^* \to \Sigma^* \mid g \text{ is computable by an } O(f(n)) \text{ time TM}\}.$$

Similarly, the *space complexity function class* $\mathsf{FSPACE}(\mathsf{f}(\mathsf{n}))$ is the set of functions

$$\mathsf{FSPACE}(\mathsf{f}(\mathsf{n})) = \{g : \Sigma^* \to \Sigma^* \mid g \text{ is computable by an } O(f(n)) \text{ space TM}\}.$$

We will primarily be interested in the following function classes:

$$\begin{aligned} \mathsf{FP} &= \cup_k \mathsf{FTIME}(\mathsf{n}^\mathsf{k}) \\ \mathsf{FPSPACE} &= \cup_k \mathsf{FSPACE}(\mathsf{n}^\mathsf{k}) \end{aligned}$$

We can extend these definitions to discrete sets in the natural way. The exception is $\mathbb{N}$, which, unless otherwise stated, will always be encoded using the unary alphabet 0^*[1]. Let $t : \mathbb{N} \to \mathbb{N}$. A set of functions $\{f_n\}$, $f_n : \Sigma^* \to \mathbb{Q}$, is a *$t(n)$-time (-space) uniformly computable* if there is a $t(n)$-time (-space) computable function $f : \mathbb{N} \times \Sigma^* \to \mathbb{Q}$ such that $f(n, w) = f_n(w)$ for every $n \in \mathbb{N}$ and $w \in \Sigma^*$.

[1] For the purposes of this survey unary encodings of $\mathbb{N}$ are often essential. We will typically use natural numbers as a precision parameter, specifying the degree of accuracy needed of some computation.

3 Resource Bounded Randomness

In this survey, we will discuss several notions of resource bounded randomness. We list the polynomial time randomness notions in the table below. Each of these notions have analogous concepts for polynomial space randomness. As we will see, in the polynomial space setting, the martingale, open cover and compressibility definitions are equivalent. Unfortunately, in the polynomial time setting, this is unknown.

Martingale	$\mathsf{RAND_p}$	$\mathsf{RAND_{S\text{-}p}}$	$\mathsf{RAND_{K\text{-}p}}$	$\mathsf{RAND_{m\text{-}p\text{-}S}}$	$\mathsf{RAND_{m\text{-}BP\text{-}p}}$	$\mathsf{dim_p^{=1}}$	$\mathsf{Dim_p^{=1}}$
Open Cover				$\mathsf{RAND_{W\text{-}p}}$	$\mathsf{RAND_{BP\text{-}p}}$		
Compressibility				$\mathsf{RAND_{Kol\text{-}p\text{-}S}}$	$\mathsf{RAND_{Kol\text{-}BP\text{-}p}}$		

3.1 Using Martingales

We begin with the first, and most widely studied, notion of resource bounded randomness, which is based on martingales. Recall that a *martingale*[2] is a function $d : \{0,1\}^* \to [0,\infty)$ satisfying

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

for every finite string $w \in \{0,1\}^*$. A martingale can be thought of as a strategy for betting on successive bits of an infinite binary sequence. The quantity $d(w)$ is, then, the amount of "money" the martingale has after betting on the first $|w|$ bits of the sequence with prefix w. The martingale condition (eq. (1)) ensures that the payoffs are fair. We say that a martingale d *succeeds* on an infinite binary sequence A if d makes an unbounded amount of money; formally d succeeds if

$$\limsup_{n\to\infty} d(A\upharpoonright n) = \infty.$$

The *success set* of a martingale d is the set

$$S^\infty(d) = \{A \in \mathbf{C} \mid d \text{ succeeds on } A\}.$$

Schnorr initiated the, now fundamental, role of martingales in algorithmic randomness [65, 63]. Schnorr also realized that martingales are naturally suited to resource bounded computation, and was the first to study resource bounded randomness [65]. Unfortunately, the resource bounded aspect of his work was not systematically pursued.

[2]Martingales were first studied by Ville, who used them to characterize Lebesgue measure zero sets [70]. Bienvenu, Shafer and Shen [9] have recently published a nice survey on the history of martingales in algorithmic randomness.

Instead, resource bounded randomness was largely unexplored until Lutz's development of *resource bounded measure* (see Section 4). Lutz, independently of Schnorr, showed that resource bounded martingales[3] give an effective notion of Lebesgue measure theorycwhich could be used to study complexity theory [46]. As noted by Ambos-Spies, et al., [5], Lutz's resource bounded measure implicitly redefines Schnorr's notion of resource bounded randomness. We now give a basic introduction to Schnorr and Lutz's definition of resource bounded randomness.

For any function $t : \mathbb{N} \to \mathbb{N}$[4], we say that a martingale d is *computable in time (resp. space)* $t(n)$ if there is a $t(n)$-time (resp. space) computable function $f : \Sigma^* \times \mathbb{N} \to \mathbb{Q}$ such that

$$|d(w) - f(w, r)| \leq 2^{-r}.$$

Definition 3.1. For a function $t : \mathbb{N} \to \mathbb{N}$, we say that an infinite sequence A is *$t(n)$-time (-space) random* if no $t(n)$-time (-space) martingale succeeds on A. We denote the set of $t(n)$-time and -space random sequences by $\mathsf{RAND}_{\mathsf{t(n)}}$ and $\mathsf{RAND}_{\mathsf{t(n)-space}}$, respectively.

While specific time and space bounds provide a fine grain definition of randomness, we are also interested in sequences which are random with respect to classes of functions. The two most prominent, and the primary focus of this survey, are the classes of polynomial time and polynomial space functions.

Definition 3.2. An infinite sequence A is *polynomial time (space) random* if no n^k-time (resp. space) martingale succeeds on A, for any $k \in \mathbb{N}$. We denote the set of polynomial time and polynomial space random sequences by $\mathsf{RAND}_{\mathsf{p}}$ and $\mathsf{RAND}_{\mathsf{pspace}}$, respectively.

Our first example, due to Schnorr, is on the law of large numbers. This proposition, although easy to prove, provides simple proofs of separations of many randomness notions in this survey.

Proposition 3.3 ([65])**.** *If A is n^2-time random, then A satisfies the law of large numbers. That is, the limiting frequency of 0's in A exists, and is equal to $\frac{1}{2}$.*

Note that, since $t(n)$-space randomness implies $t(n)$-time randomness, n^2-space random sequences also satisfy the law of large numbers.

Our second example shows that being decidable in 2^{kn} time is not a property of n^{k+1}-time random sequences[5]. For a proof, see, e.g., [2].

[3] Lutz initially used *density functions* to define resource bounded measure theory. Resource bounded measure can be equivalently defined using martingales, which is now the most common presentation. The notation denoting martingales by d is a result of this fact.

[4] In this survey, we will only consider resource bounds t which are *time constructible*. That is, functions t such that $t(n) \geq n$ and the function $f : \Sigma^* \to \mathbb{N}$, defined by $f(w) = t(|w|)$, is computable.

[5] Note that the exponential difference arises because the running time of the martingale is based on a finite prefix $w \sqsubseteq A$. The next bit on which the martingale bets, however, corresponds to a finite string of length $\lfloor \log |w| \rfloor$.

Proposition 3.4. *If $A \in \mathsf{TIME}(2^{kn})$, then A is not n^{k+1}-random.*

As a consequence, we have the following:

Corollary 3.5. *If $A \in \mathsf{E}$, then A is not polynomial time random*[6].

Defining randomness using real valued martingales greatly simplifies many proofs. However, it is sometimes preferable for the martingales to take rational values. A martingale $d : \Sigma^* \to [0, \infty)$ is *exactly computable* in time (space) $t(n)$ if d is computable in time $O(t(n))$ and $d(w) \in \mathbb{Q}$, for every $w \in \Sigma^*$. A useful lemma for time (and space) randomness is that the definition remains unchanged if we only consider exactly computable martingales. The proof of this can be found in, e.g., [2].

Lemma 3.6. *For every $t(n)$-time (-space) computable martingale d, there is a martingale d' which is exactly computable in time (resp. space) $t(n)$ such that $S^\infty(d) = S^\infty(d')$.*

It is often the case that defining a single martingale which "bets" on all of the conditions we care about becomes too technical. In this case, we may simplify the analysis by defining a sequence of martingales, each betting on a single condition. As long as the sequence is uniformly computable, there is a single martingale, computable in almost the same time, which succeeds on the union of the component success sets. The proof of this can be found in, e.g., [2].

Lemma 3.7. *If a sequence of $t(n)$-time computable martingales $\{d_n\}$ is $t(n)$-time uniformly computable, then there is a $(nt(n))$-time computable martingale d such that*

$$S^\infty(d) = \cup_n S^\infty(d_n).$$

A martingale d is *universal* for a notion of randomness if d succeeds on every nonrandom sequence. The existence of a universal martingale is a nice property enjoyed by several concepts in algorithmic randomness. An immediate question is whether the resource bounded randomness notions we've been discussing have universal martingales. It is well known that there is a $n^{k+1}k \log n$-time function $f : \Sigma^* \times \mathbb{N} \to \mathbb{Q}$, which is universal in the sense that

$$f(w, n) = f_n(w),$$

where $f_1, f_2, \ldots$ is an enumeration of all n^k-time computable functions. We can apply Lemmas 3.4 and 3.7 and the existence of this universal n^k-time function to show the existence of a weak notion of universal $t(n)$-time (and space) bounded martingale. A proof of this can be found in [47].

Lemma 3.8. *For every $k \in \mathbb{N}$, there is an $l \in \mathbb{N}$ and an n^l-time computable martingale d succeeding on every sequence A which is n^k-time nonrandom.*

[6] Recall that we identify a language with its characteristic sequence as defined in the preliminaries.

We can use Lemma 3.8 to show the existence of n^k random sets in E by diagonalizing against the universal martingale.

Lemma 3.9. *For every k, there is an $l \in \mathbb{N}$ and $A \in \mathsf{TIME}(2^{ln})$ that is n^k-time random.*

Informally, the proof of this lemma is a diagonalization argument over the universal martingale d of Lemma 3.8. We construct a sequence A inductively. Having specified a finite prefix w, we choose the next bit to be 0 if and only if $d(w0) \leq d(w1)$. Since we are always choosing the bit which minimizes the martingales winnings, we can show that d does not succeed on A. It is not hard to verify that the sequence A is decidable in E. A rigorous proof can be found in [2].

As a corollary to Lemma 3.9, we see that there is no universal polynomial time martingale.

Corollary 3.10. *There is no polynomial time computable martingale which is universal over the class of polynomial time computable martingales.*

3.1.1 Schnorr and Kurtz Randomness

One of the strengths of the martingale approach to algorithmic randomness is that it suggests investigating the rate of success of martingales. Indeed, Schnorr's principal issue [63] with computable (and Martin-Löf) randomness was that a sequence is considered nonrandom simply because a martingale succeeded on it. However, this gives no information on how quickly the martingale succeeds. Schnorr suggested that a sequence A should be considered nonrandom if a computable martingale succeeded on A, *and* we could compute (infinitely often), the number of bits the martingale must see until it increases its capital by a certain amount. This approach led to the conception of the Schnorr and Kurtz randomness notions. This idea can be naturally extended to the resource bounded setting. In this section, we describe two different ways to define the resource bounded analogs of Schnorr and Kurtz randomness.

We begin with the analogs of Schnorr and Kurtz randomness defined by Wang [74]. Recall that an *order* is an unbounded, nondecreasing function $f : \mathbb{N} \to \mathbb{N}$.

Definition 3.11 ([74])**.** An infinite sequence A *fails* a *$t(n)$-time (-space) Schnorr test* if there is a $t(n)$-time (-space) martingale d and $t(n)$-time (-space) computable order f such that

$$d(A\restriction n) \geq f(n)$$

infinitely often. A is *polynomial time (space) Schnorr random* if A passes every n^k-time (-space) Schnorr test. We denote the set of polynomial time and polynomial space Schnorr random sequences by $\mathsf{RAND}_{\mathsf{S\text{-}p}}$ and $\mathsf{RAND}_{\mathsf{S\text{-}pspace}}$, respectively.

Wang's definition of resource bounded Kurtz randomness is based on a martingale characterization of Kurtz randomness first proved by Wang [73]

Definition 3.12 ([74])**.** An infinite sequence A *fails* a $t(n)$*-time (-space) Kurtz test* if there is a $t(n)$-time (-space) martingale d and $t(n)$-time (-space) computable order f such that

$$d(A\restriction n) \geq f(n)$$

for almost every n. A is *polynomial time (space) Kurtz random* if A passes every n^k-time (-space) Schnorr test. We denote the set of polynomial time and polynomial space Kurtz random sequences by $\mathsf{RAND}_{\mathsf{K\text{-}p}}$ and $\mathsf{RAND}_{\mathsf{K\text{-}pspace}}$, respectively.

We would expect that placing requirements on the growth rate of martingales weakens the notion of randomness. Wang [74] showed that our intuitions hold in both the polynomial time and space settings.

Theorem 3.13. *The following inclusions are strict.*

1. $\mathsf{RAND}_{\mathsf{p}} \subset \mathsf{RAND}_{\mathsf{S\text{-}p}} \subset \mathsf{RAND}_{\mathsf{K\text{-}p}}$.
2. $\mathsf{RAND}_{\mathsf{pspace}} \subset \mathsf{RAND}_{\mathsf{S\text{-}pspace}} \subset \mathsf{RAND}_{\mathsf{K\text{-}pspace}}$.

However, if we restrict ourselves to computable sequences, Wang showed that the three notions of resource bounded randomness we have seen are actually equivalent.

Theorem 3.14 ([74])**.** *A computable sequence A is polynomial time random if and only if A is polynomial time Schnorr random if and only if A is polynomial time Kurtz random. That is,*

$$\mathsf{REC} \cap \mathsf{RAND}_{\mathsf{p}} = \mathsf{REC} \cap \mathsf{RAND}_{\mathsf{S\text{-}p}} = \mathsf{REC} \cap \mathsf{RAND}_{\mathsf{K\text{-}p}}.$$

The equivalent statement for polynomial space randomness also holds.

Kurtz [41] showed that, even without resource bounds, there is a Kurtz random sequence which does not satisfy the law of large numbers. As noted by Wang, this immediately implies the following proposition.

Proposition 3.15. *There is a polynomial space, and therfore time-, Kurtz random sequence which does not satisfy the law of large numbers.*

We now turn to the second way of defining resource bounded Schnorr and Kurtz randomness. As noted by Sureson [68], the classical notions of Schnorr and Kurtz randomness can be defined using *true orders*.

Definition 3.16. For an unbounded function $f : \mathbb{N} \to \mathbb{N}$, the *inverse* of f is the function $\mathrm{INV}_f(n) = \min\{k \mid f(k) \leq n\}$. An order h is a *true polynomial time (space)-order* if both h and Inv_h belong to FP (resp. $\mathsf{FPSPACE}$).

Without resource bounds, true orders are equivalent to orders, and thus Schnorr and Kurtz randomness can be defined with either. In the resource bounded setting, however, orders and true orders are different. Sureson, using true orders, gave the following notion of randomness.

Definition 3.17 ([68])**.** A sequence A is *martingale polynomial time (space) S random* if for every polynomial time (resp. space) martingale d, and every true polynomial time (resp. space) order h, $d(A\restriction n) < 2^{h(n)}$, for almost every n. We denote the set of all martingale polynomial time and space S random sequences as $\mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$ and $\mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}}$.

It is clear that $\mathsf{RAND}_{\mathsf{S\text{-}p}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$, and that $\mathsf{RAND}_{\mathsf{S\text{-}pspace}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}}$. Sureson proved that martingale polynomial time- (space-) S randomness is weaker than polynomial time (space) randomness.

Lemma 3.18 ([68])**.** *There is a computable sequence $A \in \mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$ that is not polynomial time random. This also holds in the polynomial space setting.*

This fact, together with the Lemma 3.18 and Theorem 3.14 yields the following.

Theorem 3.19 ([74])**.** *The following inclusions are strict.*

1. $\mathsf{RAND}_{\mathsf{p}} \subset \mathsf{RAND}_{\mathsf{S\text{-}p}} \subset \mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$.
2. $\mathsf{RAND}_{\mathsf{pspace}} \subset \mathsf{RAND}_{\mathsf{S\text{-}pspace}} \subset \mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}}$.

To complete the picture of the relations of the notions discussed so far, Sureson relied on the following.

Lemma 3.20 ([68])**.** *Every martingale polynomial time S random sequence satisfies the law of large numbers. Therefore so do martingale polynomial space S random sequences.*

This result, together with Theorem 3.14, Lemma 3.18 and 3.15, implies the following.

Corollary 3.21. *Polynomial time Kurtz randomness and martingale polynomial time S randomness are incomparable. Similarly, polynomial space Kurtz randomness and martingale polynomial space S randomness are incomparable.*

Buss, Cenzer and Remmel [14] recently defined the second resource bounded analog of Kurtz randomness[7]. In the language of true orders, they defined the following[8].

[7] In [14] they only considered polynomial space randomness, although their definitions extend naturally to polynomial time randomness

[8] Buss, Cenzer and Remmel's original definition used different notation. Sureson [68] showed that their definition was equivalent to the one presented here.

Definition 3.22. A sequence A is *martingale BP polynomial time (space) random* if, for every polynomial time (resp. space) martingale d, and every true polynomial time (resp. space) order h, $d(A\upharpoonright n) < 2^{h(n)}$, for infinitely many n. We denote the set of all martingale BP polynomial time and space random sequences as $\mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}p}}$ and $\mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}pspace}}$.

The following inclusions are immediate.

Observation 3.23.

1. $\mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}p}}$ *and* $\mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}pspace}}$.
2. $\mathsf{RAND}_{\mathsf{K\text{-}p}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}p}}$ *and* $\mathsf{RAND}_{\mathsf{K\text{-}pspace}} \subseteq \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}pspace}}$.

Buss, Cenzer and Remmel [14] showed that there is a computable m-BP-pspace random sequence which does not statisfy the law of Large Numbers. Using Lemma 3.20, Sureson showed the following.

Proposition 3.24. *The following inclusions are strict.*

1. $\mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}} \subset \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}p}}$.
2. $\mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}} \subset \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}pspace}}$.

As a result of Theorem 3.19, Observation 3.23 and Proposition 3.24, we have a complete picture of the relations between all the notions defined in this section. We depict the relations in the polynomial time setting in the following picture, and the analogous picture for polynomial space randomness is identical.

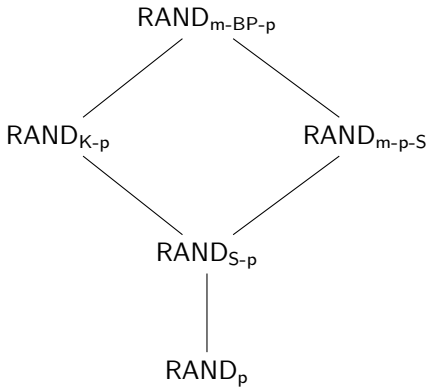


Figure 1: Hierarchy for polynomial time randomness using martingales. The diagram for polynomial space randomness is identical. All lines represent strict inclusions, and all missing lines represent incomparability.

3.2 Using Null Covers

Martin-Löf's original notion of randomness was defined using effective null covers; that is, a descending sequence of uniformly c.e. open sets whose intersection is of measure zero. Null covers are now fundamental in the theory of algorithmic randomness, and almost every significant notion has a null cover characterization.

Null covers have not had the same prominence in resource bounded randomness. One of the main obstacles in generalizing null cover definitions from the computable setting is the difficulty in imposing resource bounds on the concept of enumerability. A much more natural concept for resource bounded computation is decidability. Two notions of resource bounded randomenss, *weak randomness* and *BPS randomness*, have recently been defined using decidable null covers.

3.2.1 Weak Resource Bounded Randomness

Weak resource bounded randomness (weak randomness) was recently defined, independently, by Sureson [68][9], and Huang and Stull [35][10]. To motivate the definition of weak randomness, we first give a characterization of Schnorr randomness based on decidability. Weak randomness will then be defined as the natural resource bounded analog of this characterization.

Definition 3.25. A sequence of open sets $\{U_n\}$, $U_n \subseteq \Sigma^\infty$, is *computably approximable* if there is an array $\{S_n^k\}_{k,n\in\mathbb{N}}$, $S_n^k \subseteq \Sigma^*$, satisfying the following.

1. There is a computable function $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that for every k, n,
$$\max\{|w| \,:\, w \in S_n^k\} \leq f(k, n).$$
2. The language $L = \{\langle w, 0^k, 0^n\rangle \,|\, w \in S_n^k\}$ is decidable.
3. For every n, $U_n = \bigcup_{k\geq 1} [S_n^k]$.
4. For every n and k, $\mu(U_n - \bigcup_{j=1}^{k} [S_n^j]) \leq 2^{-k}$.

It is not hard to verify that if a sequence $\{U_n\}$ is computably approximable, then we can uniformly compute the measures $\mu(U_n)$, giving us the following proposition.

Proposition 3.26. *A sequence of open sets $\{U_n\}$ is a Schnorr test if and only if $\{U_n\}$ is computably approximable and $\mu(U_n) \leq 2^{-n}$.*

[9] Sureson defined this notion , in the polynomial time and space setting, as ML-P-S and ML-PSPACE-S randomness. Sureson used this notation due to its connection with Schnorr randomness.

[10] The original definition of weak randomness in [35] was defined for complexity for points in $\mathbb{R}^n$ instead of binary sequences. We will give the equivalent definition of weak randomness when dealing with infinite sequences instead of real numbers.

We now turn to weak resource bounded randomness. In this survey, we will focus on polynomial time and space bounds, although it easily extends to more refined time and space bounds.

Definition 3.27. A sequence of open sets $\{U_n\}$, $U_n \subseteq \Sigma^\infty$, is *polynomial time (space) approximable* if there is an array $\{S_n^k\}_{k,n\in\mathbb{N}}$, $S_n^k \subseteq \Sigma^*$, such that the following hold.

1. There is a polynomial time (space) computable function $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that, for every k and n,
$$\max\{|w| \,:\, w \in S_n^k\} \leq f(k, n).$$
2. The language $L = \{\langle w, 0^k, 0^n\rangle \,|\, w \in S_n^k\}$ is decidable in polynomial time (space).
3. For every n, $U_n = \bigcup_{k\geq 1} [S_n^k]$.
4. For every n and k, $\mu(U_n - \bigcup_{j=1}^{k} [S_n^j]) \leq 2^{-k}$.

Definition 3.28. A *weak polynomial time (space) test* is a polynomial time (space) approximable sequence of descending open sets $\{U_n\}$ such that $\mu(U_n) \leq 2^{-n}$. An infinite sequence A *passes* a weak polynomial time (space) test if $A \notin \cap_n U_n$. An infinite sequence A is *weakly polynomial time (space) random* if A passes every weak polynomial time (space) test. The set of all weakly polynomial time (space) random sequences is denoted by $\mathsf{RAND}_{\mathsf{W\text{-}p}}$ ($\mathsf{RAND}_{\mathsf{W\text{-}pspace}}$).

As noted by Sureson [68], both weak polynomial time random and weak polynomial space random sequences satisfy the law of large numbers.

Proposition 3.29. *Every weak polynomial time random sequences obeys the law of large numbers.*

This implies the following separation.

Corollary 3.30. $\mathsf{RAND}_{\mathsf{W\text{-}p}}$ *and* $\mathsf{RAND}_{\mathsf{W\text{-}pspace}}$ *are incomparable with* $\mathsf{RAND}_{\mathsf{K\text{-}p}}$ *and* $\mathsf{RAND}_{\mathsf{K\text{-}pspace}}$.

A recurring theme in this survey is that polynomial space randomness is easier to work with than polynomial time randomness[11]. In the polynomial space setting, we are able to compute, using property (1) of Definition 3.27, the measures of U_n uniformly.

Proposition 3.31 ([35], [68])**.** *Let $\{U_n\}$ be a weak polynomial space test. Then the function $f : \Sigma^* \times \mathbb{N} \to [0, 1]$ defined by $f(w, n) = \mu(C_w \cap U_n)$ is polynomial space computable.*

[11] Intuitively, this is due to the famous quote of complexity theory: space can be reused, time cannot. This allows us to enumerate exponentially large sets. Of course this intuition only holds if $\mathsf{P} \neq \mathsf{PSPACE}$.

Sureson used this measurability property (Proposition 3.31) to show that martingale pspace-S randomness is equivalent to weak pspace randomness.

Theorem 3.32 ([68]). $\mathsf{RAND}_{\mathsf{W\text{-}pspace}} = \mathsf{RAND}_{\mathsf{m\text{-}pspace\text{-}S}}$

With this martingale characterization, by Theorem 3.19, it is easy to see the following.

Lemma 3.33. $\mathsf{RAND}_{\mathsf{S\text{-}pspace}}$ *is a strict subset of* $\mathsf{RAND}_{\mathsf{W\text{-}pspace}}$.

Moreover, Huang and Stull, in a personal communication, showed how to construct a sequence which is weak polynomial space random, yet not even n-time random.

Lemma 3.34. *There is a computable sequence A such that A is weakly polynomial space random and there is an $O(n)$-time martingale d succeeding on A.*

The proof of this lemma follows from the following construction. Let A be a computable sequence which is 2^{2^n}-time random (such a sequence exists by an argument similar to that of Lemma 3.9). Then, we construct a sequence B by replacing every 2^{2^k}th bit of A with a 0. It is not difficult to show that B satisfies the required properties.

Unfortunately, we do not know if Proposition 3.31 holds in the polynomial time setting, so we do not know if weak polynomial time randomness is equivalent to martingale polynomial time-S randomness. However, it is not hard to show that weak polynomial time randomness implies martingale polynomial time-S randomness.

Lemma 3.35 ([68]). *If A is weakly polynomial time random, then A is martingale polynomial time-S random.*

Without the converse of Lemma 3.35, there are many open questions relating weak polynomial time randomness with the notions we have discussed so far.

Open Question 3.36.

1. Is there a martingale characterization of weak polynomial time randomness?
2. What is the relation between $\mathsf{RAND}_{\mathsf{W\text{-}p}}$ and $\mathsf{RAND}_{\mathsf{p}}$? What is the relation between $\mathsf{RAND}_{\mathsf{W\text{-}p}}$ and $\mathsf{RAND}_{\mathsf{S\text{-}p}}$?

3.2.2 BP Randomness

Buss, Cenzer and Remmel [14] have recently defined *bounded primitive recursive (BP) randomness*, a variant of Kurtz randomness with primitive recursive restrictions. By imposing space bounds on BP randomness, they defined *bounded polynomial space (BPS) randomness*.

Definition 3.37. A sequence of open sets $\{U_n\}$ is a *BP (resp. BPS) test* if there is a sequence $\{S_n\}_{n\in\mathbb{N}}$, $S_n \subseteq \Sigma^*$, and a polynomial time computable function $f : \mathbb{N} \to \mathbb{N}$ such that the following hold.

1. For every n, $\max\{|w| : w \in S_n\} \leq f(n)$.
2. The language $L = \{\langle w, 0^n \rangle \mid w \in S_n\}$ is decidable in polynomial time (resp. space).
3. For every n, $U_n = \bigcup_{k \geq 1} [S_n]$.
4. For every n, $\mu(U_n) \leq 2^{-n}$.

A sequence A *passes* a BP (resp. BPS) test $\{U_n\}$ if $A \notin U_n$ for some $n \in \mathbb{N}$. A sequence A is *BP-p random* (resp. *BP-pspace random*) if A passes every BP (resp. BPS) test. The set of all BP polynomial time (space) random points is denoted $\mathsf{RAND}_{\mathsf{BP\text{-}p}}$ ($\mathsf{RAND}_{\mathsf{BP\text{-}pspace}}$).

Buss, et al., showed that, in the polynomial space setting, the martingale and open covers notions BP randomness are, in fact, equivalent.

Theorem 3.38 ([14])**.** $\mathsf{RAND}_{\mathsf{BP\text{-}pspace}} = \mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}pspace}}$

This characterization shows that BP-pspace randomness is strictly weaker than polynomial space Kurtz randomness.

Lemma 3.39. $\mathsf{RAND}_{\mathsf{K\text{-}pspace}} \subset \mathsf{RAND}_{\mathsf{BP\text{-}pspace}}$.

Again, in the polynomial time setting, the situation is less clear. We do not know if $\mathsf{RAND}_{\mathsf{m\text{-}BP\text{-}p}} = \mathsf{RAND}_{\mathsf{BP\text{-}p}}$. However, we can, using essentially the same proof as Lemma 3.35, prove the following.

Lemma 3.40. *If A is BP polynomial time random, then A is martingale BP polynomial time random.*

We end this section with the following depiction of the relations between the polynomial space randomness notions introduced so far. While we have a complete picture for the polynomial space randomness concepts, we know very little about the situation in the polynomial time setting.

3.3 Using Kolmogorov Complexity

In this section we will review notions of randomness defined using resource bounded Kolmogorov complexity. Kolmogorov complexity characterizations of randomness have been extensively studied in the computable setting [18, 58]. This has not been the case in the resource bounded setting. However, there have been significant advances in this direction in the past few years. Li and Vitanyi's book [45] on Kolmogorov complexity provides an excellent introduction to Kolmogorov complexity in both settings.

We begin with the time and space bounded analog of the (plain) Kolmogorov complexity of strings.

Definition 3.41. Let M be a Turing machine, t be a time-constructible function and x, y be binary strings. The *t-time bounded Kolmogorov complexity of x relative to M conditional on y* is

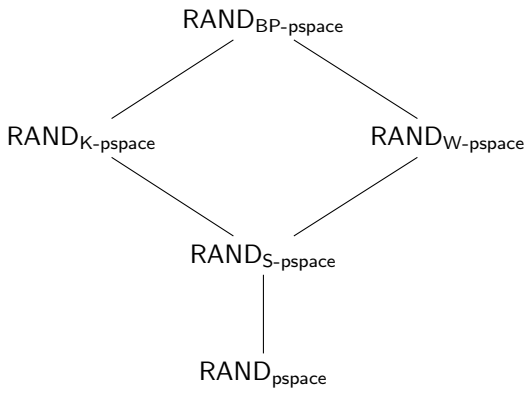


Figure 2: Hierarchy for polynomial space randomness. All lines represent strict inclusions.

$$CT^t_M(x \,|\, y) = \{|\pi| \,|\, \pi \in \Sigma^* \text{ and } M(\pi, y) = x \text{ in } t(|x|) \text{ steps}\}.$$

Similarly, the *t-space bounded Kolmogorov complexity of x relative to M conditional on y* is

$$CS^t_M(x \,|\, y) = \{|\pi| \,|\, \pi \in \Sigma^* \text{ and } M(\pi, y) = x \text{ in } t(|x|) \text{ steps}\}.$$

The *t-time (-space) bounded Kolmogorov complexity of x relative to M* is denoted $CT^t_M(x) = CT^t_M(x \,|\, \lambda)$ (resp. $CS^t_M(x) = CS^t_M(x \,|\, \lambda)$).

We will also be interested in the resource bounded analog of prefix free Kolmogorov complexity. A *self-delimiting Turing machine* is a machine with a one-way, read-only input tape. An input x to a self-delimiting machine M is valid if M halts with the input head reading the last bit of x. This ensures that the set of valid inputs is prefix-free.

Definition 3.42. Let M be a self-delimiting[12] Turing machine, t be a time-constructible function and x, y be binary strings. The *t-time bounded prefix Kolmogorov complexity of x relative to M conditional on y* is

$$KT^t_M(x \,|\, y) = \{|\pi| \,|\, \pi \in \Sigma^* \text{ and } M(\pi, y) = x \text{ in } t(|x|) \text{ steps}\}.$$

Similarly, the *t-space bounded prefix Kolmogorov complexity of x relative to M conditional on y* is

[12]In the non-resource bounded setting, the prefix Kolmogorov complexity of a string can be defined in terms of self-delimiting machines or in terms of partial recursive prefix functions. However, in the time-bounded setting, these two definitions are not necessarily equivalent. Juedes and Lutz [38] showed that the two time-bounded formulations of prefix Kolmogorov complexity are *not* equivalent if $\mathsf{P} \neq \mathsf{NP}$. In particular, they show that, assuming $\mathsf{P} \neq \mathsf{NP}$, the partial recursive definition of time bounded prefix Kolmogorov complexity has no efficient optimal machine.

$$KS^t_M(x \mid y) = \{|\pi| \mid \pi \in \Sigma^* \text{ and } M(\pi, y) = x \text{ in } t(|x|) \text{ steps}\}.$$

The *t-time (-space) bounded prefix Kolmogorov complexity of x relative to M* is denoted $KT^t_M(x) = KT^t_M(x \mid \lambda)$ (resp. $KS^t_M(x) = KS^t_M(x \mid \lambda)$).

The existence of an optimal machine is one of the cornerstones in classical Kolmogorov complexity. With this result, the Kolmogorov complexity of a string becomes an intrinsic property of the string itself, without reference to a Turing machine. We would like to show the existence of an *efficient* optimal machine; that is, an optimal machine for every fixed time or space bound. Unfortunately, this is not possible. However, if we relax the time or space bound slightly, we can show the existence of efficient optimal machines. For a full proof of the following theorem see, e.g. [45].

Theorem 3.43. *There exists an efficient universal Turing machine U such that, for every other Turing Machine M, there is a constant c such that*

$$CT^{ct\log t}_U(x \mid y) \leq CT^t_M(x \mid y) + c, \text{ and}$$
$$CS^{ct\log t}_U(x \mid y) \leq CS^t_M(x \mid y) + c,$$

for all x, where c depends on M but not x and y. Similarly, there is an efficient universal prefix free Turing machine.

We will fix such a machine U and refer to *the* t-time (-space) bounded Kolmogorov complexity of x as $CT^t(x \mid y) = CT^t_U(x \mid y)$ (resp. $CS^t(x \mid y) = CS^t_U(x \mid y)$). Similarly, by fixing a universal prefix free machine P, we denote *the* t-time and -space bounded prefix free Kolmogorov complexity of x as $KT^t(x)$ and $KS^t(x)$, respectively.

As we have mentioned, Kolmogorov complexity characterizations of randomness have played a crucial role in the development of the field. A fundamental theorem of algorithmic randomness is the characterization of Martin-Löf random sequences using prefix free Kolmogorov complexity, due to Levin [44] and Chaitin [15].

Theorem 3.44. *A sequence A is ML-random if and only if there is a constant c such that $K(A\upharpoonright n) > n - c$ for every $n \in \mathbb{N}$.*

Gacs [25] proved a similar characterization of Martin-Löf randomess using plain Kolmogorov complexity.

Theorem 3.45. *A sequence A is ML-random if and only if there is a constant c such that $C(A\upharpoonright n \mid n) > n - K(n) - c$ for every $n \in \mathbb{N}$.*

To define a notion of resource bounded randomness, a natural first attempt would be to replace the complexity notions in the above theorems with their time and space bounded counterparts. Unfortunately, the next two theorems show that this does not provide a notion which matches our intuition of the properties of resource bounded random sequences. It is reasonable to expect, for any specific time (or space) bound t, for there to be $t(n)$-random sequences

which *are* computable. Indeed, this is the case for all resource bounded notions we have considered so far. However, Ko [39] has shown that simply adding resource bounds to the above theorems gives a notion which does not conform to this intuition.

Theorem 3.46 ([39])**.** *Let $t(n) = \Omega(n)$ be a unbounded total recursive function and A be a recursive sequence. There is an unbounded total function f such that, for all n, we have*

$$CT^t(A\upharpoonright n \,|\, n) \leq n - f(n).$$

Furthermore, the conclusion holds for time bounded prefix complexity KT.

Moreover, Ko [39] noticed that Theorem 3.46 can be strengthened under the assumption that the sequence is computable in exponential time.

Theorem 3.47 ([39])**.** *Let A be a language computable in 2^{n^k} time. Then, for every constant k, and almost every n,*

$$KT^{n^k}(A\upharpoonright n \,|\, n) \leq n - \log^k n.$$

Similarly, if A is a language computable in 2^{n^k} space, then for every constant k and almost every n,

$$KS^{n^k}(A\upharpoonright n \,|\, n) \leq n - \log^k n.$$

The proof of this fact follows from viewing A as a language, and noticing that the first $\log^k n$ bits of A correspond to strings of length roughly $\log\log^k n$. We are then able to decide in time (resp. space) polynomial in n, whether each string is in the language A, thereby obtaining the first $\log^k n$ bits of A (as a sequence).

Ko [39] gave, to the best of our knowledge, the first definition of resource bounded randomness using Kolmogorov complexity. As proven by Wang [74], Ko's notion is incomparable to the widely accepted notions of Lutz and Schnorr, and we will therefore discuss it only briefly.

Definition 3.48 ([39])**.** Let A be an infinite sequence. We say that A is *polynomial time Ko random* if, for every polynomial p there is a $k \in \mathbb{N}$ such that

$$KT^p(A\upharpoonright n) \geq n - \log^k n,$$

for almost every $n \in \mathbb{N}$. We say that A is *polynomial space Ko random* if, for every polynomial p there is a $k \in \mathbb{N}$ such that

$$KS^p(A\upharpoonright n) \geq n - \log^k n,$$

for almost every $n \in \mathbb{N}$.

By Lemma 3.9 of Section 3.1, there is a language A decidable in 2^{n^2} time (space) such that A is polynomial time (resp. space) random. Combining this with Theorem 3.47 yields the following corollary.

Corollary 3.49. *There is a language which is polynomial time (space) random but not polynomial time (resp. space) Ko random.*

As observed by Wang [74], by replacing the 2^kth bit, for every k, of a ML random sequence A with a 0, the resulting sequence B is not polynomial time Kurtz random, but *is* polynomial space (hence time) Ko random. Hence Ko randomness and p-randomness are incomparable.

The next notion of randomness we will discuss is due to Sureson [68]. This notion was motivated by the work of Downey and Griffiths [19], who were the first to give a machine characterization of Schnorr randomness. Recall that a *computable measure machine* is a prefix-free machine M such that $\mu([domM])$ is computable.

Theorem 3.50 ([19])**.** *A sequence A is Schnorr random if and only if $K_M(A\upharpoonright n) \geq n - O(1)$ for all computable measure machines M.*

Sureson [68] used this idea to create the following notion of resource bounded randomness using Kolmogorov complexity. Let M be a Turing machine, and $t : \mathbb{N} \to \mathbb{N}$. For every $x, y \in \{0,1\}^*$ we set

$$M_t(x) = y \iff M(x) \text{ halts, and } M(x) \text{ outputs y in at most t steps}$$
$$M_t^{space}(x) = y \iff M(x) \text{ halts, and } M(x) \text{ outputs y using at most t cells}$$

Definition 3.51 ([68])**.** Let M be a prefix machine; that is, a machine whose domain is prefix-free. We say that M is a *p-computable measure machine* if there is a $k \in \mathbb{N}$ such that

$$\mu([\mathsf{dom}(M)] - [\mathsf{dom}(M_{n^k})]) \leq 2^{-n},$$

for every n. We say that M is a *pspace-computable measure machine* if there is a $k \in \mathbb{N}$ such that

$$\mu([\mathsf{dom}(M)] - [\mathsf{dom}(M_{n^k}^{space})]) \leq 2^{-n},$$

for every n.

Definition 3.52. Let A be an infinite sequence. We say that A is *Kolmogorov-p-S random* if for any p-computable measure machine M, there is a $b \in \mathbb{N}$ such that

$$K_M(A\upharpoonright n) > n - b,$$

for every n.

We say that A is *Kolmogorov-pspace-S random* if, for any *pspace*-computable measure machine M, there is a $b \in \mathbb{N}$ such that

$$K_M(A\upharpoonright n) > n - b,$$

for every n.

Sureson [68] showed that, in the polynomial space setting, weak randomness and Kolmogorov-S randomness coincide.

Theorem 3.53. *A sequence A is weak polynomial space random if and only if A is Kolmogorov-pspace-S random.*

This theorem, together with Theorem 3.32, shows that weak polynomial space randomness can be characterized by each of the three principal paradigms: statistical tests, martingales and compressibility (Kolmogorov complexity).

Unfortunately, in the polynomial time setting, we are not able to equate weak randomness and Kolmogorov-S randomness.

Open Question 3.54. Is weak polynomial time randomness equivalent to Kolmogorov-p-S randomness?

Our final notion of resource bounded randomness using Kolmogorov complexity is due to Buss, Cenzer and Remmel[14].

Definition 3.55. An infinite sequence A is *Kolmogorov BP-pspace random* if, for every $k \in \mathbb{N}$, and every polynomial time computable function $f : \mathbb{N} \to \mathbb{N}$,

$$CS^{n^k}(A\restriction |f(n)| - 1) > |f(n)| - n,$$

for all $n \in \mathbb{N}$.

As with weak randomness, Buss, Cenzer and Remmel [14] showed that, in the polynomial space setting, the open cover definition of BP-randomness and the Kolmogorov complexity definition of BP-randomness coincide.

Theorem 3.56 ([14])**.** *A sequence A is BP-pspace random if and only if A is Kolmogorov BP-pspace random.*

Together with Theorem 3.38, this shows that, in the polynomial space setting, BP-randomness can also be characterized using the three principal paradigms.

3.4 Resource Bounded Dimension

The notions of randomness we have described so far have been "binary"; i.e., either a sequence is random or it is not. However, it is clear that some nonrandom sequences are more nonrandom than others. Algorithmic dimension, effectivizations of classical fractal dimensions[13], gives us a tool for quantitatively distinguishing nonrandom sequences.

Lutz [48] gave the first notion of algorithmic dimension by effectivizing Hausdorff dimension. His effectivization was based on *gales*, a generalization of martingales. Subsequently, Athreya, Hitchcock, Lutz and Mayordomo [6] used gales to give an effectivization of packing dimension. As we will see, both notions have strong connections with the success rates of martingales as studied by Schnorr

[13] Falconer's text [20] is an excellent introduction to fractal dimensions and their wide application in mathematics.

[63], Staiger [67] and Ryabko [62]. Algorithmic dimension has since become an important subfield of algorithmic randomness. In particular, resource bounded dimension has become a valuable tool for studying complexity classes.

Chapter 13 of Downey and Hirschfeldt's text [18] gives a nice overview on algorithmic dimension in the computable setting. The survey of Hitchcock, Lutz and Mayordomo [32] gives further information on the applications of resource bounded dimension to complexity theory. More information on most of the content of this section can be found in Mayordomo's survey [54].

As we have stated, Lutz effectivized Hausdorff dimension by defining generalized martingales called *gales*.

Definition 3.57. Let $s \in [0, \infty)$. An *s-gale* is a function $d : \{0,1\}^* \to [0, \infty)$, such that, for every finite string $w \in \{0,1\}^*$,

$$d(w) = 2^{-s}[d(w0) + d(w1)]. \tag{2}$$

As with martingales, we may think of s-gales as a betting strategy. However, unlike martingales, the "payouts" for s-gales become increasingly unfair as s goes to 0.

The benefit of this effectivization is that gales, like martingales, are naturally suited for resource bounded computation. For any function $t : \mathbb{N} \to \mathbb{N}$, we say that an s-gale d is *computable in time (resp. space)* $t(n)$ if there is a $t(n)$-time (resp. space) computable function $f : \Sigma^* \times \mathbb{N} \to \mathbb{Q}$ such that

$$|d(w) - f(w, r)| \leq 2^{-r}.$$

We say that an s-gale d *succeeds* on an infinite binary sequence A if

$$\limsup_{n\to\infty} d(A\upharpoonright n) = \infty.$$

We say that an s-gale d *strongly succeeds* on A if

$$\liminf_{n\to\infty} d(A\upharpoonright n) = \infty.$$

The *success set* of an s-gale d, $S^\infty(d)$, is the set of sequences on which d succeeds. The *strong success set* of an s-gale d, $S^\infty_{str}(d)$ is the set of sequences on which d strongly succeeds.

Lutz [48] and Athreya et, al. [6], used the success and strong success sets of s-gales to define effectivizations of Hausdorff and packing dimensions, respectively. As in the case of resource bounded randomness, both notions were originally defined for classes of languages, instead of sequences. However, as noted by Lutz [49], both implicitly define a notion of dimension for sequences.

Definition 3.58. Let $t : \mathbb{N} \to \mathbb{N}$ be a function and $A \in \mathbf{C}$.

1. The *$t(n)$-time (-space) dimension* of A is

$$\dim_t(A) = \inf\{s \mid (\exists t(n)\text{-time (-space) s-gale } d)\, A \in S^\infty(d)\}.$$

2. The *$t(n)$-time (-space) strong dimension* of A is

$$\mathrm{Dim}_t(A) = \inf\{s \mid (\exists t(n)\text{-time (-space) s-gale } d)\, A \in S^\infty_{str}(d)\}.$$

It is immediate from the definition that $\dim_t(A) \leq \mathrm{Dim}_t(A)$ for every sequence A and function $t(n)$.

We will be interested in the dimension of sequences relative to classes of functions, particularly the class of polynomials.

Definition 3.59. Let $A \in \mathbf{C}$.

1. The *polynomial time dimension* of A is

$$\dim_p(A) = \inf\{s \mid (\exists k)\, \text{there is a } n^k\text{-time s-gale } d \text{ s.t. } A \in S^\infty(d)\}.$$

2. The *polynomial time strong dimension* of A is

$$\mathrm{Dim}_p(A) = \inf\{s \mid (\exists k)\, \text{there is a } n^k\text{-time s-gale } d \text{ s.t. } A \in S^\infty_{str}(d)\}.$$

The polynomial space dimension, $\dim_{pspace}$ and strong dimension Dim_{pspace}, are defined similarly.

We now show that the analogs of Lemmas 3.6 and 3.7 hold for both resource bounded dimension and strong dimension. An s-gale $d : \Sigma^* \to [0, \infty)$ is *exactly $t(n)$-time (-space) computable* if d is $t(n)$-time (-space) computable and $d(w) \in \mathbb{Q}$ for every $w \in \Sigma^*$.

Lemma 3.60 ([48],[6])**.** *If d is a polynomial time computable s-gale and 2^s is rational, then there is an exactly polynomial time computable s-gale d' such that*

$$S^\infty(d) = S^\infty(d'), \text{ and}$$
$$S^\infty_{str}(d) = S^\infty_{str}(d').$$

Lemma 3.61 ([48],[6])**.** *Let 2^s be rational. If a set of polynomial time computable s-gales $\{d_n\}$ is uniformly polynomial time computable, then there is a polynomial time computable s-gale d such that*

$$S^\infty(d) = \cup_n S^\infty(d_n), \text{ and}$$
$$S^\infty_{str}(d) = \cup_n S^\infty_{str}(d_n).$$

As mentioned at the beginning of this section, resource bounded dimension gives us a tool for quantitatively distinguishing nonrandom sequences. For example, we would intuitively say that the join of a polynomial time random sequence with a trivial sequence is "half" random. The next example shows that dimension, and strong dimension, match this intuition.

Example 3.62. Let $A \in \mathbf{C}$ be polynomial time random. Then $\mathrm{Dim}_p(A \bigoplus 0) = \dim_p(A \bigoplus 0) = \frac{1}{2}$, where $\bigoplus$ is the join operator and 0 is the sequence of all zeros.

While we have so far introduced resource bounded dimension as a tool for distinguishing nonrandom sequences, it also gives a weak notion of randomness. Let $\dim_p^{=1}$ and $\mathrm{Dim}_p^{=1}$ denote the set of all sequences of dimension and strong dimension 1, respectively. Similarly, define $\dim_{pspace}^{=1}$ and $\mathrm{Dim}_{pspace}^{=1}$. These sets define randomness notions. It is clear that

$$\mathsf{RAND}_p \subseteq \dim_p^{=1} \subseteq \mathrm{Dim}_p^{=1}, \text{ and}$$
$$\mathsf{RAND}_{pspace} \subseteq \dim_{pspace}^{=1} \subseteq \mathrm{Dim}_{pspace}^{=1}.$$

Moreover, all of these inclusions are strict. To see this, it suffices to take a computable sequence which is polynomial time random, and replace the 2^kth bit with 0. The resulting sequence is not polynomial time random, yet has polynomial time dimension 1.

Effective dimension as a notion of randomness has a close connection to the rate of success of martingales. Schnorr [65, 63] defined a martingale d to have *exponential order* on a sequence A if

$$\limsup_{n\to\infty} \frac{\log d(A\upharpoonright n)}{n} > 0, \tag{3}$$

Subsequently, Ryabko [62] and Staiger [67] studied the left hand limit of inequality (3). As noted by Terwijn, in a personal communication to Lutz, there is a martingale with exponential order succeeding on a sequence A if and only if A has algorithmic dimension less than 1. This fact, with an essentially identical proof, generalizes to resource bounded dimension.

Lemma 3.63. *A sequence A has $\dim_p(A) < 1$ if and only if there is a polynomial time computable martingale d such that*

$$\limsup_{n\to\infty} \frac{\log d(A\upharpoonright n)}{n} > 0.$$

Similarly, A has $\mathrm{Dim}_p(A) < 1$ if and only if there is a polynomial time computable martingale d such that

$$\liminf_{n\to\infty} \frac{\log d(A\upharpoonright n)}{n} > 0.$$

From the characterization of resource bounded dimension and strong dimension of Lemma 3.63, it is not hard to show the following.

Corollary 3.64. *The following inclusions are strict.*

$$\mathsf{RAND}_{\mathsf{K\text{-}p}} \subset \mathrm{Dim}_p^{=1}, \text{ and}$$
$$\mathsf{RAND}_{\mathsf{S\text{-}p}} \subset \dim_p^{=1}.$$

These (strict) inclusions also hold in the polynomial space setting.

We are also able to separate polynomial time (space) Kurtz randomness and $\dim_p^{=1}$ (resp. $\dim_{pspace}^{=1}$).

Lemma 3.65. *Polynomial time (space) Kurtz randomness is incomparable with* $\dim_p^{=1}$ *and* $\dim_{pspace}^{=1}$.

To see that there is a polynomial time Kurtz random sequence which is not of polynomial time dimension 1 , recall that there are Kurtz random sequences which do not satisfy the law of large numbers. However, every sequence of polynomial time dimension 1 does[14]. For the other direction, there are computable sequences which are of polynomial time dimension 1 which are not polynomial time random. The proof follows from Wang's result 3.14.

Lemma 3.63, combined with the martingale characterization of weak polynomial space randomness (Theorem 3.32) also relates resource bounded dimension and weak randomness.

Corollary 3.66.

1. $\mathsf{RAND}_{\mathsf{W\text{-}pspace}}$ *is a subset of* $\dim_{pspace}^{=1}$.
2. $\mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$ *is a subset of* $\dim_p^{=1}$.

Corollary 3.66 can, in fact, be strengthened.

Lemma 3.67.

1. $\mathsf{RAND}_{\mathsf{W\text{-}pspace}}$ *is a strict subset of* $\dim_{pspace}^{=1}$.
2. $\mathsf{RAND}_{\mathsf{m\text{-}p\text{-}S}}$ *is a strict subset of* $\dim_p^{=1}$.

The proof of this lemma follows from the following construction. Let A be a Martin-Löf random sequence. Replace the k^2th bit of A with a 0. It is not difficult to verify that the resulting sequence is not martingale polynomial time Schnorr random, yet has polynomial space dimension 1.

This construction actually proves that there is a sequence of polynomial time (space) dimension 1 which is not polynomial time (resp. space) BP random. This, along with the fact that there are BP random sequences which do not satisfy the law of large numbers, proves the following.

Lemma 3.68. *Polynomial time (space) BP randomness is incomparable with* $\dim_p^{=1}$ *(resp.* $\dim_{pspace}^{=1}$*).*

In the computable setting, Mayordomo has shown that effective dimension can be characterized using the density of algorithmic information [53]. Hitchcock, in his PhD thesis, [31] showed that Mayordomo's result extends to polynomial space dimension[15].

Theorem 3.69 ([31])**.** *Let* $A \in \mathbf{C}$. *Then*

$$\dim_{pspace}(A) = \liminf_{n\to\infty} \frac{KS^p(A\upharpoonright n)}{n},$$

[14] In fact, every sequence of *finite state* dimension 1 satisfies the law of large numbers. We will discuss finite state dimension in the next section.

[15] See Section 3.3 for the definitions of resource bounded Kolmogorov complexity.

and

$$\mathrm{Dim}_{pspace}(A) = \limsup_{n\to\infty} \frac{KS^p(A\upharpoonright n)}{n},$$

In the polynomial time setting, only a lower bound is known, also due to Hitchcock [31].

Theorem 3.70 ([31])**.** *Let $A \in \mathbf{C}$. Then,*

$$\dim_p(A) \geq \liminf_{n\to\infty} \frac{KT^p(A\upharpoonright n)}{n}.$$

Open Question 3.71. Is it true that $\dim_p(A) = \liminf_{n\to\infty} \frac{KT^p(A\upharpoonright n)}{n}$? Does this hold for strong dimension?

3.5 Using Finite State Machines

In this section, we introduce the most restrictive notion of resource bounded randomness, finite state dimension. Finite state dimension is based on the definition of a finite state gambler. Finite state gamblers and randomness were first studied by Schnorr and Stimm [64] and Feder [21]. Finite state dimension was introduced by Dai, Lathrop, Lutz and Mayordomo as a finite state version of algorithmic dimension.

Before giving definitions of finite state gamblers and dimension, we will use the following notation. Let $k \geq 2$. A probability function on $\Sigma_k = \{0, 1, \ldots, k-1\}$ is a function $\pi : \Sigma_k \to [0, 1]$ satisfying

$$\sum_{a\in\Sigma_k} \pi(a) = 1.$$

The set of all rational probability measures on Σ_k is denoted by $\Delta_{\mathbb{Q}}(\Sigma_k)$.

From the intuition of gales as a betting strategy, we informally define a sequence as non-random if some computationally bounded gambler can win an unbounded amount of money by betting on the sequence. Informally, finite state dimension is defined by restricting the computational power of the gambler to that of a finite state machine. With such a severe restriction, we will have to make a notable change. A martingale is, more accurately, the bookkeeper of a gambler. It keeps track of how much money the gambler currently has. Under less restrictive resource bounds, this is computationally equivalent to the gambling strategy itself. However, finite state machines are a *memory-less* model of computation, and, therefore, unable to keep track of the gambler's current wealth. Instead, we will separate notions of the gambler and the gale (the bookkeeper).

A finite state gambler is, informally, a finite state machine which, at each state, specifies the amount of money to bet on the next symbol.

Definition 3.72. Let $k \geq 2$. A *finite-state gambler (FSG)* is a 5-tuple

$$G = (Q, \Sigma_k, \delta, \beta, q_0),$$

where

- Q is a finite set of states,
- $\Sigma_k = \{0, 1, \ldots, k-1\}$ is a finite alphabet,
- $\delta : Q \times \Sigma_k \to Q$ is the *transition function*,
- $\beta : Q \to \Delta_{\mathbb{Q}}(\Sigma_k)$ is the *betting function*, and
- $q_0 \in Q$ is the *initial state*.

We may extend the transition function δ to a function $\delta : Q \times \Sigma_k^* \to Q$ in the standard way, and then denote $\delta(w) = \delta(q_0, w)$.

Given a finite state gambler G, we can define the s-gale induced by G in a natural way. It is useful to think of the gale as the bookkeeper of G.

Definition 3.73. Let $k \geq 2$, $G = (A, \Sigma_k, \delta, \beta, q_0)$ be a finite-state gambler, and $s \in [0, \infty)$. The *s-gale* of G is the function

$$d_G^{(s)} : \Sigma_k^* \to [0, \infty),$$

defined by the recursion

$$d_G^{(s)}(\lambda) = 1$$
$$d_G^{(s)}(wa) = k^s d_G^{(s)}(w)\beta(\delta(w))(a),$$

for all $w \in \Sigma_k^*$ and $a \in \Sigma_k$.

We say that an FSG G *s-succeeds* on an infinite k-ary sequence $A \in \Sigma_k^\infty$ if

$$\limsup_{n\to\infty} d_G^{(s)}(A\upharpoonright n) = \infty.$$

An FSG G *strongly s-succeeds* on an infinite k-ary sequence A if

$$\liminf_{n\to\infty} d_G^{(s)}(A\upharpoonright n) = \infty.$$

Definition 3.74. Let $A \in \Sigma_k^\infty$. The *finite-state dimension of* A is

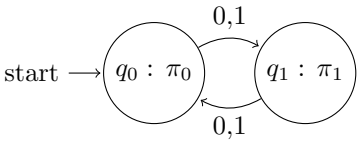
$$\dim_{\mathrm{FS}}(A) = \inf\{s \mid \text{ there is an FSG that s-succeeds on } A\}.$$

The *finite-state strong dimension of* A is

$$\mathrm{Dim}_{\mathrm{FS}}(A) = \inf\{s \mid \text{ there is an FSG that strongly s-succeeds on } A\}.$$

If A is a sequence with $\dim_{FS}(A) = 1$, we say that A is *finite state random*.

Example 3.75. Let $A \in \mathbf{C}$. We will construct a finite state gambler which bets on $A \bigoplus 0$ in the obvious way. On every even bit it bets all of its money on 0, and on every odd bit it "hedges", and bets half its money on 0, half on 1. Let $G = (Q, \Sigma, \delta, \beta, q_0)$ (depicted below), where, $\pi_0(0) = 1$, $\pi_0(1) = 0$, and $\pi_1(0) = \pi_1(1) = \frac{1}{2}$



Then the $\frac{1}{2}$-gale of G succeeds on A.

As we will see in the next section, finite state dimension is closely connected with the theory of Borel normal sequences. For now, we will state a weaker theorem.

Theorem 3.76 ([64])**.** *Every sequence of finite state dimension* 1 *satisfies the law of large numbers.*

4 Applications to Complexity Theory

In this section we will describe applications of resource bounded randomness to computational complexity theory. We will begin by defining Lutz's notion of resource bounded measure and resource bounded dimension. Both notions are highly connected with resource bounded randomness, which we describe in Section 4.3. In Sections 4.4, 4.5, 4.6 and 4.7, we describe several applications of resource bounded measure and dimension to complexity theory. The surveys of Ambos-Spies and Mayordomo [2] and Lutz [47] are excellent, albeit outdated, overviews of resource bounded measure in complexity theory. The survey of Hitchcock, Mayordomo and Lutz [32] gives a great overview of the applications of resource bounded dimension. As much as is possible, we will try to highlight results not found in these surveys. However, even with this constraint, a full overview of randomness in complexity theory would require a survey unto itself.

4.1 Resource Bounded Measure

In this section we will formally define Lutz's notion of resource bounded measure theory. We will then state several fundamental theorems, and briefly describe its applications to complexity theory, which will be expanded upon in Sections 4.4, 4.5, 4.6 and 4.7.

Let $X \subseteq \mathbf{C}$ be a set of infinite sequences and let d be a martingale. We say that d *succeeds on* X if d succeeds on every sequence $A \in X$.

Definition 4.1. Let $X \subseteq \mathbf{C}$ be a set of infinite sequences and let t be a time-constructible function. X *has* t*-time (-space) measure* 0, denoted $\mu_t(X) = 0$ (resp. $\mu_{t-space}(X) = 0$), if there is a t-time (-space) computable martingale d which succeeds on X. X *has* t*-time (-space) measure* 1, denoted $\mu_t(X) = 1$ (resp. $\mu_{t-space}(X) = 1$), if $\mu_t(X^C) = 0$ (resp. $\mu_{t-space}(X^C) = 0$).

As throughout this survey, we will be interested in resource bounded measure with respect to classes of functions, particularly the class of polynomial time

and space computable functions. We will also be interested in the class of quasi-polynomial time computable functions.

Definition 4.2. Let $X \subseteq \mathbf{C}$ be a set of infinite sequences.

1. *X has p-measure* 0, denoted $\mu_p(X) = 0$, if $\mu_{n^k}(X) = 0$ for some $k \in \mathbb{N}$.
2. *X has p_2-measure* 0, denoted $\mu_{p_2}(X) = 0$, if $\mu_{n^{(\log n)^k}}(X) = 0$ for some $k \in \mathbb{N}$.
3. *X has pspace-measure* 0, denoted $\mu_{pspace}(X) = 0$, if $\mu_{n^k\text{-}space}(X) = 0$ for some $k \in \mathbb{N}$.

We then define measure 1 sets X as before. For example, *X has p-measure* 1, denoted $\mu_p(X) = 0$ if $\mu_p(X^C) = 0$.

Lutz used the above (resource bounded) measures to define probability measures on well studied complexity classes, such as E, EXP and ESPACE.

Definition 4.3. Let $X \subseteq \mathbf{C}$. *X has measure* 0 *in* E, denoted $\mu(X \mid E) = 0$, if $\mu_{p_1}(X \cap \mathsf{E}) = 0$. *$X$ has measure* 1 *in* E, denoted $\mu(X \mid E) = 1$ if $\mu(X^C \mid E) = 0$. We can extend these definitions to EXP and ESPACE by replacing p_1 with p_2 and *pspace* appropriately.

The following observation is immediate.

Observation 4.4. *For every $X \subseteq \mathbf{C}$ the following hold.*

$$\begin{array}{ccc} \mu_p(X) = 0 & \Longrightarrow & \mu_{p_2}(X) = 0 \\ \Downarrow & & \Downarrow \\ \mu(X \mid \mathsf{E}) = 0 & & \mu(X \mid \mathsf{EXP}) = 0 \end{array}$$

$$\begin{array}{ccc} \mu_p(X) = 0 & \Longrightarrow & \mu_{pspace}(X) = 0 \\ \Downarrow & & \Downarrow \\ \mu(X \mid \mathsf{E}) = 0 & & \mu(X \mid \mathsf{ESPACE}) = 0 \end{array}$$

Lutz [46] showed that this definition endows complexity classes with an internal measure structure. We now state several results which justify this claim.

Observation 4.5 ([46])**.** *Let X, Y be sets of infinite sequences with $X \subseteq Y$. Then*

$$\mu(X \mid \mathsf{E}) \leq \mu(Y \mid \mathsf{E}).$$

This also holds for EXP *and* ESPACE.

One of the defining properties of a measure, in the classical sense, is its countable additivity. While resource bounded measure does not statisfy countable additivity for arbitrary unions, Lutz [46] showed that a restricted form does hold.

Lemma 4.6. *Let $X = \bigcup_i X_i$ be a union of classes and $k \in \mathbb{N}$ such that, for every j, $\mu_{n^k}(X_j) = 0$. Then $\mu_p(X) = 0$. The analogous statement for p_2 and pspace holds, as well as for the measures in* E*,* EXP*, and* ESPACE*.*

The most important justification that Definition 4.3 endows the classes E, EXP and ESPACE with an internal measure structure is the following theorem.

Theorem 4.7 ([46])**.** $\mu(\mathsf{E} \mid E) = 1$, $\mu(\mathsf{EXP} \mid EXP) = 1$, $\mu(\mathsf{ESPACE} \mid \mathsf{ESPACE}) = 1$.

As noted by Lutz [47], with this theorem we are able to say that, if $\mu(X \mid \mathsf{E})$, then $X \cap \mathsf{E}$ is a negligibly small subset of E (and similarly for EXP and ESPACE).

A very useful theorem, due to Regan, Sivakumar and Cai [61], shows that if a class X satisfies basic assumptions and has full measure in E, EXP or ESPACE, then X contains the corresponding class.

Theorem 4.8. *Let X be a set of languages that is either closed under symmetric difference or closed under (finite) union and intersection.*

1. *If $\mu(X \mid \mathsf{E}) = 1$, then $\mathsf{E} \subseteq X$.*
2. *If $\mu(X \mid \mathsf{EXP}) = 1$, then $\mathsf{EXP} \subseteq X$.*
3. *If $\mu(X \mid \mathsf{ESPACE}) = 1$, then $\mathsf{ESPACE} \subseteq X$.*

This implies, for example, that if NP has measure 1 in EXP, then NP = EXP.

We will end this section with a brief discussion of the application of resource bounded measure to computational complexity theory. We will expand on this topic in Sections 4.4, 4.5, 4.6 and 4.7.

Resource bounded measure theory was introduced as a tool to study complexity classes. With a notion of measure, we can state that a certain class X is *small* in, for example, E, rather than simply separating X and E. Our first example, due to Lutz [46], shows that for every fixed $k \in \mathbb{N}$, $\mathsf{Time}(2^{kn})$ is small in E.

Lemma 4.9 ([46])**.**

1. $\mu_p(\mathsf{Time}(2^{kn})) = 0$.
2. $\mu_{p_2}(\mathsf{Time}(2^{n^k})) = 0$.
3. $\mu_{pspace}(\mathsf{Space}(2^{kn})) = 0$.

This lemma follows from the fact that, for every fixed k, there is an enumeration of machines $M_1, M_2, \ldots$ whose union decides every language in $\mathsf{Time}(2^{kn})$ and a universal machine U such that

$$U(i, w) = M_i(w),$$

for every string w. For each M_i, it is not hard to construct an n^k-time martingale d_i that bets according to M_i, and thus succeeding on $L(M_i)$, the language of M_i. The conclusion then follows from Lemma 4.6.

One of the major successes of resource bounded measure is its use in the Lutz conjecture.

Conjecture 4.10. *[Lutz Conjecture]* NP *does not have p-measure* 0 *(*$\mu_p(\mathsf{NP}) \neq 0$*).*

This conjecture has proven to be an influential scientific hypothesis. Many commonly believed conjectures have been shown to follow from the assumption that NP is not small. For example, by Lemma 4.9, proving that $\mu(\mathsf{NP}) \neq 0$ would immediately solve the celebrated P vs NP conjecture.

4.2 Resource Bounded Dimension

As we have discussed, resource bounded dimension was originally defined by Lutz [48] in order to study the Hausdorff dimension of complexity classes. We will use the definitions of Section 3.4 to give a brief overview of resource bounded dimension.

Definition 4.11 ([48], [6])**.** Let $X \subseteq \mathbf{C}$ be a set of infinite sequences.The *p-dimension of* X is

$$\dim_p(X) = \inf\{s \mid (\exists \text{ polynomial time } s\text{-gale } d)\, X \subseteq S^\infty(d)\}.$$

The definitions of p_2-dimension ($\dim_{p_2}$) and *pspace*-dimension ($\dim_{pspace}$) are defined analogously.

The *strong p-dimension of* X is

$$\mathrm{Dim}_p(X) = \inf\{s \mid (\exists \text{ polynomial time } s\text{-gale } d)\, X \subseteq S^\infty_{str}(d)\}.$$

The definitions of strong p_2-dimension (Dim_{p_2}) and strong *pspace*-dimension (Dim_{pspace}) are defined analogously.

As with resource bounded measure, we will also be interested in the dimension of classes *inside* E, EXP and ESPACE.

Definition 4.12 ([48], [6])**.** Let $X \subseteq \mathbf{C}$ be a set of infinite sequences.

1. The *dimension of* X *in* E is $\dim(X \mid \mathsf{E}) = \dim_p(X \cap \mathsf{E})$.
2. The *dimension of* X *in* EXP is $\dim(X \mid \mathsf{EXP}) = \dim_{p_2}(X \cap \mathsf{EXP})$.
3. The *dimension of* X *in* ESPACE is $\dim(X \mid \mathsf{ESPACE}) = \dim_{pspace}(X \cap \mathsf{ESPACE})$.

The strong dimensions $\mathrm{Dim}(X \mid \mathsf{E})$, $\mathrm{Dim}(X \mid \mathsf{EXP})$, $\mathrm{Dim}(X \mid \mathsf{ESPACE})$ are defined analogously.

Many of the common notions of dimension in Fractal Geometry, including the Hausdorff and packing dimensions, satisfy the following properties.

1. $\dim(\emptyset) = 0$, and $\dim(\mathbf{C}) = 1$.
2. For every $X, Y \subseteq \mathbf{C}$ such that $X \subseteq Y$, then $\dim(X) \leq \dim(Y)$.
3. If $\dim(X) < 1$, then $\mu(X) = 0$.
4. For any countable union $X = \cup_{i \in \mathbb{N}} X_i$, $\dim(X) = \sup_i \dim(X_i)$.

Lutz [48] and Athreya et al [6] observed that the Hausdorff and packing dimensions in E, EXP and ESPACE satisfy items (2) and (3) above.

Observation 4.13.

1. *For every $X \subseteq Y$,*

$$\begin{aligned} \dim_p(X) &\leq \dim_p(Y), \text{ and} \\ \dim(X \mid \mathsf{E}) &\leq \dim(Y \mid \mathsf{E}) \end{aligned}$$

2. *For every X,*

$$0 \leq \dim(X \mid \mathsf{E}) \leq \dim_p(X) \leq 1$$

3. *For every X,*

$$\begin{aligned} \dim_p(X) < 1 &\implies \mu_p(X) = 0, \text{ and} \\ \dim(X \mid \mathsf{E}) < 1 &\implies \mu(X \mid \mathsf{E}) = 0 \end{aligned}$$

These properties also hold for the strong dimension Dim*, and for the classes* EXP *and* ESPACE*.*

As a corollary of Theorem 4.7 and Observation 4.13, we see that the notions of resource bounded dimension satisfy the first item of the properties above.

Corollary 4.14.

1. $\dim(\mathsf{E} \mid \mathsf{E}) = \mathrm{Dim}(\mathsf{E} \mid \mathsf{E}) = 1$
2. $\dim(\mathsf{EXP} \mid \mathsf{EXP}) = \mathrm{Dim}(\mathsf{EXP} \mid \mathsf{EXP}) = 1$
3. $\dim(\mathsf{ESPACE} \mid \mathsf{ESPACE}) = \mathrm{Dim}(\mathsf{ESPACE} \mid \mathsf{ESPACE}) = 1$

The final property (property (4) above) we would like our notions of resource bounded dimension to have does *not* hold. However, as with resource bounded measure, if we restrict the countable union to be uniform in some fixed resource bound, the property does hold.

Definition 4.15 ([48])**.**

1. X is a *p-union of the p-dimensional sets* $X_1, X_2, \ldots$ if $X = \cup_i X_i$ and for each $s > \sup_i \dim_p(X_i)$ with 2^s rational, there is a function $d : \mathbb{N} \times \{0,1\}^* \to [0, \infty)$ with the following properties.
   - d is p-computable.
   - For each i, if we write $d_i(w) = d(i, w)$, then the function d_i is an s-gale.
   - For each i, $X_i \subseteq S^\infty(d_i)$.
2. X is a *p-union of the set* $X_1, X_2, \ldots$ *dimensioned in* E if $X = \cup_i X_i$ and $X \cap \mathsf{E}$ is a p-union of the p-dimensional sets $X_1 \cap \mathsf{E}, X_2 \cap \mathsf{E}, \ldots$.

We define *p_2-unions of p_2-dimensional sets*, and *pspace-unions of pspace-dimensional sets* in the same way.

Lemma 4.16 ([48], [6])**.** *If X is a p-union of p-dimensional sets $X_1, X_2, \ldots$, then*

1. $\dim_p(X) = \sup_i \dim_p(X_i)$*, and*
2. $\mathrm{Dim}_p(X) = \sup_i \mathrm{Dim}_p(X_i)$.

If X is the p-union of the sets X_1, $X_2, \ldots$ dimensioned in E, then

1. $\dim(X \mid \mathsf{E}) = \sup_i \dim(X_i \mid \mathsf{E})$*, and*
2. $\mathrm{Dim}(X \mid \mathsf{E}) = \sup_i \mathrm{Dim}(X_i \mid \mathsf{E})$

This lemma also holds for p_2- and pspace-unions.

We will end this section with a brief discussion of the application of resource bounded dimension to computational complexity theory. We will expand on this topic in Sections 4.4, 4.5, 4.6 and 4.7.

Resource bounded dimension was initially conceived by Lutz as a tool to study complexity classes. In particular, it provides a way to quantitatively distinguish "small" complexity classes, classes of resource bounded measure zero. As a first example, we show that Lemma 4.9 can be improved to show that for fixed k, $\mathsf{Time}(2^{kn})$ has 0 dimension. The proof of this is essentially that of Lemma 4.9, using Lemma 4.16.

Lemma 4.17 ([48], [6])**.**

1. $\dim_p(\mathsf{Time}(2^{kn})) = \mathrm{Dim}_p(\mathsf{Time}(2^{kn})) = 0$.
2. $\dim_{p_2}(\mathsf{Time}(2^{n^k})) = \mathrm{Dim}_{p_2}(\mathsf{Time}(2^{n^k})) = 0$.
3. $\dim_{pspace}(\mathsf{Space}(2^{kn})) = \mathrm{Dim}_{pspace}(\mathsf{Space}(2^{kn})) = 0$.

Lutz also formulated the dimension analog of the Lutz conjecture.

Conjecture 4.18 (Weak Lutz Conjecture)**.** $\dim_p(\mathsf{NP}) \neq 0$

Clearly, the Lutz conjecture (Conjecture 4.10) implies the weak Lutz conjecture. By Lemma 4.17, the weak Lutz conjecture also implies $\mathsf{P} \neq \mathsf{NP}$. The weak Lutz conjecture has also proven to be a scientific hypothesis with a lot of explanatory power. We will describe some of the implications in the next sections.

4.3 Connection with Resource Bounded Randomness

As we have mentioned, Lutz's resource bounded measure theory has strong connections with the notion of resource bounded randomness presented in Section 3.1. We will illustrate this with the following three results. The first, due to Ambos-Spies and Mayorodomo [2], relates $t(n)$-measure and $t(n)$-randomness.

Proposition 4.19. *For any sequence A, the following are equivalent.*

1. *A is $t(n)$ random.*
2. *$\mu_t(\{A\}) \neq 0$.*
3. *For every $t(n)$-measure-1 set X, $A \in X$.*

The second, due to Ambos-Spies et al. [5], characterizes polynomial time, polynomial space and quasi-polynomial time measure zero sets with their corresponding randomness notions.

Lemma 4.20. *Let X be a set of infinite sequences.*

$$\begin{aligned}
\mu_p(X) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^k} \cap X = \emptyset \\
\mu_{p_2}(X) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^{\log^k n}} \cap X = \emptyset \\
\mu_{pspace}(X) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^k\text{-}\mathbf{space}} \cap X = \emptyset \\
\mu(X \mid \mathsf{E}) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^k} \cap \mathsf{E} \cap X = \emptyset \\
\mu(X \mid \mathsf{EXP}) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^{\log^k n}} \cap \mathsf{EXP} \cap X = \emptyset \\
\mu(X \mid \mathsf{ESPACE}) = 0 &\iff (\exists k \in \mathbb{N})\, \mathsf{RAND}_{\mathbf{n}^k\text{-}\mathbf{space}} \cap \mathsf{ESPACE} \cap X = \emptyset
\end{aligned}$$

The third result, due to Juedes and Lutz [37] and Ambos-Spies, Terwijn and Zhang [5] is the strongest result connected resource bounded measure and randomness. However, it requires (a relatively weak) assumption on the class X.

Theorem 4.21. *Let X be downward closed under $\leq_m^p$ reductions. The following are equivalent.*

1. *For some $k \geq 2$, X contains a n^k-random sequence.*
2. *For all $k \geq 2$ X contains a n^k-random sequence.*

3. *For all $k \geq 2$, X contains a $n^{\log^k n}$-random sequence.*

4. $\mu_p(X) \neq 0$.

5. $\mu_{p_2}(X) \neq 0$.

The analogous theorem for pspace and p_2-space also holds.

As to be expected, the resource bounded dimension of sets is strongly connected with the resource bounded dimension of sequences discussed in Section 3.4. Ambos-Spies et al. proved the following characterization of resource bounded dimension [4].

Proposition 4.22. *Let X be a set of infinite sequences. Then*

$$\begin{aligned}
\dim_p(X) &= \inf_{k\geq 1} \sup_{A\in X} \dim_{n^k}(A) \\
\dim_{p_2}(X) &= \inf_{k\geq 1} \sup_{A\in X} \dim_{n^{\log^k n}}(A) \\
\dim_{pspace}(X) &= \inf_{k\geq 1} \sup_{A\in X} \dim_{n^k\text{-}space}(A)
\end{aligned}$$

Essentially the same proof shows that this extends to resource bounded *strong* dimension.

Proposition 4.23. *Let X be a set of infinite sequences. Then*

$$\begin{aligned}
\mathrm{Dim}_p(X) &= \inf_{k\geq 1} \sup_{A\in X} \mathrm{Dim}_{n^k}(A) \\
\mathrm{Dim}_{p_2}(X) &= \inf_{k\geq 1} \sup_{A\in X} \mathrm{Dim}_{n^{\log^k n}}(A) \\
\mathrm{Dim}_{pspace}(X) &= \inf_{k\geq 1} \sup_{A\in X} \mathrm{Dim}_{n^k\text{-}space}(A)
\end{aligned}$$

4.4 Derandomization and Zero-one Laws

We will use resource bounded measure theory to prove results on *probabilistic classes*, under varying assumptions. We will begin by defining the most common probabilistic complexity classes. A diagram of known inclusions is on the following page (Figure 1).

The class of *bounded-error probabilistic polynomial time* languages, denoted BPP[16], consists of all languages for which there is a polynomial time Turing machine M and a polynomial p such that

$$\Pr_{w\in\{0,1\}^{p(|x|)}}[M(x,w) = [x \in L]] \geq \tfrac{2}{3},$$

for every $x \in \{0,1\}^*$. The choice of $\frac{2}{3}$ is arbitrary. We may replace this with any constant fraction greater than $\frac{1}{2}$. The class of *randomized polynomial time*

[16] BPP, RP and ZPP may also be defined using *probabilistic Turing machines*.

languages, denoted RP, consists of all languages for which there is a polynomial time Turing machine M and a polynomial p such that

$$x \in L \implies \Pr_{w \in \{0,1\}^{p(|x|)}}[M(x,w) = 1] \geq \frac{1}{2}$$
$$x \notin L \implies (\forall w \in \{0,1\}^{p(|x|)})\, M(x,w) = 0,$$

for every $x \in \{0,1\}^*$. The definition of RP remains unaffected if we choose any constant fraction greater than 0 instead of $\frac{1}{2}$. The class of *zero-error probabilistic polynomial time* languages, denoted ZPP, consists of all languages for which there is a polynomial time Turing machine M and a polynomial p such that

1. $M(x,w)$ halts $\implies M(x,w) = [x \in L]$
2. $\Pr_{w \in \{0,1\}^{p(|x|)}}[M(x,w) \text{ halts}] \geq \frac{1}{2}$

As before, the definition of ZPP remains unaffected if we choose any constant fraction greater than 0 instead of $\frac{1}{2}$ As shown by Gill [27], $\mathsf{ZPP} = \mathsf{RP} \cap \mathsf{coRP}$[17].

There are two notions of "probabilistic" NP languages, the classes MA and AM. The class of *Merlin-Arthur languages*[18], MA, consists of all languages L for which there is a polynomial time machine M and polynomials p, q such that

$$x \in L \implies (\exists y \in \{0,1\}^{p(|x|)}) \Pr_{z \in \{0,1\}^{q(|x|)}}[M(x,y,z) = 1] \geq \frac{2}{3}$$
$$x \notin L \implies (\forall y \in \{0,1\}^{p(|x|)}) \Pr_{z \in \{0,1\}^{q(|x|)}}[M(x,y,z) = 0] \geq \frac{2}{3}$$

The class of *Arthur-Merlin languages*, AM, consists of all languages L for which there is a polynomial time machine M and polynomials p, q such that

$$x \in L \implies \Pr_{z \in \{0,1\}^{q(|x|)}}[(\exists y \in \{0,1\}^{p(|x|)})\, M(x,y,z) = 1] \geq \frac{2}{3}$$
$$x \notin L \implies \Pr_{z \in \{0,1\}^{q(|x|)}}[(\exists y \in \{0,1\}^{p(|x|)})\, M(x,y,z) = 1] \leq \frac{1}{3}$$

Again, the choice of $\frac{2}{3}$ is arbitrary in the definitions of MA and AM.

The role of probabilistic classes has become increasingly central to complexity theory. The current opinion, based on the plausibility of languages with high circuit complexity, is that probabilistic computation does not significantly increase computational power. That is, the following conjectures are generally believed to be true.

[17] Recall that for any class C, $\mathsf{coC} = \{L \mid \overline{L} \in C\}$, the set of all languages whose complement is in C

[18] The nomenclature comes from an equivalent definition of MA and AM in terms of interactive proofs.

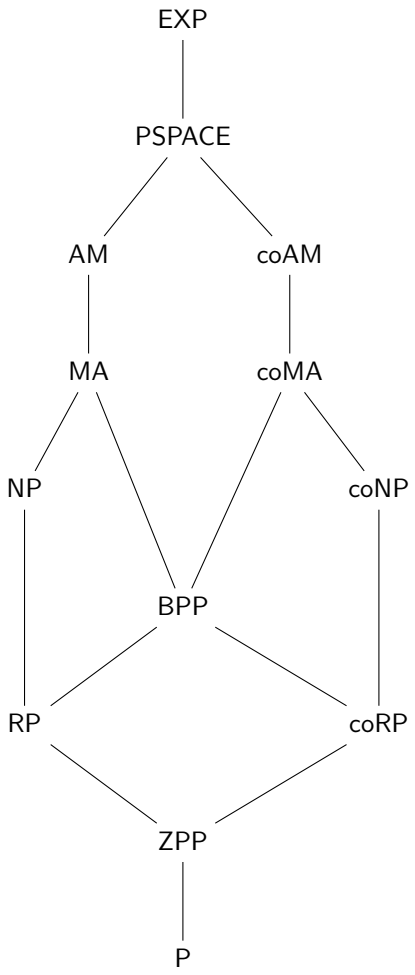


Figure 3: Hierarchy of complexity classes. It is widely conjectured that NP = MA = AM and P = ZPP = RP = BPP.

Conjecture 4.24.

$\mathsf{P} = \mathsf{ZPP} = \mathsf{RP} = \mathsf{BPP}$.

$\mathsf{NP} = \mathsf{MA} = \mathsf{AM}$.

Buhrman, Fenner and Fortnow [13] showed that, if $\mu_p(\mathsf{BPP}) \neq 0$, then $\mathsf{MA} = \mathsf{EXP}$. Subsequently, van Melkebeek [69] improved this theorem to show that, if BPP is not small, then it equals EXP.

Theorem 4.25. *Either* $\mu_{p_2}(\mathsf{BPP}) = 0$ *or* $\mathsf{BPP} = \mathsf{EXP}$.

With Lutz's formulation of resource bounded dimension, Moser [57] showed that van Melkebeeks' result can be strengthened to show that if BPP does not have *dimension* 0, then it equals EXP.

Theorem 4.26. *Either* $\dim_{p_2}(\mathsf{BPP}) = 0$ *or* $\mathsf{BPP} = \mathsf{EXP}$.

A natural question is whether these results can be adapted for the smaller probabilistic classes, such as RP and ZPP. Impagliazzo and Moser [36] showed that the analog of Theorem 4.25 does hold for RP.

Theorem 4.27. *Either* $\mu_{p_2}(\mathsf{RP}) = 0$ *or* $\mathsf{RP} = \mathsf{EXP}$.

As a corollary, Impagliazzo and Moser showed that Theorem 4.25 also holds for ZPP, and that the measure of ZPP is equal to the measure of RP.

Corollary 4.28. $\mu_{p_2}(\mathsf{ZPP}) = \mu_{p_2}(\mathsf{RP})$,

Proof. Since $\mathsf{ZPP} \subseteq \mathsf{RP}$, if $\mu_{p_2}(\mathsf{RP}) = 0$, then $\mu_{p_2}(\mathsf{ZPP}) = 0$. Otherwise, by Theorem 4.27, $\mathsf{RP} = \mathsf{EXP}$. Since EXP is closed under intersection and complementation, and $\mathsf{ZPP} = \mathsf{RP} \cap \mathsf{coRP}$, we see that $\mathsf{ZPP} = \mathsf{EXP}$. □

Unfortunately, it is still open whether Moser's zero-one result (Theorem 4.26) holds for smaller probabilistic classes. Such a result would not only be interesting in its own right, but it seems to require new tools.

Open Question 4.29. Does a dimension zero-one law hold for RP? For ZPP? Is it true that $\dim_p(\mathsf{RP}) = \dim_p(\mathsf{ZPP})$?

Recently, Fortnow, et al. [22] proved that a dimension zero-one law holds for all classes satisfying certain assumptions.

Theorem 4.30. *Let C be a class that is closed under exponential-time truth-table reductions. Then* $\mathrm{Dim}(C \mid \mathsf{ESPACE})$ *is either* 0 *or* 1.

As we have stated, it is widely conjectured that AM can be derandomized in NP. This conjecture is based on the assumption that there are languages in NE[19] with high circuit complexity. Impagliazzo and Moser [36] showed that this conjecture also follows if NP is not small.

Theorem 4.31. *If* $\mu_p(\mathsf{NP}) \neq 0$, *then* $\mathsf{NP} = \mathsf{AM}$.

[19] The nondetermnistic analog of E

4.5 Classes of Complete Languages

Mayordomo [52] showed that p-m-complete languages, although considered the "hardest" languages in a class, are small in the measure sense.

Theorem 4.32.

1. *The class of p-m-complete languages for* E *has measure* 0 *in* E.
2. *The class of p-m-complete languages for* NE *has p_2-measure* 0.

Ambos-Spies, et al. showed that, although the class of complete sets is small in the measure-theoretic sense, it is large in the dimensiona sense [4].

Theorem 4.33.
1. *The class of p-m-complete languages for* E *has dimension* 1 *in* E.
2. *The class of p-m-complete languages for* EXP *has dimension* 1 *in* EXP.

This theorem has several interesting consequences. The first is that there is a language which is complete for EXP which has p-dimension 1. This theorem also gives a natural class that is of p-measure 0, but of p-dimension 1. In the same paper, Ambos-Spies et al., [4] show that the class of complete languages for NP has the same dimension as NP. In particular, if NP does not have small dimension, then neither does the class of complete languages.

Theorem 4.34. *The class of p-m-complete languages for* NP *has the same dimension in* EXP *as* NP.

The next notion of reducibility that we will discuss is autoreducibility.

Definition 4.35. Let r be a notion of resource bounded reducibility; i.e. many-one, Turing, etc. A set A is *r-autoreducible* if there is an r-reduction that computes, for every x, the value $A(x)$ from the oracle A without querying A at x.

Many natural complete languages are autoreducible. Indeed, the NP-complete language SAT is tt-autoreducible with two queries. This fact has been exploited frequently in a variety of proofs[20]

As noted by Beigel, Fortnow and Stephan [7], Ambos-Spies et al., [4] implicitly showed that the class of many-one-, truth-table- and Turing-autoreducible sets in EXP have full p_2-dimension.

Theorem 4.36. *The classes of the many-one autoreducible, truth-table autoreducible and Turing autoreducible sets in* EXP *have p_2-dimension* 1.

In contrast, Beigel, Fortnow and Stephan [7] showed that the class of languages which are *not* infinitely-often autoreducible also has full p_2-dimension.

Theorem 4.37. *The class of all sets in* EXP *which are not infinitely-often autoreducible has p_2-dimension* 1.

[20] The entire first chapter of The Complexity Theory Companion by Hemaspaandra and Ogihara [30] is devoted to this technique.

4.6 Isomorphisms of Complete Languages

In this section, we review some of the results using resource bounded measure to study the Berman-Hartmanis conjecture.

Let r be a type of reduction. A function f is an *r-isomorphism* from a language A to a language B if f is an r-reduction from A to B, f is a bijection, and f^{-1} is an r-reduction.

Conjecture 4.38 (Berman-Hartmanis Isomorphism Conjecture)**.** *All polynomial-time* NP*-complete sets are polynomial-time isomorphic.*

The Berman-Hartmanis conjecture has been extensively studied. A positive proof of this conjecture would imply, for instance, that $\mathsf{P} \neq \mathsf{NP}$. While the prevailing opinion of the truth of this conjecture is mixed, all known NP many-one complete languages are $\leq_m^p$-isomorphic.

The Berman-Hartmanis conjecture is based on the following resource bounded effectivization of the Cantor's isomorphism theorem.

Theorem 4.39. *Let A and B be languages, and f, g be functions satisfying the following properties.*

1. *f is an $\leq_m^p$-reduction from A to B, and g is an $\leq_m^p$-reduction from B to A.*
2. *f and g are $1-1$.*
3. *f and g are length increasing; $|f(x)|, |g(x)| > |x|$ for all x.*
4. *f and g are polynomial time invertible. That is, there is a polynomial time machine M which, given x, outputs $f^{-1}(x)$ if it exists, and otherwise outputs a special symbol. Similarly, there is a machine for g.*

Then A and B are $\leq_m^p$-isomorphic.

Given the importance of length increasing reductions for the Berman-Hartmanis conjecture, it is natural to attempt to prove that all complete languages are complete via length increasing reductions. Hitchcock and Pavan [33] showed that if NP is not small, then every NP $\leq_m^p$-complete language is complete under length increasing $\leq_m^{p/poly}$ *reductions.* A $\leq_m^{p/poly}$-reduction is a many-one reduction f such that f is computed by a family of polynomial size circuits, one for each input length.

Theorem 4.40. *If $\mu_p(\mathsf{NP}) \neq 0$, then every* NP*-complete language is complete under length-increasing $\leq_m^{p/poly}$ reductions.*

Definition 4.41. Let f be a multi-valued function. We say that f is SNP-computable if there is a nondeterministic polynomial time machine M such that, for every x, every path of M on x outputs a member of $f(x)$ or outputs a special symbol $\perp$. We will require that at least one path of $M(x)$ outputs a member of $f(x)$.

Definition 4.42. Let A and B be languages. We say that A is *strong nondeterministic isomorphic* (SNP-isomorphic) to B if there is a function f (which may be multi-valued) such that the following hold.

1. A reduces to B via f.
2. B reduces to A via f^{-1}.
3. Both f and f^{-1} are SNP-computable.
4. There is a single valued refinement g of f that is an isomorphism from A to B.

Harkins, Hitchcock and Pavan [29] have recently shown that, if NP is not small, then all PSPACE-complete languages are SNP-isomorphic. Under the stronger assumption that $\mathsf{NP} \cap \mathsf{coNP}$, they show that the Berman-Hartmanis conjecture holds for NP under SNP-reductions.

Theorem 4.43.

1. *If* NP *contains a p-random language, then all polynomial-time complete sets for* PSPACE *are* SNP*-isomorphic.*
2. *If* $\mathsf{NP} \cap \mathsf{coNP}$ *contains a p-random language, then all polynomial-time complete sets for* NP *are* SNP*-isomorphic.*

4.7 Separating Completeness Notions

Lutz and Mayorodomo [50] separated Turing-completeness for many-one-completeness for NP under the assumption that NP is not small.

Theorem 4.44. *If* $\mu_p(\mathsf{NP}) \neq 0$*, then there is a problem that is* $\leq_T^p$*-complete for* NP *but not* $\leq_m^p$*-complete.*

Hitchcock and Pavan [33] strengthened Lutz and Mayordomo's theorem by, under the same assumption, separating Turing-completeness and truth-table-completeness for NP. In the same paper, they also separated SNP-completeness and Turing completeness for NP[21].

Theorem 4.45. *Assume that* $\mu_p(\mathsf{NP}) \neq 0$*. Then the following hold.*

1. *There is a problem that is* $\leq_T^p$*-complete for* NP *but not* $\leq_{tt}^p$*-complete.*
2. *There is a problem that is* $\leq_m^{SNP}$*-complete for* NP *but not* $\leq_T^p$*-complete.*

[21] They actually proved this using a weaker hypothesis: the NP *machine hypothesis*. This assumes that there is an $\epsilon > 0$ and a polynomial time nondeterministic machine M accepting the unary language 0^* such that no deterministic Turing machine running in time 2^{n^ϵ} can output an accepting path of M infinitely often. It is known that the $\mu_p(\mathsf{NP}) \neq 0$ implies the NP machine hypothesis.

Hitchcock and Shafei [34] recently proved separations of nonuniform reductions. A $p/1$-many-one reduction is a many-one reduction which is computable by a polynomial time Turing machine with one bit of *advice*. That is, f is a $p/1$-many-one reduction if f is a reduction, and there is a Turing machine M such that, for every $n \in \mathbb{N}$ and every $x \in \{0,1\}^n$, there is a bit $b \in \{0,1\}$ (the advice) such that

$$M(x,b) = f(x).$$

Note that the same bit must work for every string of a given length. However, the bit is allowed to change values at different input lengths.

Theorem 4.46 ([34]). *Assume*[22] *that* $\mu_p(\mathsf{NP}) \neq 0$. *Then the following hold.*

1. *There is a language* $A \in \mathsf{NP}$ *that is* NP*-complete with respect to* $\leq_m^{p/1}$*-reductions but is not* $\leq_m^p$*-complete.*

2. *There is a language* $A \in \mathsf{NP}$ *that is* NP*-complete with respect to* $\leq_m^{p/poly}$*-reductions but is not* $\leq_T^p$*-complete.*

5 Further Applications

5.1 Analysis

We now describe recent work connecting resource bounded randomness and classical analysis. We will therefore be computing over Euclidean space $\mathbb{R}^n$. We will encode real numbers in $[0,1]$ using their binary expansion, and say that a real number x is random (under some definition) if its binary expansion is random.

5.1.1 Computational Complexity in $\mathbb{R}^n$

We first give a brief overview of Ko's framework for complexity theory over $\mathbb{R}^n$. For more detail, his text [40] on the subject is an excellent reference.

A *dyadic rational number* d is a rational number that has a finite binary expansion; that is $d = \frac{m}{2^r}$ for integers m and r with $r \geq 0$. We denote the set of all dyadic rational numbers by $\mathbf{D}$. We denote the set of all dyadic rationals d of *precision* r by $\mathbf{D}_r$. Formally,

$$\mathbf{D}_r = \{\frac{m}{2^r} \mid m \in \mathbb{Z}\}.$$

We will be computing over dyadic rational numbers which are encoded as finite binary strings in the standard way.

A function $f : [0,1]^n \to \mathbb{R}$ is a *simple step function* if f is a step function such that

[22] Hitchcock and Shafei show that only the NP machine hypothesis is needed for the second item to hold.

1. $f(x) \in \mathbf{D}$ for all $x \in [0,1]^n$ and
2. there exists a finite number of (disjoint) dyadic boxes $Q_1, \ldots, Q_k$ and dyadic rationals $d_1, \ldots, d_k$ such that $f(x) = \sum_{i=1}^{k} d_i \chi_{Q_i}(x)$, where χ_Q is the characteristic function of a set Q.

A function $f : [0,1]^n \to \mathbb{R}$ is *polynomial time computable*[23] if there exists a sequence of simple step functions, $\{f_m\}_{m\in\mathbb{N}}$, and a polynomial p such that for all $d \in \mathbf{D}^n$,

1. for every m, $f_m(x) = \sum_{i=1}^{k} d_i \chi_{Q_i}(x)$, such that the endpoints of each Q_i are in $\mathbf{D}^n_{p(m)}$,
2. for every n, and every $d \in \mathbf{D}_{p(n)}$, $|f_n(d) - f_n(d + 2^{-p(n)})| \leq 2^{-n}$,
3. for every n and each $x \in [0,1]$, $|f_n(x) - f(x)| \leq 2^{-n}$,
4. there is a polynomial time TM M such that, for every n and $d \in \mathbf{D}_{p(n)}$,

$$M(d, 0^n) = f_n(d).$$

Note that if f is polynomial time computable, then f is continuous.

A function $f \in L_1([0,1]^n)$ is *polynomial time (space) L_1-computable* if there exists a sequence of simple step functions, $\{f_m\}_{m\in\mathbb{N}}$, and a polynomial p such that for all $d \in \mathbf{D}^n$,

1. for every m, $f_m(x) = \sum_{i=1}^{k} d_i \chi_{Q_i}(x)$, such that the endpoints of each Q_i are in $\mathbf{D}^n_{p(m)}$,
2. there is a polynomial time (resp. space) TM M computing f_m in the sense that
$$M(0^m, d) = \begin{cases} f_m(d) & \text{if } d \text{ is not a breakpoint of } f_m \\ \# & \text{otherwise} \end{cases}$$
3. $\|f - f_m\|_1 \leq 2^{-n}$.

Unlike polynomial time computable functions, polynomial space computable functions need not be continuous. Ko showed that a function being polynomial time L_1-computable does *not* imply that we can feasibly compute the integral $\int f d\mu$, under a reasonable complexity hypothesis.

Theorem 5.1. *The following are equivalent.*

[23]The most common definition of polynomial time computability uses oracle Turing machines, with oracles encoding *names* of a real number. We use the following, equivalent, definition to cut down on the number of preliminaries.

1. *For every f, if f is polynomial time L_1-computable, then $\int_0^1 f d\mu$ is polynomial time computable.*

2. $\mathsf{FP} = \#\mathsf{P}$.

The complexity class $\#\mathsf{P}$ is, informally, a functional analog of NP (a class of decision problems). $\mathsf{FP} \neq \#\mathsf{P}$ is a widely believed conjecture in complexity theory. Indeed, it follows (and is, in fact, weaker than) the famous $\mathsf{P} \neq \mathsf{NP}$ conjecture.

5.1.2 Randomness and Analysis

Recent research has revealed a strong connection between algorithmic randomness and measure theoretic analysis. The rise of measure theory in analysis has resulted in many theorems being "almost everywhere" results; that is, proving that a certain property holds except for a measure zero set. Various notions of randomness have been characterized as the set of all points satisfying an effectivized almost everywhere theorem [8, 24, 23, 71].

Of particular interest is the connection between randomness and differentiation theorems [12, 56, 60, 55]. Differentiation theorems, stating that a certain class of functions is differentiable almost everywhere, are fundamental in analysis. An important differentiation theorem of Lebesgue [42] states that if $f : [0,1] \to \mathbb{R}$ is a nondecreasing function, then f is differentiable almost everywhere. Brattka, Miller and Nies [12] used this theorem to characterize computable randomness.

Theorem 5.2. *A point $x \in [0,1]$ is computably random if and only if $f'(x)$ exists, for every computable nondecreasing function $: [0,1] \to \mathbb{R}$.*

In the past few years, this line of research has been pursued in the resource bounded setting. Although not as well studied as in the computable setting, a growing body of work suggests that there is a deep connection between resource bounded randomness and analysis.

Nies [59] proved that an analog of Theorem 5.2 holds for polynomial time randomness[24].

Theorem 5.3. *Let $x \in [0,1]$. Then x is polynomial time random if and only if $f'(x)$ exists for every nondecreasing polynomial time computable function $f : [0,1] \to \mathbb{R}$.*

Galicki [26] recently showed that Rademacher's theorem, another fundamental differentiation theorem, also characterizes polynomial time randomness.

Theorem 5.4. *If $f : \mathbb{R}^n \to \mathbb{R}$ is Lipschitz and polynomial time computable, then it is differentiable at every polynomial time random point $x \in R^n$.*

[24] Nies used a notion of polynomial time computable functions defined using Cauchy sequences. As the two notions of polynomial time computability are equivalent, we rephrase his proof in language of Ko's framework.

The final application we will highlight also concerns a classical theorem of Lebesgue [43].

Theorem 5.5 (Lebesgue Differentiation Theorem)**.** *Let $f : [0,1]^n \to \mathbb{R}$ be integrable. Then for almost every x,*

$$f(x) = \lim_{Q \to x} \frac{\int_Q f d\mu}{\mu(Q)},$$

where the limit is taken over all balls containing x as the diameter goes to 0.

The Lebesgue differentiation theorem can be thought of as an n-dimensional generalization of the fundamental theorem of calculus, except we only require f to be integrable instead of continuous. This weakening of the hypothesis results in the conclusion holds almost everywhere, instead of for every point.

Pathak, Rojas and Simpson used the Lebesgue differentiation theorem to characterize Schnorr randomness [60]. Huang and Stull [35] showed that this connection extends to the resource bounded setting by proving that the Lebesgue differentiation theorem characterizes weak polynomial space randomness.

Theorem 5.6. *Let $x \in [0,1]^n$. Then x is weakly polynomial space random if and only if for every polynomial space L_1-computable function $f : [0,1]^n \to \mathbb{R}$ and every polynomial space computable sequence $\{f_m\}$ approximating f*

$$\lim_{m \to \infty} f_m(x) = \lim_{Q \to x} \frac{\int_Q f d\mu}{\mu(Q)}. \tag{4}$$

Note that equation (4) is slightly different than that of Theorem 5.5. The left hand limit in (4) is a necessary inclusion. For any point x, it is easy to see that the function f_x defined by

$$f_x(y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise.} \end{cases}$$

is polynomial space L_1-computable, yet $f(x) \neq \lim_{Q \to x} \frac{\int_Q f d\mu}{\mu(Q)}$.

We should remark that one of the principal reasons for using polynomial space instead of polynomial time, is Ko's result on the complexity of integration (Theorem 5.1).

5.2 Borel Normality

Let $k \geq 2$. We denote the k-ary alphabet by $\Sigma_k = \{0, 1, \ldots, k-1\}$. Borel introduced the concept of a normal sequence [10]. A sequence $A \in \Sigma_k$ is *normal* if, for every $w \in \Sigma_k^*$,

$$\lim_{n \to \infty} \tfrac{1}{n} |\{i < n \,:\, A[i \ldots i + |w| - 1] = w\}| = 2^{-|w|}.$$

We note that Borel's original definition of normality was for real numbers. Borel defined a real number as *normal in base-k* if its base k expansion is normal in the sense presented here. Since its introduction, the concept of Borel normality has been extensively studied. A recent line of research has shown that normality is intimately connected with finite state dimension. We now highlight several applications of finite state dimension to the theory of normal sequences.

Schnorr and Stimm implicitly showed that finite state randomness characterizes normal sequences [64]. This was later explicitly proved by Bourke, Hitchcock and Vinodchandran, using an entropy characterization of finite state dimension [11].

Theorem 5.7. *A k-ary sequence $A \in \Sigma_k$ is normal if and only if A is finite state random; i.e.,* $\dim_{FS}(A) = 1$.

A natural question (for real numbers) is the closure properties of normality. Wall, in his PhD thesis [72], proved that base k normality is closed under addition and multiplication by rational numbers.

Theorem 5.8. *Let $k \geq 2$, $q \in \mathbb{Q}$ be nonzero and $\alpha \in \mathbb{R}$. If α is normal base k, then the sum $q + \alpha$ and the product $q\alpha$ are also normal base k.*

Doty, Lutz and Nandakumar [17] used finite state dimension to extend, and give a new proof of, Wall's theorem. Their proof relied on an entropy characterization of finite state dimension, established by Bourke et, al [11]. Using this entropy characterization, and Schur's result on the concavity of the entropy function [66], they proved the following.

Theorem 5.9. *For every integer $k \geq 2$, every nonzero $q \in \mathbb{Q}$ and every real α,*

$$\dim_{FS}(q + \alpha) = \dim_{FS}(q\alpha) = \dim_{FS}(\alpha),$$

and

$$\mathrm{Dim}_{FS}(q + \alpha) = \mathrm{Dim}_{FS}(q\alpha) = \mathrm{Dim}_{FS}(\alpha).$$

The final application we will highlight is on Copeland-Erdös sequences. Let $A \subseteq \mathbb{Z}^+$ be an infinite set of positive integers and $k \geq 2$. The *base-k Copeland-Erdös sequence of A*, $\mathrm{CE}_k(A)$, is the infinite sequence over alphabet Σ_k, formed by concatenating the base-k representations of the elements of A in order.

Example 5.10. The base-10 Copeland-Erdös sequences of the positive integers and prime numbers are:

1. $\mathrm{CE}_{10}(\mathbb{Z}^+) = 1234567891011\ldots$.
2. $\mathrm{CE}_{10}(\mathrm{PRIMES}) = 23571113171923\ldots$.

Definition 5.11. A set of positive integers $A \subseteq \mathbb{Z}^+$ satisfies the *Copeland-Erdös hypothesis* if, for every real $\alpha < 1$, for all sufficiently large $n \in \mathbb{N}$,

$$|A \cap \{1, \ldots, n-1\}| > n^\alpha.$$

Copeland and Erdös showed that this hypothesis is a sufficient condition for proving that $\mathrm{CE}_k(A)$ is normal [16].

Theorem 5.12. *Let $A \subseteq \mathbb{Z}^+$ be a set satisfying the Copeland-Erdös hypothesis. Then, for every $k \geq 2$, the sequence $CE_k(A)$ is normal over Σ_k.*

Gu, Lutz and Moser [28] used finite state dimension to extend Theorem 5.12. Their proof relies on relating finite state dimension to another quantity, the *zeta dimension* of a set $A \subseteq \mathbb{Z}^+$.

Definition 5.13. Let $A \subseteq \mathbb{Z}^+$. The *zeta-dimension of A* is

$$\mathrm{Dim}_\zeta(A) = \limsup_{n\to\infty} \frac{\log |A \cap \{1,\ldots,n\}|}{\log n}.$$

The *lower zeta-dimension of A* is

$$\dim_\zeta(A) = \liminf_{n\to\infty} \frac{\log |A \cap \{1,\ldots,n\}|}{\log n}.$$

It is clear that a set A satisfies the Copeland-Erdös hypothesis if and only if $\dim_\zeta(A) = 1$. Hence the following theorem of Gu, Lutz and Moser [28] implies Theorem 5.12.

Theorem 5.14. *For every infinite $A \subseteq \mathbb{Z}^+$ and $k \geq 2$,*

$$\dim_{FS}(CE_k(A)) \geq \dim_\zeta(A),$$

and

$$\mathrm{Dim}_{FS}(CE_k(A)) \geq \mathrm{Dim}_\zeta(A),$$

6 Conclusion

The field of resource bounded randomness has enjoyed significant growth in the past few decades. Many new notions of randomness have been introduced, some in the past few years. One of the goals of resource bounded randomness is to understand the relations between the various notions. In the polynomial space setting, we have a complete understanding of the relations. The relations of the polynomial space notions discussed in this survey are represented in Figure 1. The picture for the polynomial time setting, however, is much less understood. This is an important problem, and it seems quite difficult. Given that, if $\mathsf{P} = \mathsf{PSPACE}$, the picture for polynomial time randomness is equal to this figure, it is quite possible that an accurate picture of the polynomial time relations depends on complexity theoretic conjectures, such as $\mathsf{P} \neq \mathsf{PSPACE}$.

Another direction in resource bounded randomness is the exploration of notions based on concepts other than the martingale approach. We have seen, in Sections 3.2 and 3.3 two notions of randomness using null covers and Kolmogorov complexity. We believe that more definitions using both paradigms

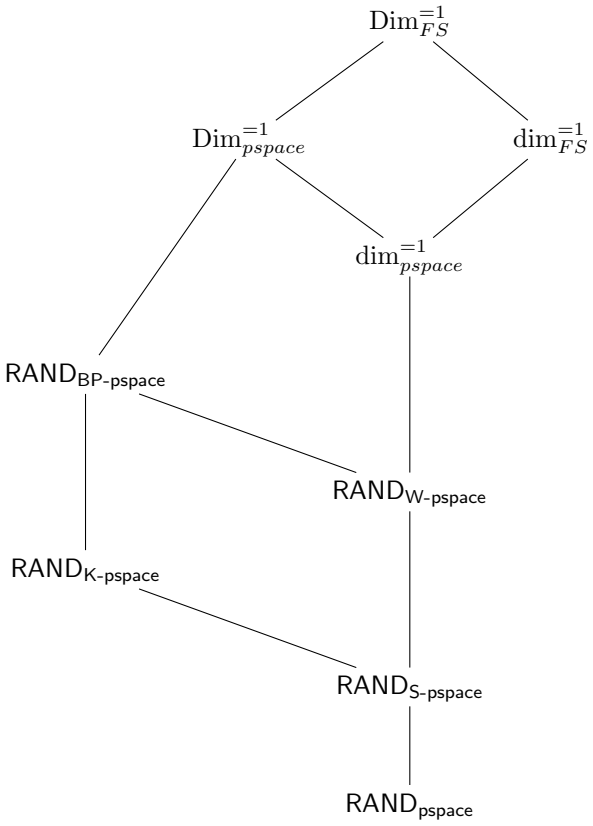


Figure 4: Hierarchy for polynomial space randomness. Lines represent strict inclusions.

would be beneficial to resource bounded randomness. As discussed in the introduction, compressibility (Kolmogorov complexity), "statistical tests" (null covers) and betting strategies (martingales) all give different intuitions, each useful under different circumstances. Defining notions using different paradigms will also help in determining which notions are fundamental.

Part of any successful field is its applications to other areas of mathematics. To date, the most numerous, and successful, applications of resource bounded randomness are to complexity theory. However, as we have seen, resource bounded randomness can also be successfully applied to analysis and number theory. We believe that more work in these areas will benefit both resource bounded randomness and the fields to which they are applied. For example, as mentioned in the Introduction, an important problem is to find the resource bounded analogs of the most common notions of randomness in the computable setting. As we have seen throughout this survey, there are typically multiple ways to extend a notion of randomness to the resource bounded setting. We will need evidence that a particular analog is the most natural. A very interesting approach to this problem is to use the theorems characterizing randomness using analysis as described in Section 5.1.2. It seems likely, given the theorem of Nies, Theorem 5.3, and the strength of its applications to complexity theory, that polynomial time and space randomness is the correct analog of computable randomness. The theorem of Huang and Stull, Theorem 5.6, gives evidence that weak polynomial space randomness might be the correct analog of Schnorr randomness. However, as of this time, we need more evidence in order to choose this notion as the "correct" resource bounded analog (as opposed to, say, polynomial space Schnorr randomness as described in Section 3.1.1). With this caution in mind, it is the opinion of the author that we do now have the correct analogs in the polynomial space setting. I believe we should use the following as a "dictionary" to pass between the resource bounded and non-resource bounded domains.

1. Computable randomness $\equiv$ polynomial space randomness.
2. Schnorr randomness $\equiv$ weak polynomial space randomness.
3. Kurtz randomness $\equiv$ polynomial space BP randomness.

In the polynomial time setting, it is still very much an open question to provide the correct analogs.

Along these lines, a very interesting question is the following.

Open Question 6.1. What is the resource bounded analog of Martin-Löf randomness?

While we have tried to give a broad overview of resource bounded randomness, there are many notions we were unable to cover. These include the notions of resource bounded genericity of Ambos-Spies, Fleischhack and Huwig [1], and resource bounded stochasticity of Ambos-Spies, Mayordomo, Wang and Zheng [3].

References

[1] Ambos-Spies, K., Fleischhack, H., and Huwig, H. Diagonalizations over polynomial time computable sets. *Theor. Comput. Sci. 51* (1987), 177–204.

[2] Ambos-Spies, K., and Mayordomo, E. Resource-bounded measure and randomness. *Lecture Notes in Pure and Applied Mathematics* (1997), 1–48.

[3] Ambos-Spies, K., Mayordomo, E., Wang, Y., and Zheng, X. Resource-bounded balanced genericity, stochasticity and weak randomness. In *STACS 96, 13th Annual Symposium on Theoretical Aspects of Computer Science, Grenoble, France, February 22-24, 1996, Proceedings* (1996), pp. 63–74.

[4] Ambos-Spies, K., Merkle, W., Reimann, J., and Stephan, F. Hausdorff dimension in exponential time. In *Proceedings of the 16th Annual IEEE Conference on Computational Complexity, Chicago, Illinois, USA, June 18-21, 2001* (2001), pp. 210–217.

[5] Ambos-Spies, K., Terwijn, S., and Zheng, X. Resource bounded randomness and weakly complete problems. *Theor. Comput. Sci. 172*, 1-2 (1997), 195–207.

[6] Athreya, K. B., Hitchcock, J. M., Lutz, J. H., and Mayordomo, E. Effective strong dimension in algorithmic information and computational complexity. *SIAM J. Comput. 37*, 3 (2007), 671–705.

[7] Beigel, R., Fortnow, L., and Stephan, F. Infinitely-often autoreducible sets. *SIAM J. Comput. 36*, 3 (2006), 595–608.

[8] Bienvenu, L., Day, A. R., Hoyrup, M., Mezhirov, I., and Shen, A. A constructive version of Birkhoff's ergodic theorem for Martin-Löf random points. *Inform. and Comput. 210* (2012), 21–30.

[9] Bienvenu, L., Shafer, G., and Shen, A. On the history of martingales in the study of randomness. *J. Électron. Hist. Probab. Stat. 5*, 1 (2009), 40.

[10] Borel, M. É. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti del Circolo Matematico di Palermo (1884-1940) 27*, 1 (1909), 247–271.

[11] Bourke, C., Hitchcock, J. M., and Vinodchandran, N. V. Entropy rates and finite-state dimension. *Theor. Comput. Sci. 349*, 3 (2005), 392–406.

[12] Brattka, V., Miller, J. S., and Nies, A. Randomness and differentiability. *Trans. Amer. Math. Soc. 368*, 1 (2016), 581–605.

[13] Buhrman, H., Fenner, S. A., and Fortnow, L. Results on resource-bounded measure. In *Automata, Languages and Programming, 24th International Colloquium, ICALP'97, Bologna, Italy, 7-11 July 1997, Proceedings* (1997), pp. 188–194.

[14] Buss, S., Cenzer, D., and Remmel, J. B. Sub-computable boundedness randomness. *Logical Methods in Computer Science 10*, 4 (2014).

[15] Chaitin, G. J. A theory of program size formally identical to information theory. *Journal of the ACM (JACM) 22*, 3 (1975), 329–340.

[16] Copeland, A. H., and Erdös, P. Note on normal numbers. *Bull. Amer. Math. Soc. 52* (1946), 857–860.

[17] Doty, D., Lutz, J. H., and Nandakumar, S. Finite-state dimension and real arithmetic. *Inf. Comput. 205*, 11 (2007), 1640–1651.

[18] Downey, R., and Hirschfeldt, D. *Algorithmic Randomness and Complexity*. Springer-Verlag, 2010.

[19] Downey, R. G., and Griffiths, E. J. Schnorr randomness. *J. Symb. Log. 69*, 2 (2004), 533–554.

[20] Falconer, K. *Fractal geometry*, third ed. John Wiley & Sons, Ltd., Chichester, 2014. Mathematical foundations and applications.

[21] Feder, M. Gambling using a finite state machine. *IEEE Transactions on Information Theory 37*, 5 (1991), 1459–1465.

[22] Fortnow, L., Hitchcock, J. M., Pavan, A., Vinodchandran, N. V., and Wang, F. Extracting kolmogorov complexity with applications to dimension zero-one laws. *Inf. Comput. 209*, 4 (2011), 627–636.

[23] Franklin, J. N. Y., Greenberg, N., Miller, J. S., and Ng, K. M. Martin-Löf random points satisfy Birkhoff's ergodic theorem for effectively closed sets. *Proc. Amer. Math. Soc. 140*, 10 (2012), 3623–3628.

[24] Franklin, J. N. Y., and Towsner, H. Randomness and non-ergodic systems. *Mosc. Math. J. 14*, 4 (2014), 711–744, 827.

[25] Gács, P. Exact expressions for some randomness tests. In *Theoretical Computer Science, 4th GI-Conference, Aachen, Germany, March 26-28, 1979, Proceedings* (1979), pp. 124–131.

[26] Galicki, A. Polynomial-time rademacher theorem, porosity and randomness. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland* (2017), pp. 30:1–30:13.

[27] Gill, J. Computational complexity of probabilistic turing machines. *SIAM Journal on Computing 6*, 4 (1977), 675–695.

[28] Gu, X., Lutz, J. H., and Moser, P. Dimensions of copeland-erdös sequences. *Inf. Comput. 205*, 9 (2007), 1317–1333.

[29] Harkins, R. C., Hitchcock, J. M., and Pavan, A. Strong reductions and isomorphism of complete sets. *Computability 3*, 2 (2014), 91–104.

[30] Hemaspaandra, L. A., and Ogihara, M. *The Complexity Theory Companion.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2002.

[31] Hitchcock, J. M. Effective fractal dimension: foundations and applications.

[32] Hitchcock, J. M., Lutz, J. H., and Mayordomo, E. The fractal geometry of complexity classes. *SIGACT News 36*, 3 (2005), 24–38.

[33] Hitchcock, J. M., and Pavan, A. Comparing reductions to np-complete sets. *Inf. Comput. 205*, 5 (2007), 694–706.

[34] Hitchcock, J. M., and Shafei, H. Nonuniform reductions and np-completeness. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France* (2018), pp. 40:1–40:13.

[35] Huang, X., and Stull, D. M. Polynomial space randomness in analysis. In *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland* (2016), pp. 86:1–86:13.

[36] Impagliazzo, R., and Moser, P. A zero-one law for RP and derandomization of AM if NP is not small. *Inf. Comput. 207*, 7 (2009), 787–792.

[37] Juedes, D. W., and Lutz, J. H. The complexity and distribution of hard problems. *SIAM J. Comput. 24*, 2 (1995), 279–295.

[38] Juedes, D. W., and Lutz, J. H. Modeling time-bounded prefix kolmogorov complexity. *Theory Comput. Syst. 33*, 2 (2000), 111–123.

[39] Ko, K. On the notion of infinite pseudorandom sequences. *Theor. Comput. Sci. 48*, 3 (1986), 9–33.

[40] Ko, K.-I. *Computational complexity of real functions.* Springer, 1991.

[41] Kurtz, S. A. Randomness and genericity in the degrees of unsolvability.

[42] Lebesgue, H. Sur les intégrales singulières. *Annales de la Faculté des sciences de Toulouse: Mathématiques 3* (1909), 25–117.

[43] Lebesgue, H. Sur l'intégration des fonctions discontinues. 361–450.

[44] Levin, L. A. Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. *Problemy Peredachi Informatsii 10*, 3 (1974), 30–35.

[45] Li, M., and Vitányi, P. M. B. *An Introduction to Kolmogorov Complexity and Its Applications, Third Edition*. Texts in Computer Science. Springer, 2008.

[46] Lutz, J. H. Almost everywhere high nonuniform complexity. *J. Comput. Syst. Sci. 44*, 2 (1992), 220–258.

[47] Lutz, J. H. The quantitative structure of exponential time. In *Proceedings of the Eigth Annual Structure in Complexity Theory Conference, San Diego, CA, USA, May 18-21, 1993* (1993), pp. 158–175.

[48] Lutz, J. H. Dimension in complexity classes. *SIAM J. Comput. 32*, 5 (2003), 1236–1259.

[49] Lutz, J. H. The dimensions of individual strings and sequences. *Inf. Comput. 187*, 1 (2003), 49–79.

[50] Lutz, J. H., and Mayordomo, E. Cook versus karp-levin: Separating completeness notions if NP is not small. *Theor. Comput. Sci. 164*, 1&2 (1996), 141–163.

[51] Martin-Löf, P. The definition of random sequences. *Information and control 9*, 6 (1966), 602–619.

[52] Mayordomo, E. Almost every set in exponential time is p-bi-immune. *Theor. Comput. Sci. 136*, 2 (1994), 487–506.

[53] Mayordomo, E. A kolmogorov complexity characterization of constructive hausdorff dimension. *Inf. Process. Lett. 84*, 1 (2002), 1–3.

[54] Mayordomo, E. Effective fractal dimension in algorithmic information theory. *New Computational Paradigms: Changing Conceptions of What is Computable* (2008), 259–285.

[55] Miyabe, K. Characterization of Kurtz randomness by a differentiation theorem. *Theory Comput. Syst. 52*, 1 (2013), 113–132.

[56] Miyabe, K., Nies, A., and Zhang, J. Using almost-everywhere theorems from analysis to study randomness. *Bull. Symb. Log. 22*, 3 (2016), 305–331.

[57] Moser, P. A zero-one subexp-dimension law for BPP. *Inf. Process. Lett. 111*, 9 (2011), 429–432.

[58] Nies, A. *Computability and Randomness*. Oxford University Press, Inc., New York, NY, USA, 2009.

[59] Nies, A. Differentiability of polynomial time computable functions. In *31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), STACS 2014, March 5-8, 2014, Lyon, France* (2014), pp. 602–613.

[60] Pathak, N., Rojas, C., and Simpson, S. G. Schnorr randomness and the Lebesgue differentiation theorem. *Proc. Amer. Math. Soc. 142*, 1 (2014), 335–349.

[61] Regan, K. W., Sivakumar, D., and Cai, J. Pseudorandom generators, measure theory, and natural proofs. In *36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, 23-25 October 1995* (1995), pp. 26–35.

[62] Ryabko, B. Y. The complexity and effectiveness of prediction algorithms. *J. Complexity 10*, 3 (1994), 281–295.

[63] Schnorr, C. A unified approach to the definition of random sequences. *Mathematical Systems Theory 5*, 3 (1971), 246–258.

[64] Schnorr, C., and Stimm, H. Endliche automaten und zufallsfolgen. *Acta Inf. 1* (1972), 345–359.

[65] Schnorr, C. P. *Zufälligkeit und Wahrscheinlichkeit*, vol. 218. Springer-Verlag, 1971.

[66] Schur, I. Uber eine klasse von mittelbildungen mit anwendungen auf die determinantentheorie. *Sitzungsberichte der Berliner Mathematischen Gesellschaft 22* (1923), 9–20.

[67] Staiger, L. A tight upper bound on Kolmogorov complexity and uniformly optimal prediction. *Theory Comput. Syst. 31*, 3 (1998), 215–229.

[68] Sureson, C. Subcomputable schnorr randomness. *Logical Methods in Computer Science 13*, 2 (2017).

[69] van Melkebeek, D. The zero-one law holds for BPP. *Theor. Comput. Sci. 244*, 1-2 (2000), 283–288.

[70] Ville, J. *Etude critique de la notion de collectif.* Gauthier-Villars Paris, 1939.

[71] Vyugin, V. V. Ergodic theorems for individual random sequences. *Theoret. Comput. Sci. 207*, 2 (1998), 343–361.

[72] Wall, D. D. *Normal Numbers.* PhD thesis, University of California, Berkeley, 1949.

[73] Wang, Y. *Randomness and complexity.* PhD thesis, Verlag nicht ermittelbar, 1996.

[74] Wang, Y. Resource bounded randomness and computational complexity. *Theor. Comput. Sci. 237*, 1-2 (2000), 33–55.