Algorithmic Randomness

Rod Downey
School of Mathematics and Statistics
Victoria University
Wellington, New Zealand
rod.downey@vuw.ac.nz

Denis R. Hirschfeldt
Department of Mathematics
The University of Chicago
Chicago, Illinois
drh@math.uchicago.edu

ACM Reference Format:
Rod Downey and Denis R. Hirschfeldt. 2018. Algorithmic Randomness. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Classical probability theory gives all sequences of fair coin tosses of the same length the same probability. On the other hand, when considering sequences such as

$$10101010101010101010101010101010\ldots$$

and

$$10110101110101011110000101010001011 1\ldots,$$

none but the most contrarian among us would deny that the second (obtained by the first author by tossing a coin) is more random than the first. Indeed, we might well want to say that the second sequence is entirely random, while the first one is entirely nonrandom. But what are we to make in this context of, say, the sequence obtained by taking our first sequence, tossing a coin for each bit, and if the coin comes up heads, replacing that bit by the corresponding one in the second sequence? There are deep and fundamental questions involved in trying to understand why some sequences should count as "random", or "partially random", and others as "lawful", and how we can transform our intuitions about these concepts into meaningful mathematical notions.

One goal of the theory of algorithmic randomness is to give meaning to the notion of a random *individual* (infinite) sequence. Questions immediately arise: How should we try to define randomness? How can we measure whether one sequence is more random than another? How are computational power and randomness related? Is a theory of randomness for individual sequences at all useful? How does such a theory relate to classical probability theory?

The modern development of the theory of algorithmic randomness goes back to the 1960s (with even earlier roots, as we will discuss), but there has been a particular surge of development in the last couple of decades. In this article, we hope to give some of the flavor of this work, though we will of course be able to mention only a few samples of what is by now a vast area of research. Our book [10], for example, is over 800 pages long and still manages to cover only a fraction of the area… Another book covering some of the work we discuss here is Nies [33].

For simplicity, we assume all sequences are binary unless we say otherwise. We use the terms *sequence* to mean an infinite sequence and *string* to mean a finite one. We write $|\sigma|$ for the length of a string σ, write $X(n)$ for the nth bit of the sequence X (beginning with the 0th bit $X(0)$), and write $X \upharpoonright n$ for the string consisting of the first n bits of X. We identify a real number x in the interval $[0, 1]$ with its binary expansion, i.e., the sequence X such that $x = 0.X$. There are reals that have two binary expansions, but they are all rational and will not be relevant here.

2 HISTORICAL ROOTS

2.1 Borel

In the beginning of the 20th century, Émile Borel was interested in sequences that satisfy the *law of large numbers*, which says that if we repeat an experiment with a numerical result many times, the average value of the result should be close to its expected value. If we toss a fair coin many times, for example, we expect the frequency of heads to be about $\frac{1}{2}$. Let X be a sequence representing infinitely many such tosses. After s many coin tosses, we can see how we are doing so far by looking at how many heads we have seen in the first s tosses compared to s, i.e., the ratio

$$\frac{|\{X(k) = 1 \mid k < s\}|}{s},$$

where we think of a 1 as representing heads. If this is indeed a fair coin, this ratio should get closer and closer to $\frac{1}{2}$ as s increases.

More generally, we say that an n-ary sequence sequence X is *(Borel) normal* if it has the same property relative to an "n-sided coin", in other words, if for any digit between 0 and $n - 1$,

$$\lim_{s\to\infty} \frac{|\{X(k) = i \mid k < s\}|}{s} = \frac{1}{n}.$$

Borel defined a real number to be *normal to base n* if its base n representation is normal, and *absolutely normal* if it is normal to every base. He observed that almost every real number is absolutely normal. Mathematically, this fact can be expressed by saying that the collection of absolutely normal numbers has Lebesgue measure 1, which corresponds to saying that if we threw a dart at the real line, with probability 1 it would hit an absolutely normal number. We would thus expect a random sequence to be normal, and indeed (recalling that we identify the sequence X with the real number $0.X$) we would expect a random sequence to be absolutely normal.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2.2 Von Mises and Ville

The late 1920s and early 1930s saw the development, particularly by Kolmogorov, of an adequate foundation for probability theory, using measure theory and based on the idea of the expected behavior of events in a probability space. This theory does not give any meaning to the idea of randomness of an individual object, such as a particular sequence of coin tosses. Tossing a fair coin n times takes place in a "space of possibilities" (in this case, the collection of all binary strings of length n), and we assign any sequence of length n the probability 2^{-n} of occurring. For example, as we are taught in school, any particular sequence of three coin tosses occurs with probability $2^{-3} = \frac{1}{8}$.

In the infinite case, we might look at the event that a sequence has a certain string, say 101, as an initial segment. The probability that we begin a sequence of coin tosses with heads, tails, heads is $2^{-3} = \frac{1}{8}$. The mathematical way to express this fact is that the (*uniform*) *measure* (also known as the *Lebesgue measure*) of the collection of all sequences beginning with 101 is 2^{-3}, or, more generally, the measure of the set of sequences beginning with any particular string of length n is 2^{-n}. Probability theory is of course a vast and complex field, but for our purposes this simple example suffices.

It is less commonly known that Kolmogorov's work came after earlier attempts to give meaning to the notion of randomness for individual objects such as infinite sequences. This idea is completely contrary to the approach in which all sequences are equally likely, but is quite reasonable when thinking about the difference between sequences like the two that open this article. The question is how to differentiate between a sequence like 010101 . . ., which is clearly nonrandom, and one arising from a random source. There are certain tests we can clearly apply to a sequence to try to verify its apparent randomness. For example, a random sequence should be normal in the sense of the previous section. However, that is clearly not a sufficient condition, since the sequence 010101 . . . is in fact normal.

In 1919, Richard von Mises[1] attempted to give a definition of randomness for a sequence X based upon a generalization of the law of large numbers. His idea was to require normality not only of X itself, but also of (certain) infinite subsequences of X. The point here is that the sequence 010101 . . . is normal, but if we select every other bit of this sequence, the resulting subsequence 0000 . . . is no longer normal. It is not reasonable that selecting every other bit of a random sequence should result in all 0's, so our sequence fails this randomness test.

Von Mises generalized this idea as follows. Let $f : \mathbb{N} \to \mathbb{N}$ be an increasing function. We think of f as a *selection function* for determining a subsequence of a given sequence. That is, $f(i)$ is the ith place selected in forming this subsequence. In the law of large numbers itself, where we consider the entire sequence, $f(i) = i$. In the nonrandomness argument in the previous paragraph, $f(i) = 2i$. Von Mises proposed replacing the ratio

$$\frac{|\{X(k) = 1 \mid k < s\}|}{s}$$

used in the law of large numbers by

$$\frac{|\{X(f(k)) = 1 \mid k < s\}|}{s},$$

the ratio of the number of *selected places* at which X has value 1 to the total number of selected places. For each choice of f, the requirement that this ratio approach $\frac{1}{2}$ as s goes to infinity constitutes a randomness test.

So when should X be regarded as random? We could perhaps try to say that X is random if and only if it passes this test for all possible selection functions, reflecting the idea that in a sequence of coin tosses, there should be no way to select a subsequence ahead of time that will have a greater proportion of heads than tails. There is a big problem with this idea, though. No sequence X can be random for *all* selection functions. Since any nontrivial X has infinitely many 0's, there is an f that chooses the positions of the 0's of X in increasing order. But surely this counterexample is unfair to the spirit of von Mises' idea: we are trying to capture the notion that we should not be able to *predict* the values of bits of X, and this f is chosen *after* defining X. It is always easy to predict the answer if you know it in advance! The question then is what kinds of selection functions should be allowed, to capture the notion of prediction. A reasonable intuition is that prediction is somehow a computational process, and hence from a modern perspective we might want to restrict ourselves to *computable* selection functions, a suggestion later made by Church.

Von Mises' work predated the definition of computable function, however, so he had no canonical choice of "acceptable selection rules" and left his definition mathematically vague. But Wald showed that for any countably infinite collection of selection functions, there is a sequence that is random in the sense of passing all tests corresponding to the functions in this collection.

However, von Mises' program was dealt a major blow in 1939 by Ville, who showed that for any countable collection of selection functions, there is a sequence X that passes all of the resulting tests, but such that for each n, there are always more 0's than 1's in $X \upharpoonright n$. If we were told that there would always be more tails than heads in a sequence of coin flips, we would not believe the coin to be a standard fair coin, and could use this information to make some money betting on its flips. Thus Ville's sequence is random in the sense of von Mises, but certainly not random in the intuitive sense.

Ville suggested adding versions of another law (the law of the iterated logarithm) to the list of tests that a sequence would need to pass to be considered random. Perhaps von Mises' tests together with these additional tests would capture the notion of algorithmic randomness. But this all begins to look very ad hoc, and immediately raises the natural question of whether there is a Ville-like counterexample for this new set of laws. (As it turns out, there is, as discussed e.g. in [10].)

Notice that in these discussions we are abandoning the idea of *absolute randomness* in some metaphysical sense in favor of a notion of *algorithmic randomness*, where we use tools from computability theory to define and quantify randomness. Abandoning absolute randomness leads to the idea of "levels of randomness" that can be defined by calibrating the computability-theoretic complexity of the tests we require our random sequences to pass. But, of course,

[1] See [10] for references to this and other sources mentioned in this section.

following Ville's work it was not clear that even one reasonably robust level of algorithmic randomness could be defined.

2.3 Martin-Löf

This is how matters stood until 1966 and the work of Per Martin-Löf, who effectivized the notion of null set from classical measure theory and gave a satisfying definition of algorithmic randomness based on this effectivization. The basic idea is that a random sequence should not have any "rare" property, i.e., that if we find a way to distinguish and describe a small collection of sequences, then no random sequence should be in our collection. The notion of null set allows us to make precise what we mean by "small".

Randomness tests like the ones suggested by von Mises are computable ways of narrowing down the set of sequences that can be considered to be random. For example, consider sequences like $0101\ldots$ that have 0's in all even places. We do not want any such "bad" sequence to be consider random. To test whether a sequence is of this form, we can take a "level-by-level" approach: Given a sequence X, we first ask whether $X(0) = 0$. If so, then X fails the first level of our test. (That is, X has failed to demonstrate so far that it is not one of our bad sequences.) Notice that exactly half of all sequences X have $X(0) = 0$, which can be formalized by saying that the set of sequences X such that $X(0) = 0$ has measure $\frac{1}{2}$.

Next, we ask whether $X(0) = 0$ *and* $X(2) = 0$. If so, then X fails the second level of our test. The proportion of all sequences X that fail this second level is $\frac{1}{4}$ (corresponding to the fact that exactly $\frac{1}{4}$ of all strings of length 3 have 0's at positions 0 and 2). We continue in this fashion, testing more and more even places. A sequence X is one of our bad sequences if and only if it fails *all* levels of our test. The fact that the set T_n of sequences that fail the nth level of our test has measure 2^{-n} implies that the set of bad sequences, which is the intersection of all the T_n's, has measure 0, i.e., that it is what we call a *null set*.

Martin-Löf's approach was to generalize this process by considering all possible level-by-level procedures for testing randomness. We can think of such a procedure as being generated by a machine M. At each level n, this machine determines a set T_n of sequences that are deemed to have failed the test so far. It does so by enumerating strings $\sigma_0^n, \sigma_1^n, \ldots$, where we then let T_n be the collection of all sequences that begin with some σ_i^n. Of course, M needs to be fair and not, say, consider all sequences to be nonrandom, so we insist that, like in the above example, T_n contain at most a proportion 2^{-n} of all sequences (which we can formalize by saying that the measure of T_n is at most 2^{-n}). Now a sequence X *fails* M's test if it is contained in every T_n, and otherwise it *passes* this test.

We say that a sequence is *Martin-Löf random* if and only if it pass *all* such tests.[2] It can be shown that almost all sequences are Martin-Löf random (i.e., that the collection of Martin-Löf random sequences has measure 1). Furthermore, Martin-Löf's notion of tests includes the ones proposed by von Mises (in the specific realization suggested by Church), the ones proposed by Ville, and indeed all "algorithmically performable" randomness tests. Thus the objection to the idea of adding more and more specific tests as we uncover more and more Ville-like sequences is neatly circumvented.

As it turns out, Martin-Löf randomness is also quite well-behaved mathematically, and has provided a robust basis for the theory of algorithmic randomness. As Jack Lutz put it in a lecture at the *7th Conference on Computability, Complexity, and Randomness*, held in Cambridge in 2012 (in connection with work of Turing that we will discuss in Section 3.3),

> Placing computability constraints on a nonconstructive theory like Lebesgue measure seems a priori to weaken the theory, but it may strengthen the theory for some purposes. This vision is crucial for present-day investigations of individual random sequences, dimensions of individual sequences, measure and category in complexity classes, etc.

In summary, Martin-Löf reformulated all the laws that we would expect a random sequence to obey at an abstract level, based upon the idea of effectivizing measure theory. The measure of a set of sequences is the mathematical version of the probability that a sequence is in this set. Martin-Löf randomness says that we regard X as random if and only if it passes each effective test that determines a set of effective measure 0 (as the intersection of the levels of the test). Such an X has every property that we can algorithmically describe as a set of probability 1.

2.4 Solomonoff, Kolmogorov, Levin, Chaitin, and Schnorr

There are other approaches to a definition of algorithmic randomness. For (finite) strings, a suitable definition was formulated by Kolmogorov, who argued that if a string has identifiable regularities, then we should be able to compress it, and that a compressible string should not be thought of as random. Here we think of a machine M as a descriptional process. If an input τ is processed by M to yield an output σ, then τ is a description of σ, i.e., a program that M can use to print σ. A random σ should have no short descriptions.

As an illustration, consider the sequence $\sigma = 010101010\ldots$ (1000 times). A short description τ of σ is "print 01 1000 times". This brief program produces an output of length 2000. We are exploiting the regularities of this particular σ to compress it into a short description. Kolmogorov's intuition was that for a random sequence there should be no regularities, so that the only way to describe σ is to essentially use σ itself. More precisely, a string of length n should be random relative to some descriptional process if and only if its shortest description has length n. Like white noise, a random string should be *incompressible*.

To give a physical analog of this idea, suppose that we have a maze shaped like a binary tree of height 6, with boxes at the end. That is, there are 2^6 possible routes to get to the boxes. One of the boxes has money in it, and someone is to tell us which. If the box is the leftmost one, all they have to say is "always turn left". If the box is to be found by say, left-right-left etc., this path is again easy to describe. If the place of the prize is determined randomly, though, the person would likely need to tell us the whole

[2] Formally, a *Martin-Löf test* is a collection $S_0, S_1, \ldots$ of uniformly computably enumerable sets of strings such that, if we let T_n be set of all sequences that begin with some element of S_n, then T_n has measure at most 2^{-n}. (The notion of computable enumerability is also known as recursive enumerability.) A sequence X *passes* this test if $X \notin \bigcap_n T_n$. A sequence is *Martin-Löf random* if it passes all Martin-Löf tests.

sequence of turns.[3] This compressibility approach gives rise to what is now called *Kolmogorov complexity*. For a Turing machine M, the Kolmogorov complexity $C_M(\sigma)$ of σ relative to M is the length of the shortest program τ such that $M(\tau) = \sigma$. We can then take a universal Turing machine U, which can emulate any other given machine M with at most a constant increase in the size of programs, and define the (plain) Kolmogorov complexity of σ to be $C(\sigma) = C_U(\sigma)$.

A natural guess is that a sequence X is random if and only if for all n, the first n bits of X are incompressible in the sense outlined above. As it turns out, however, plain Kolmogorov complexity is not quite the correct notion for infinite sequences. (The reason is that in the above account, M can use more than just the bits of τ to generate σ. It can also use the length of τ, which provides an additional $\log|\tau|$ many bits of information. Using this idea, Martin-Löf showed that for any X, the plain Kolmogorov complexity of $X \upharpoonright n$ must always go significantly below n for some lengths n.)

There are several ways to modify the definition of Kolmogorov complexity to avoid this issue, the best-known being to use prefix-free codes[4] and the resulting notion of *prefix-free Kolmogorov complexity*, denoted by K in place of C, as in the work of Levin, Chaitin, and Schnorr, and in a certain sense even earlier in that of Solomonoff. As shown by Schnorr, it is indeed the case that X is Martin-Löf random if and only if the prefix-free Kolmogorov complexity of the first n bits of X is at least n (up to a constant factor), that is, $K(X \upharpoonright n) \geqslant n - O(1)$.

(There are many other flavors of Kolmogorov complexity, including time- and space-bounded ones, but C and K have been the most studied. They have a complex relationship. It is easy to show that $K(\sigma) \leqslant C(\sigma) + 2\log|\sigma| + O(1)$. Solovay proved the remarkable fact that $K(\sigma) = C(\sigma) + C(C(\sigma)) + O(C^{(3)}(\sigma))$ and this result is tight in that we cannot extend it to $C^{(4)}(\sigma)$. There is a huge amount of research on the Kolmogorov complexity of finite strings and its applications. See for instance Li and Vitanyi [25].)

Returning to the story of the definition of algorithmic randomness, there is another approach, developed by Schnorr, that is close in spirit to von Mises' ideas. A *martingale*[5] is a function d from strings to nonnegative reals satisfying a fairness condition

$$d(\sigma) = \frac{d(\sigma 0) + d(\sigma 1)}{2}.$$

We think of d as representing a betting strategy. We begin with some capital $d(\lambda)$, where λ is the empty string, and bet on the values of the successive bits of a sequence X so that the amount of money we have after n many bets is $d(X \upharpoonright n)$. We are allowed to hedge our bets by betting some amount of our capital on 0 and the rest on 1. The displayed equation ensures that this betting is fair, i.e., that the average of the returns of our bets on 0 and on 1 equals our current total. A martingale d *succeeds* on a sequence X if and only if the associated betting strategy allows us to make arbitrarily much money when betting on the bits of X, that is, $\limsup_{n\to\infty} d(X \upharpoonright n) = \infty$. Schnorr showed that there is a notion of effective martingale such that X is Martin-Löf random if and only if no such martingale succeeds on X. This idea is close to von Mises' prediction-based approach, except that martingales allow us to spread our bets between the outcomes 0 and 1, so von Mises' intuition has a realization that works after all!

In summary, there are three basic approaches to defining random sequences:

- the *statistician's approach*, that a random sequence should have no computably rare properties;
- the *coder's approach*, that a random sequence should have no regularities that allow for compression; and
- the *gambler's approach*, that a random sequence should be unpredictable.

In each of these cases, a natural effective realization leads to the same notion, Martin-Löf randomness.

3 SOME THINGS WE HAVE LEARNED

3.1 Calibrating randomness

As natural and robust as Martin-Löf's definition of algorithmic randomness is, it is only one among many reasonable notions that together allow us to calibrate levels of randomness. One way to obtain new notions of randomness is to change the collection of tests that a sequence is required to pass to be considered random. For instance, we can consider Martin-Löf tests with computable measures (that is, where the measure of each level T_n is *exactly* 2^{-n}, for instance), which yields a notion called *Schnorr randomness*. Another possibility is to use martingales with different levels of effectiveness, for instance ones that are computable functions from strings to nonnegative rationals, which yields a notion called *computable randomness*. Computable randomness can also be miniaturized to complexity classes, giving rise to notions such as polynomial-time randomness.

It can be shown that Martin-Löf randomness implies computable randomness, which in turn implies Schnorr randomness, and that neither of these implications can be reversed. But the separations between these notions are quite subtle, and indeed the notions coincide for sequences that are in a sense "close to computable". (More precisely, they coincide outside what are know as the *high* sequences, which resemble the Halting Problem in a certain technical sense; see Nies, Stephan and Terwijn [34].) Indeed, there is a notion of *nonmonotonic randomness*—which is like computable randomness but allows for strategies that can bet on the values of the bits of a sequence in any computable order—for which equivalence to Martin-Löf randomness is still a longstanding open question.

We can also modify our tests to yield notions stronger than Martin-Löf randomness. For instance, relaxing the condition that the nth level T_n of a Martin-Löf test must have measure at most 2^{-n}, and requiring only that the measures of the T_n's go to 0 as n goes to infinity, yields the notion of *weak 2-randomness*, which is intermediate between Martin-Löf randomness and the notion of 2-randomness discussed below.

In some ways, weak 2-randomness is better-behaved than Martin-Löf randomness. To give an example, let us begin by considering the fact that, although almost every sequence is Martin-Löf random,

[3] Li and Vitanyi [25] report on an experiment of this kind about ant communication, with the "food-finding" ants describing the location of a food box in a similar kind of setup to the "food-gathering" ants.

[4] That is, descriptions that are like telephone numbers in that if τ and ρ are input descriptions to M and both give outputs, then τ is not a prefix of ρ. No telephone number should be a prefix of another! The point here is that in plain Kolmogorov complexity both 100 and 1001 could be descriptions, even of different strings, which requires us to have an implicit termination symbol.

[5] This notion is related to but distinct from that of martingale in probability theory.

it is not that easy to come up with an explicit example. That is at it should be, of course. Easily describable sequences (such as computable ones, for example) should not be random. Nevertheless, such examples do exist, the best-known being Chaitin's Ω, defined as the probability that a universal prefix-free Turing machine U halts on a given input, or, more formally, as

$$\Omega = \sum_{U(\sigma) \text{ halts}} 2^{-|\sigma|}.$$

(The exact value of Ω depends on the choice of U, but its basic properties do not.) While Ω is Martin-Löf random, it is also computationally powerful, being Turing equivalent to the Halting Problem.[6]

The existence of computationally powerful Martin-Löf random sequences is somewhat surprising, as intuitively we should expect random sequences not to contain much "useful information". (The distinction here is between the kind of information that makes a sequence hard to describe and the kind that can actually be used. If we choose 1000 characters at random, we expect the resulting text to be hard to describe, but would be shocked to find that it contains instructions for making a soufflé.) However, not only is it possible for a Martin-Löf random sequence to compute the Halting Problem, but *every* sequence can be computed from some Martin-Löf random sequence, as shown by Kučera [23] and Gács [16]. By increasing the level of randomness, we can make these "pathological" examples disappear. If X is weakly 2-random, then it cannot compute the Halting Problem. In fact, it cannot compute any noncomputable sequence that is computed by the Halting Problem, and in particular cannot compute any noncomputable, computably enumerable set.

We do not have to go all the way to weak 2-randomness, though. There are results, beginning with work of Stephan [37], that indicate that the Martin-Löf random sequences split into two classes: powerful ones that can compute the Halting Problem, and weaker ones that exhibit much more of the behavior we expect of random sequences, and in particular are computationally much weaker than the sequences in the first class. Franklin and Ng [15] showed that the level of randomness of these "true Martin-Löf randoms" can be captured by a natural test-based notion known as *difference randomness*. The study of notions of algorithmic randomness like this one, which are intermediate between Martin-Löf randomness and weak 2-randomness, has had an important role in recent research in the area, and helped us refine our understanding of the relationship between levels of randomness and computational power.

Another way to calibrate randomness is to relativize notions such as Martin-Löf randomness. For instance, we can consider Martin-Löf tests that are produced not by a standard Turing machine, but by a Turing machine with access to an oracle Z. If Z is the Halting Problem, for example, we obtain a notion called *2-randomness*. More generally, we have a notion of *n-randomness*, where we relativize Martin-Löf tests to the $(n-1)$st iterate of the Halting Problem.[7] Here 1-randomness is just Martin-Löf randomness.

[6] When we say that X can be computed from Y, we mean it in the sense of Turing reducibility. That is, there is a Turing machine M with an oracle tape so that if the oracle tape contains Y, then M computes X. Two objects are *Turing equivalent* if each can be computed from the other. Turing's Halting Problem is the classic example of a *complete* computably enumerable set; that is, it is itself computably enumerable, and it can compute every computably enumerable set.

[7] The kth iterate of the Halting Problem is just the Halting Problem for Turing machines with the $(k-1)$st iterate of the Halting Problem as an oracle.

Much is known about this hierarchy, including some surprising facts. Here are a few examples: As noted by Miller and Yu [31], it follows from a fundamental result about Martin-Löf randomness known as van Lambalgen's Theorem (see [10]) that if X is Martin-Löf random and is computed by an n-random sequence, then X is itself n-random. We have mentioned that we can never have $C(X \upharpoonright n) \geqslant n - O(1)$ for *all* n, but it is possible to have a sequence X such that $C(X \upharpoonright n) \geqslant n - O(1)$ for *infinitely many* n. Remarkably, Miller [28] and Nies, Stephan, and Terwijn [34] showed that this condition is equivalent to 2-randomness. Miller [29] also proved a similar result saying that 2-randomness also coincides with having infinitely often maximal initial segment prefix-free Kolmogorov complexity. Indeed, it is possible to give characterizations of n-randomness for all n using unrelativized Kolmogorov complexity, by results of Vereshchagin [39] and Bienvenu, Muchnik, Shen, and Vereshchagin [7]. These facts are examples of the often subtle interplay that recent research in this area has uncovered between levels of randomness, initial-segment complexity, and relative computability.

3.2 Calibrating nonrandomness

For sequences that are not Martin-Löf random, there are ways to calibrate how close they come to randomness. A natural way to do this is to consider the (prefix-free) Kolmogorov complexity of their initial segments. For example, a sequence X is *complex* if there is a computable, nondecreasing, unbounded function f such that $K(X \upharpoonright n) \geqslant f(n)$ for all n. Complex sequences can be characterized in terms of their ability to compute certain sequences that resemble the Halting Problem to some extent (see [10]), which is another example of the interplay between randomness and computability.

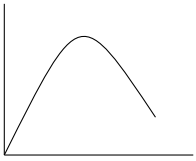
At the other extreme from random sequences are those that have strong "anti-randomness" properties. Identifying a natural number with its binary expansion, we always have $C(\sigma) \geqslant C(|\sigma|) - O(1)$, since if we know a string then we can easily describe its length. Thus for a sequence X, the lowest the plain Kolmogorov complexity of the initial segments of X can be is $C(X \upharpoonright n) \leqslant C(n) + O(1)$. In the 1970s, Chaitin showed that this condition is satisfied if and only if X is computable, and asked whether the same holds for prefix-free Kolmogorov complexity.

In an unpublished manuscript written in 1975, Solovay showed the surprising fact that there are noncomputable sequences X such that $K(X \upharpoonright n) \leqslant K(n) + O(1)$ for all n, though Chaitin had already shown that there are only countably many of them, and indeed that they are all computable from the Halting Problem. Such sequences are said to be *K-trivial*, and they have played a major role in the theory of algorithmic randomness. For those who know some computability theory, we mention that, as shown by Nies [32], the K-trivial sequences form an ideal in the Turing degrees, and that they can be seen as giving a kind of priority-free solution to Post's Problem (see Downey, Hirschfeldt, Nies, and Stephan [11]). Nies [32] showed that these sequences are computability-theoretically weak, and gave several characterizations of K-triviality in terms of natural notions of randomness-theoretic weakness. For example, when we relativize the notion of Martin-Löf randomness to a noncomputable X, we expect the notion to change, because the

noncomputability of X should result in some amount of derandomization power. Nies showed that the K-trivial sequences are exactly those for which this intuition fails.

Since then, many other characterizations of K-triviality have been given. For example, a result of Hirschfeldt, Nies, and Stephan [19] and more recent work of Bienvenu, Day, Greenberg, Kučera, Miller, Nies, and Turetsky [5] show that a computably enumerable set is K-trivial if and only if it is computed by a difference random sequence (i.e., one of the "true Martin-Löf randoms" that does not compute the Halting Problem). Recent work on K-triviality has also revealed subclasses of the K-trivials that can further help us understand the fine structure of the interaction between randomness and computability.

Considering the properties of highly nonrandom sequences like the K-trivials, and those of sequences with increasing levels of randomness, leads to the following heuristic graph, where the horizontal axis represents randomness level and the vertical axis represents maximum computational power. (Another way to see this graph is that the horizontal axis represents information content, while the vertical axis represents maximum *useful* information content.)



Among the sequences that are neither random nor highly nonrandom are ones that can be thought of as being "partially random". For example, if Z is Martin-Löf random and we replace every other bit of Z by a 0, we obtain a new sequence Y such that $K(Y \upharpoonright n)$ is roughly $\frac{n}{2}$. It makes sense to think of such a sequence as being "$\frac{1}{2}$-random". More generally, we can think of the limit behavior of the ratio $\frac{K(X\upharpoonright n)}{n}$ as a measure of the partial randomness of a sequence X. This ratio does not necessarily have a limit, but we can look at

$$\liminf_{n\to\infty} \frac{K(X \upharpoonright n)}{n} \qquad \text{and} \qquad \limsup_{n\to\infty} \frac{K(X \upharpoonright n)}{n},$$

which both give us values between 0 and 1.

These values are also central to the theory of effective dimension. In 1919, Hausdorff introduced a notion of dimension that measures the "local size" of a set in a metric space, for example a subset of the plane. As usual, points have dimension 0, lines have dimension 1, and the whole plane has dimension 2, but there are also objects of fractional dimension, such as well-known fractals like the Koch curve (which has Hausdorff dimension $\log_3(4)$). Starting with the work of Lutz in the early 2000's, the theory of dimension has been effectivized, initially by using a characterization of Hausdorff dimension in terms of martingales and passing to effective martingales as in Schnorr's approach to algorithmic randomness. This process has also been carried out for other notions of dimension, most notably that of packing dimension. An important fact here is that the effective Hausdorff dimension and effective packing dimension of a sequence X turn out to be exactly the liminf and limsup, respectively, in the above displayed equation. Thus these dimensions can be seen as measures of partial randomness. (See e.g. [10] for details.)

The theory of effective dimension has also been extended to points on the plane and higher-dimensional Euclidean spaces. A remarkable feature of this theory is that there is a tight correspondence between the classical Hausdorff dimension of a set and the effective Hausdorff dimension of its points. For a fairly wide class of sets $S \subseteq \mathbb{R}^n$, Hitchcock [20] showed that the Hausdorff dimension of S is the supremum of the effective Hausdorff dimensions of its individual elements, and Lutz and Lutz [26] have now given versions of this result for arbitrary sets (and for both Hausdorff and packing dimension) using relativizations of effective dimension. It is surprising that the notion of dimension, which seems so clearly to be a global property of a set, based on its "overall shape", can be completely understood by focusing on the individual elements of the set and understanding them from a computability-theoretic perspective. This correspondence is also quite useful, and can be used to obtain new proofs and results in areas such as fractal geometry, as in Lutz and Lutz [26] and Lutz and Stull [27], for instance.

Randomness amplification is an issue that can be investigated in many settings. A basic question is whether (a greater degree of) randomness can always be extracted from a partially random source. In our setting, effective dimension can be used to measure the degree of randomness of a sequence, and extraction can be interpreted algorithmically, i.e., as relative computation. One way to think of this question is that it is easy to decrease the effective dimension of a sequence in a computable way, say by changing a large proportion of its bits to 0's, but it is less clear in general whether there is a way to reverse this process.

As it turns out, the answer depends on which notion of dimension we are considering. Fortnow, Hitchcock, Pavan, Vinochandran, and Wang [14] showed that if X has nonzero effective packing dimension and $\varepsilon > 0$, then there is a Y that is computable from X such that the effective packing dimension of Y is at least $1 - \varepsilon$.[8] On the other hand, Miller [30] showed that there is a sequence X of effective Hausdorff dimension $\frac{1}{2}$ such that if Y is computable from X then the effective Hausdorff dimension of Y is at most $\frac{1}{2}$. (The specific value $\frac{1}{2}$ does not matter here.) Greenberg and Miller [17] showed that there is a sequence of effective Hausdorff dimension 1 that does not compute any Martin-Löf random sequence. Thus we see that there are some strong senses in which randomness amplification is not possible. However, Zimand [40] showed that, remarkably, if we have *two* sequences of nonzero effective Hausdorff dimension that are sufficiently independent in a certain technical sense, then they together compute a sequence of effective Hausdorff dimension 1.

This is still an area of significant research interest. For example, we can ask about a randomness amplification process where, instead of using computable reductions, we simply seek to increase the randomness of a sequence by changing a relatively small proportion of its bits. Greenberg, Miller, Shen, and Westrick [18] recently gave precise bounds on the proportion of bits of a sequence of effective Hausdorff dimension s that need to be changed to increase the Hausdorff dimension to a given $t > s$, in terms of the binary entropy function from information theory. They also showed that if X has

[8] In fact, they showed that Y can be chosen to be Turing equivalent to X via polynomial-time reductions, making the randomness amplification process quite efficient in this case.

effective Hausdorff dimension 1 then X can be transformed into a Martin-Löf random sequence by changing it only on the bits in a set $S \subset \mathbb{N}$ of density 0 (which means that $\lim_{n\to\infty} \frac{|S\restriction n|}{n} = 0$).

3.3 Turing and absolute normality

We return to Borel's notion of normality. This is a very weak form of randomness; polynomial-time randomness is more than enough to ensure absolute normality, and indeed, Schnorr and Stimm [35] showed that a sequence is normal if and only if it satisfies a notion of randomness defined using certain finite state machines, which are much weaker than arbitrary Turing machines. Borel asked whether there are explicit examples of absolutely normal numbers. It is conjectured that e, π, and all irrational algebraic numbers, such as $\sqrt{2}$, are absolutely normal, but *none* of these have been proven to be normal to *any* base. In an unpublished manuscript, Turing attacked the question of an explicit construction of an absolutely normal number by interpreting "explicit" to mean *computable*. His manuscript, entitled *A note on normal numbers* and presumably written in 1938, gives the best kind of answer to date to Borel's question: an algorithm that produces an absolutely normal number.

An interesting aspect of Turing's construction is that he more or less anticipated Martin-Löf's work by looking at a collection of computable tests sensitive enough to make a number normal in all bases, yet insensitive enough to allow computable sequences to pass all such tests. He began by giving an extension of the law of large numbers to "blocks" of digits. Indeed, it makes sense that not just single digits, but fixed blocks of digits should occur with the appropriate frequencies in a random sequence. Translating between bases results in correlations between blocks of digits in one base and blocks of digits in the other, which is why this extension allowed Turing to construct absolutely normal numbers.

Turing's construction remained largely unknown, because his manuscript was published only in his 1997 Collected Works [38]. The editorial notes in that volume say that the proof given by Turing is inadequate and speculate that the theorem could be false. Becher, Figueira, and Picchi [4] reconstructed and completed Turing's manuscript, preserving his ideas as accurately as possible while correcting minor errors. More recently, there has been a highly productive line of research connecting algorithmic randomness, computability theory, normal numbers, and approximability notions such as that of Liouville numbers; see for instance the papers listed at http://www-2.dc.uba.ar/profesores/becher/publications.html. Some of this work has yielded results in the classical theory of normal numbers, as in Becher, Bugeaud, and Slaman [3].

3.4 Some further applications

There have been several other applications of ideas related to algorithmic randomness in areas such as logic, complexity theory, analysis, and ergodic theory. Chaitin famously used Kolmogorov complexity to give a proof of a version of Gödel's First Incompleteness Theorem, by showing that for any sufficiently strong, computably axiomatizable, consistent theory T, there is a number c such that T cannot prove that $C(\sigma) > c$ for any given string σ. More recently, Kritchman and Raz [22] used his methods to give a proof of the Second Incompleteness Theorem as well. (Their paper also includes an account of Chaitin's proof.) We can also ask about the effect of adding axioms asserting the incompressibility of certain strings in a probabilistic way. Bienvenu, Romashchenko, Shen, Taveneaux, and Vermeeren [8] have shown that this kind of procedure does not help us to prove new interesting theorems, but that the situation changes if we take into account the size of the proofs: randomly chosen axioms can help to make proofs much shorter under a reasonable complexity-theoretic assumption.

Randomness is used in several algorithms to accelerate computations. A classic example is the use of randomness for primality testing by Solovay and Strassen [36], and there are problems like *polynomial identity testing*—which asks whether a polynomial in many variables is identically zero, like $x_1x_2 - x_2x_1$, say—for which there are efficient algorithms if we have a randomness source, but no known fast deterministic algorithms. It is thought that a wide class of randomized algorithms can be derandomized to yield deterministic polynomial-time algorithms, following the work of Impagliazzo and Wigderson [21], who showed that if certain problems are as hard as we think they are, then we can provide enough randomness efficiently to derandomize problems in the complexity class BPP. A recent result of Bienvenu and Downey [6] implies that randomness can always be used to accelerate *some* computations. They showed that if X is Schnorr random, then there is a computable language L such that X can compute L (in exponential time) via a computation Φ^X (i.e., a Turing machine Φ with oracle X) so that for any Turing machine M that computes L, the computation Φ^X is faster than M by more than a polynomial factor. (That is, Φ^X computes L in time f, and there are no Turing machine M and polynomial p such that M computes L in time $p \circ f$.)

Another connection with complexity theory comes from looking at the computational power of the set of random strings. There are a few reasonable ways to define what we mean by this set; one of them is to consider the strings that are incompressible in the sense of plain Kolmogorov complexity, that is

$$R = \{\sigma \mid C(\sigma) \geqslant |\sigma|\}.$$

It turns out to be particularly interesting to consider what sets can be reduced to this one via polynomial-time reductions. For instance, Allender, Buhrman, Koucký, van Melkebeek, and Ronneburger [1] showed that the complexity class PSPACE is contained in the collection of sets that are polynomial-time reducible to R, and other connections with complexity theory have been explored in this paper and other such as Allender, Friedman, and Gasarch [2].

A particularly promising current line of research is the use of notions of algorithmic randomness to give precise, "quantitative" versions of results about almost everywhere behavior in areas such as analysis and ergodic theory, an idea that goes back to the work of Demuth in the 1970's.[9] For example, it is a result of basic analysis that every nondecreasing function $[0, 1] \to \mathbb{R}$ is differentiable at almost every $x \in [0, 1]$ (that is, the set of x at which it is differentiable has measure 1). Brattka, Miller, and Nies [9] showed that the reals $x \in [0, 1]$ such that every nondecreasing computable function (in the sense of computable analysis) is differentiable at x are

[9] Demuth came from the constructivist tradition, but he independently rediscovered notions of randomness like Martin-Löf randomness by working on questions such as the ones discussed in this paragraph. See Kučera, Nies, and Porter [24] for an account of his work.

exactly the computably random ones. Thus computable randomness is exactly the level of randomness needed for this particular almost everywhere behavior to manifest itself. For other similar conditions, the relevant level of randomness can vary. For instance, for functions of bounded variation in place of nondecreasing ones, the corresponding level of randomness is exactly Martin-Löf randomness, as shown in [9] as a recasting of a result by Demuth.

One source for overviews of some recent work at the intersection of algorithmic randomness with analysis and ergodic theory is the collection of slides at https://www.birs.ca/cmo-workshops/2016/16w5072/files/.

Another interesting application is to the study of tilings (of the plane, say). For a sequence X, let $X[m, n]$ be the string consisting of the bits of X from position m to position n. One might think that for a Martin-Löf random X, we should have $K(X[m, n]) \geqslant n-m-O(1)$, or that at least $K(X[m, n])$ should not dip too far below $n - m$. This is not the case, though, because random sequences must have long simple substrings, such as long runs of 0's. (If we know that X has infinitely many runs of 6 consecutive 0's, but only finitely many runs of 7 consecutive 0's, then we can make money betting on the values of the bits of X by betting that the next value is 1 each time we see 6 consecutive 0's.) However, for any $\varepsilon > 0$, there are *ε-shift complex* sequences X for which

$$K(X[m, n]) \geqslant (1 - \varepsilon)(n - m) - O(1)$$

for all m and n. These sets can be coded to yield tilings with various interesting properties, such as certain kinds of pattern-avoidance. See for instance Durand, Levin, and Shen [12] and Durand, Romashchenko, and Shen [13].

ACKNOWLEDGMENTS

Downey wishes to thank the Marsden Fund of New Zealand. Hirschfeldt is partially supported by NSF Grant DMS-1600543.

REFERENCES

[1] E. Allender, H. Buhrman, M. Koucký, D. van Melkebeek, and D. Ronneburger. 2006. Power from random strings. *SIAM J. Comput.* 35, 6 (2006), 1467–1493.

[2] E. Allender, L. Friedman, and W. Gasarch. 2013. Limits on the computational power of random strings. *Information and Computation* 222 (2013), 80–92.

[3] V. Becher, Y. Bugeaud, and T. A. Slaman. 2016. On simply normal numbers to different bases. *Math. Ann.* 364, 1–2 (2016), 125–150.

[4] V. Becher, S. Figueira, and R. Picchi. 2007. Turing's unpublished algorithm for normal numbers. *Theoretical Computer Science* 377, 1–3 (2007), 126–138.

[5] L. Bienvenu, A. R. Day, N. Greenberg, A. Kučera, J. S. Miller, A. Nies, and D. Turetsky. 2014. Computing K-trivial sets by incomplete random sets. *The Bulletin of Symbolic Logic* 20, 1 (2014), 80–90.

[6] L. Bienvenu and R. Downey. To appear. On low for speed oracles. (To appear). arXiv:1712.09710.

[7] L. Bienvenu, An. A. Muchnik, A. Shen, and N. Vereshchagin. 2008. Limit complexities revisited. In *25th International Symposium on Theoretical Aspects of Computer Science.*, S. Albers and P. Weil (Eds.). Leibniz Int. Proc. Inform., Vol. 1. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 73–84 (electronic).

[8] L. Bienvenu, A. Romashchenko, A. Shen, A. Taveneaux, and S. Vermeeren. 2014. The axiomatic power of Kolmogorov complexity. *Annals of Pure and Applied Logic* 165, 9 (2014), 1380–1402.

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

[10] R. G. Downey and D. R. Hirschfeldt. 2010. *Algorithmic Randomness and Complexity.* Springer, New York.

[11] R. G. Downey, D. R. Hirschfeldt, A. Nies, and F. Stephan. 2003. Trivial reals. In *Proceedings of the 7th and 8th Asian Logic Conferences*, R. G. Downey, D. Ding, S. P. Tung, Y. H. Qiu, and M. Yasugi (Eds.). Singapore University Press and World Scientific, Singapore, 103–131.

[12] B. Durand, L. A. Levin, and A. Shen. 2008. Complex tilings. *The Journal of Symbolic Logic* 73, 2 (2008), 593–613.

[13] B. Durand, A. Romashchenko, and A. Shen. 2012. Fixed-point tile sets and their applications. *J. Comput. System Sci.* 78, 3 (2012), 731–764.

[14] L. Fortnow, J. M. Hitchcock, A. Pavan, V. Vinochandran, and F. Wang. 2006. Extracting Kolmogorov complexity with applications to dimension zero-one laws. In *Automata, Languages and Programming. 33rd International Colloquium, ICALP 2006. Venice, Italy, July 10–14, 2006. Proceedings, Part I*, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener (Eds.). Lecture Notes in Computer Science, Vol. 4051. Springer, Berlin, 335–345.

[15] J. N. Y. Franklin and K. M. Ng. 2011. Difference randomness. *Proc. Amer. Math. Soc.* 139, 1 (2011), 345–360.

[16] P. Gács. 1986. Every set is reducible to a random one. *Information and Control* 70 (1986), 186–192.

[17] N. Greenberg and J. S. Miller. 2011. Diagonally non-recursive functions and effective Hausdorff dimension. *Bulletin of the London Mathematical Society* 43, 4 (2011), 636–654.

[18] N. Greenberg, J. S. Miller, A. Shen, and L. B. Westrick. 2018. Dimension 1 sequences are close to randoms. *Theoretical Computer Science* 705 (2018), 99–112.

[19] D. R. Hirschfeldt, A. Nies, and F. Stephan. 2007. Using random sets as oracles. *Journal of the London Mathematical Society. Second Series* 75 (2007), 610–622.

[20] J. M. Hitchcock. 2005. Correspondence Principles for Effective Dimensions. *Theory of Computing Systems* 38 (2005), 559–571.

[21] R. Impagliazzo and A. Wigderson. 1999. P = BPP if E requires exponential circuits: derandomizing the XOR lemma. In *STOC '97 (El Paso, TX).* ACM, New York, 220–229.

[22] S. Kritchman and R. Raz. 2010. The surprise examination paradox and the second incompleteness theorem. *Notices of the American Mathematical Society* 57, 11 (2010), 1454–1458.

[23] A. Kučera. 1985. Measure, Π^0_1 classes, and complete extensions of PA. In *Recursion Theory Week. Proceedings of the Conference Held at the Mathematisches Forschungsinstitut in Oberwolfach, April 15–21, 1984*, H.-D. Ebbinghaus, G. H. Müller, and G. E. Sacks (Eds.). Lecture Notes in Mathematics, Vol. 1141. Springer, Berlin, 245–259.

[24] A. Kučera, A. Nies, and C. P. Porter. 2015. Demuth's path to randomness. *The Bulletin of Symbolic Logic* 21, 3 (2015), 270–305. https://doi.org/10.1017/bsl.2015.24

[25] M. Li and P. Vitanyi. 1993. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer-Verlag, Berlin.

[26] J. H. Lutz and N. Lutz. 2017. Algorithmic information, plane Kakeya sets, and conditional dimension. In *34th Symposium on Theoretical Aspects of Computer Science.* LIPIcs. Leibniz Int. Proc. Inform., Vol. 66. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, Art. No. 53, 13.

[27] N. Lutz and D. M. Stull. 2017. Bounding the dimension of points on a line. In *Theory and applications of models of computation.* Lecture Notes in Comput. Sci., Vol. 10185. Springer, Cham, 425–439.

[28] J. S. Miller. 2004. Kolmogorov random reals are 2-random. *The Journal of Symbolic Logic* 69 (2004), 907–913.

[29] J. S. Miller. 2010. The K-degrees, low for K-degrees, and weakly low for K sets. *Notre Dame Journal of Formal Logic* 50 (2010), 381–391.

[30] J. S. Miller. 2011. Extracting information is hard: a Turing degree of non-integral effective Hausdorff dimension. *Advances in Mathematics* 226, 1 (2011), 373–384.

[31] J. S. Miller and L. Yu. 2008. On initial segment complexity and degrees of randomness. *Trans. Amer. Math. Soc.* 360 (2008), 3193–3210.

[32] A. Nies. 2005. Lowness properties and randomness. *Advances in Mathematics* 197 (2005), 274–305.

[33] A. Nies. 2009. *Computability and Randomness.* Oxford Logic Guides, Vol. 51. Oxford University Press, Oxford.

[34] A. Nies, F. Stephan, and S. A. Terwijn. 2005. Randomness, relativization, and Turing degrees. *The Journal of Symbolic Logic* 70 (2005), 515–535.

[35] C.-P. Schnorr and H. Stimm. 1971/72. Endliche Automaten und Zufallsfolgen. *Acta Informatica* 1, 4 (1971/72), 345–359.

[36] R. Solovay and V. Strassen. 1977. A fast Monte-Carlo test for primality. *SIAM J. Comput.* 6, 1 (1977), 84–85.

[37] F. Stephan. 2006. Martin-Löf random sets and PA-complete sets. In *Logic Colloquium '02*, Z. Chatzidakis, P. Koepke, and W. Pohlers (Eds.). Lecture Notes in Logic, Vol. 27. Association for Symbolic Logic and A K Peters, Ltd., La Jolla, CA and Wellesley, MA, 342–348.

[38] A. M. Turing. 1992. *Pure Mathematics.* North-Holland Publishing Co., Amsterdam. Edited and with an introduction and postscript by J. L. Britton, With a preface by P. N. Furbank.

[39] N. Vereshchagin. 2002. Kolmogorov complexity conditional to large integers. *Theoretical Computer Science* 271 (2002), 59–67.

[40] M. Zimand. 2010. Two sources are better than one for increasing the Kolmogorov complexity of infinite sequences. *Theory of Computing Systems* 46 (2010), 707–722.