THE MATHEMATICAL FOUNDATIONS OF RANDOMNESS

SEBASTIAAN A. TERWIJN

Abstract. We give a nontechnical account of the mathematical theory of randomness. The theory of randomness is founded on computability theory, and it is nowadays often referred to as algorithmic randomness. It comes in two varieties: A theory of finite objects, that emerged in the 1960s through the work of Solomonoff, Kolmogorov, Chaitin and others, and a theory of infinite objects (starting with von Mises in the early 20th century, culminating in the notions introduced by Martin-Löf and Schnorr in the 1960s and 1970s) and there are many deep and beautiful connections between the two. Research in algorithmic randomness connects computability and complexity theory with mathematical logic, proof theory, probability and measure theory, analysis, computer science, and philosophy. It also has surprising applications in a variety of fields, including biology, physics, and linguistics. Founded on the theory of computation, the study of randomness has itself profoundly influenced computability theory in recent years.

Contents

Date: July 6, 2015.

1. Introduction

In this chapter we aim to give a nontechnical account of the mathematical theory of randomness. This theory can be seen as an extension of classical probability theory that allows us to talk about individual random objects. Besides answering the philosophical question what it *means* to be random, the theory of randomness has applications ranging from biology, computer science, physics, and linguistics, to mathematics itself.

The theory comes in two flavors: A theory of randomness for finite objects (for which the textbook by Li and Vitányi [18] is the standard reference) and a theory for infinite ones. The latter theory, as well as the relation between the two theories of randomness, is surveyed in the paper [10], and developed more in full in the recent textbooks by Downey and Hirschfeldt [9] and Nies [21]. Built on the theory of computation, the theory of randomness has itself deeply influenced computability theory in recent years.

We warn the reader who is afraid of mathematics that there will be formulas and mathematical notation, but we promise that they will be *explained* at a nontechnical level. Some more background information about the concepts involved is given in footnotes and in two appendices. It is fair to say, however, that to come to a better understanding of the subject, there is of course no way around the formulas, and we quote Euclid, who supposedly told King Ptolemy I, when the latter asked about an easier way of learning geometry than Euclid's Elements, that "there is no royal road to geometry".[1]

2. What is randomness?

Classical probability theory talks about random objects, for example by saying that if you randomly select four cards from a standard deck, the probability of getting four aces is very small. However, every configuration of four cards has the *same* small probability of appearing, so there is no qualitative difference between individual configurations in this setting. Similarly, if we flip a fair coin one hundred times, and we get a sequence of one hundred tails in succession, we may feel that this outcome is very special, but how do we justify our excitement over this outcome? Is the probability for this outcome not exactly the same as that of any other sequence of one hundred heads and tails?

Probability theory has been, and continues to be, a highly successful theory, with applications in almost every branch of mathematics. It

[1]As with many anecdotes of this kind, it is highly questionable if these words were really spoken, but the message they convey is nevertheless true.

was put on a sound mathematical foundation in 1933 by Andrei Kolmogorov [14], and in its modern formulation it is part of the branch of mathematics called *measure theory*. (See appendix A.) In this form it allows us to also talk not only about randomness in discrete domains (such as cards and coin flips), but also in continuous domains such as numbers on the real line. However, it is important to realize that even in this general setting, probability theory is a theory about *sets* of objects, not of individual objects. In particular, it does not answer the question what an *individual* random object is, or how we could call a sequence of fifty zero's less random than any other sequence of the same length. Consider the following two sequences of coin flips, where 0 stands for heads and 1 for tails:

00000000000000000000000000000000000000000000000000

00001110011111011110011110010010101111001111010111

The first sequence consists of fifty 0's, and the second was obtained by flipping a coin fifty times.[2] Is there any way in which we can make our feeling that the first sequence is special, and that the second is less so, mathematically precise?

3. Can randomness be defined?

A common misconception about the notion of randomness is that it cannot be formally defined, by applying a tautological reasoning of the form: As soon as something can be precisely defined, it ceases to be random. The following quotation by the Dutch topologist Hans Freudenthal [11] (taken from [16]) may serve to illustrate this point:

> It may be taken for granted that any attempt at defining disorder in a formal way will lead to a contradiction. This does not mean that the notion of disorder is contradictory. It is so, however, as soon as I try to formalize it.

A recent discussion of randomness and definability, and what can happen if we equate "random" with "not definable", is in Doyle [8].[3] The

[2]The author actually took the trouble of doing this. We could have tried to write down a random sequence ourselves, but it is known that humans are notoriously bad at producing random sequences, and such sequences can usually be recognized by the fact that most people avoid long subsequences of zero's, feeling that after three or four zero's it is really time for a one. Indeed, depending on one's temperament, some people may feel that the first four zero's in the above sequence look suspicious.

[3]The notion of mathematical definability is *itself* definable in set theory, see Kunen [15, Chapter V]. If "random" is equated with "not definable", then the following problem arises: By a result of Gödel [12] *it is consistent with the axioms*

problem is not that the notion of definability is inherently vague (because it is not), but that no *absolute* notion of randomness can exist, and that in order to properly define the notion, one has to specify *with respect to what* the supposed random objects should be random. This is precisely what happens in the modern theory of randomness: A random object is defined as an object that is random with respect to a given type of definition, or class of sets. As the class may vary, this yields a *scale* of notions of randomness, which may be adapted to the specific context in which the notion is to be applied.

The first person to attempt to give a mathematical definition of randomness was Richard von Mises [28], and his proposed definition met with a great deal of opposition of the kind indicated above. Von Mises formalized the intuition that a random sequence should be *unpredictable*. Without giving technical details, his definition can be described as follows. Suppose that X is an infinite binary sequence, that is, a sequence

$$X(0), X(1), X(2), X(3), \ldots$$

where for each positive integer n, $X(n)$ is either 0 or 1. Suppose further that the values of X are unknown to us. We now play a game: At every stage of the game we point to a new location n in the sequence, and then the value of $X(n)$ is revealed to us. Now, according to von Mises, for X to be called random, we should not be able to predict in this way the values of X with probability better than $\frac{1}{2}$, no matter how we select the locations in X. A strategy to select locations in X is formalized by a *selection function*, and hence this notion says that no selection function should be able to give us an edge in predicting values from X. However, as in the above discussion on absolute randomness, in this full generality, *this notion is vacuous!* To counter this, von Mises proposed to restrict attention to "acceptable" selection rules, without further specifying which these should be. He called the sequences satisfying his requirement for randomness *Kollektiv*'s.[4]

Later Wald [29, 30] showed that von Mises' notion of Kollektiv is nonempty if we restrict to any *countable* set of selection functions.[5] Wald did not specify a canonical choice for such a set, but later Church [6]

of set theory that all sets are definable, and hence the notion of randomness becomes empty. The solution to this problem is to be more modest in defining randomness, by only considering more restricted classes of sets, as is explained in what follows.

[4]For a more elaborate discussion of the notion of Kollektiv see van Lambalgen [16].

[5]A set is called *countable* if its elements can be indexed by the natural numbers $0, 1, 2, 3, \ldots$ These sets represent the smallest kind of infinity in the hierarchy of infinite sets.

suggested that the (countable) set of *computable* selection rules would be such a canonical choice. We thus arrive at the notion of *Mises–Wald–Church randomness*, defined as the set of Kollektiv's based on computable selection rules. This notion of random sequence already contains several of the key ingredients of the modern theory of randomness, namely:

- the insight that randomness is a *relative* notion, not an absolute one, in that it depends on the choice of the set of selection rules;
- it is founded on the theory of computation, by restricting attention to the *computable* selection functions (cf. section 4).

Ville [27] later showed that von Mises' notion of Kollektiv is flawed in the sense that there are basic statistical laws that are not satisfied by them. Nevertheless, the notion of Mises–Wald–Church randomness has been decisive for the subsequent developments in the theory of randomness.[6]

The Mises–Wald–Church notion formalized the intuition that a random sequence should be *unpredictable*. This was taken further by Ville using the notion of martingale. We discuss this approach in section 7. The approach using Kolmogorov complexity formalizes the intuition that a random sequence, since it is lacking in recognizable structure, is hard to *describe*. We discuss this approach in section 5. Finally, the notion randomness proposed by Martin-Löf formalizes the intuitions underlying classical probability and measure theory. This is discussed in section 6. It is a highly remarkable fact that these approaches are intimately related, and ultimately turn out to be essentially equivalent. As the theory of computation is an essential ingredient in all of this, we have to briefly discuss it before we can proceed.

4. Computability theory

The theory of computation arose in the 1930s out of concerns about what is provable in mathematics and what is not. Gödel's famous incompleteness theorem from 1931 states, informally speaking, that in any formal system strong enough to reason about arithmetic, *there always exist true statements that are not provable in the system*. This shows that there can never be a definitive formal system encompassing all of mathematics. Although it is a statement about mathematical provability, the proof of the incompleteness theorem shows that it is in essence a result about *computability*. The recursive functions used by Gödel in his proof of the incompleteness theorem were later shown by

[6]In the light of the defects in the definition of Mises–Wald–Church random sequences, these sequences are nowadays called *stochastic* rather than random.

Turing [26] to define the same class of functions computable by a Turing machine. Subsequently, many equivalent definitions of the same class of computable functions were found, leading to a robust foundation for a general theory of computation, called *recursion theory*, referring to the recursive functions in Gödel's proof. Nowadays the area is mostly called *computability theory*, to emphasize that it is about what is computable and what is not, rather than about recursion.

Turing machines serve as a very basic model of computation, which are nevertheless able to perform any type of algorithmic computation.[7] The fortunate circumstance that there are so many equivalent definitions of the same class of computable functions allows us to treat this notion very informally, without giving a precise definition of what a Turing machine is. Thus, a *computable function* is a function for which there is an algorithm, i.e. a finite step-by-step procedure, that computes it. It is an empirical fact that any reasonable formalization of this concept leads to the same class of functions.[8]

Having a precise mathematical definition of the notion of computability allows us to prove that certain functions or problems are *not* computable. One of the most famous examples is Turing's Halting Problem:

Definition 4.1. The *Halting Problem* is the problem, given a Turing machine M and an input x, to decide whether M produces an output on x in a finite number of steps (as opposed to continuing indefinitely).

Turing [26] showed that the Halting Problem is *undecidable*, that is, that there is no algorithm deciding it. (Note the self-referential flavor of this statement: There is no algorithm deciding the behavior of algorithms.) Not only does this point to a fundamental obstacle in computer science (which did not yet exist in at the time that Turing proved this result), but it also entails the undecidability of a host of

[7]It is interesting to note that the Turing machine model has been a blueprint for all modern electronic computers. In particular, instead of performing specific algorithms, Turing machines are *universally programmable*, i.e. any algorithmic procedure can be implemented on them. Thus, the theory of computation *preceded* the actual building of electronic computers, and the fact that the first computers were universally programmable was directly influenced by it (cf. [7]). This situation is currently being repeated in the area of *quantum computing*, were the theory is being developed before any actual quantum computers have been built. (see e.g. [2]).

[8]The statement that the informal and the formal notions of computability coincide is the content of the so-called *Church-Turing thesis*, cf. Odifreddi [22] for a discussion.

other problems.[9] Its importance for the theory of randomness will become clear in what follows.

5. Kolmogorov complexity

An old and venerable philosophical principle, called *Occam's razor*, says that when given the choice between several hypotheses or explanations, one should always select the simplest one. The problem in applying this principle has always been to determine which is the simplest explanation: that which is simple in one context may be complicated in another, and there does not seem to be a canonical choice for a frame of reference.

A similar problem arises when we consider the two sequences on page 3: We would like to say that the first one, consisting of only 0's, is simpler than the second, because it has a shorter *description*. But what are we to choose as our description mechanism? When we require, as seems reasonable, that an object can be effectively reconstructed from its description, the notion of Turing machine comes to mind. For simplicity we will for the moment only consider finite binary strings. (This is not a severe restriction, since many objects such as numbers and graphs can be *represented* as binary strings in a natural way.) Thus, given a Turing machine M, we define a string y to be a description of a string x if $M(y) = x$, i.e. M produces x when given y as input. Now we can take the *length* of the string y as a measure of the complexity of x. However, this definition still depends on the choice of M. Kolmogorov observed that a canonical choice for M would be a *universal* Turing machine, that is, a machine that is able to simulate all other Turing machines. It is an elementary fact of computability theory that such universal machines exist. We thus arrive at the following definition:

Definition 5.1. Fix a universal Turing machine U. The *Kolmogorov complexity* of of a finite binary string x is the smallest length of a string y such that

$$U(y) = x.$$

We denote the Kolmogorov complexity of the string x by $C(x)$.

Hence, to say that $C(x) = n$ means that there is a string y of length n such that $U(y) = x$, and that there is no such y of length smaller than n. Note that the definition of $C(x)$ *still* depends on the choice of

[9] In [26] Turing used the undecidability of the Halting Problem to show the undecidability of the *Entscheidungsproblem*, that says (in modern terminology) that first-order predicate logic is undecidable.

U. However, and this is the essential point, *the theory of Kolmogorov complexity is independent of the choice of* U in the sense that when we choose a different universal Turing machine U' as our frame of reference, the whole theory only shifts by a fixed constant.[10] For this reason, the reference to U is suppressed from this point onwards, and we will simply speak about *the* Kolmogorov complexity of a string.

Armed with this definition of descriptive complexity, we can now define what it means for a finite string to be random. The idea is that a string is random if it has no description that is shorter than the string itself, that is, if there is no way to describe the string more efficiently than by listing it completely.

Definition 5.2. A finite string x is *Kolmogorov random* if $C(x)$ is at least the length of x itself.

For example, a sequence of 1000 zero's is far from random, since its shortest description is much shorter than the string itself: The string itself has length 1000, but we have just described it using only a few words.[11] More generally, if a string contains a regular pattern that can be used to efficiently describe it, then it is not random. Thus this notion of randomness is related to the *compression* of strings: If $U(y) = x$, and y is shorter than x, we may think of y as a *compressed* version of x, and random strings are those that cannot be compressed.

A major hindrance in using Kolmogorov complexity is the fact that *the complexity function* C *is noncomputable*. A precise proof of this fact is given in appendix B (see Corollary B.2), but it is also intuitively plausible, since to compute the complexity of y we have to see for which inputs x the universal machine U produces y as output. But as we have seen in section 4, this is in general impossible to do by the undecidability of the Halting Problem! This leaves us with a definition that may be wonderful for theoretical purposes, but that one would not expect to be of much practical relevance. One of the miracles of Kolmogorov complexity is that the subject *does* indeed have genuine applications,

[10]This is not difficult to see: Since both U and U' are universal, they can simulate each other, and any description of x relative to U can be translated into a description relative to U' using only a fixed constant number of extra steps, where this constant is independent of x.

[11]Notice that the definition requires the description to be a string of 0's and 1's, but we can easily convert a description in natural language into such a string by using a suitable coding, that only changes the length of descriptions by a small constant factor. Indeed, the theory described in this chapter applies to anything that can be represented or coded by binary strings, which includes many familiar mathematical objects such as numbers, sets, and graphs, but also objects such as DNA strings or texts in any language.

many of which are discussed in the book by Li and Vitányi [18]. We will briefly discuss applications in section 11.

We will not go into the delicate subject of the history of Kolmogorov complexity, other than saying that it was invented by Solomonoff, Kolmogorov, and Chaitin (in that order), and we refer to [18] and [9] for further information.

6. Martin-Löf randomness

The notion of Martin-Löf randomness, introduced by Martin-Löf in [20], is based on classical probability theory, which in its modern formulation is phrased in terms of *measure theory*. In appendix A the notion of a measure space is explained in some detail, but for now we keep the discussion as light as possible.

The unit interval $[0, 1]$ consists of all the numbers on the real line between 0 and 1. We wish to discuss probabilities in this setting by assigning to subsets A of the unit interval, called *events*, a probability, which informally should be the probability that when we "randomly" pick a real from $[0, 1]$ that we end up in A. The *uniform* or *Lebesgue measure* on $[0, 1]$ assigns the measure $b-a$ to every interval $[a, b]$, i.e. the measure of an interval is simply its length. For example, the interval $[0, \frac{1}{2}]$ has measure $\frac{1}{2}$, the interval $[\frac{3}{4}, 1]$ has measure $\frac{1}{4}$. Note that $[0, 1]$ itself has measure 1.

Given this, we can also define the measure of more complicated sets by considering combinations of intervals. For example, we give the combined event consisting of the union of the intervals $[0, \frac{1}{2}]$ and $[\frac{3}{4}, 1]$ the measure $\frac{1}{2} + \frac{1}{4} = \frac{3}{4}$. Since the measures of the subsets of $[0, 1]$ defined in this way satisfy the laws of probability (cf. appendix A), we can think of them as *probabilities*.

A series of intervals is called a *cover* for an event A if A is contained in the union of all the intervals in the series. Now an event A is defined to have measure 0 if it is possible to cover A with intervals in such a way that the total sum of the lengths of all the intervals can be chosen arbitrarily small.

For example, for every real x in $[0, 1]$, the event A consisting only of the real x has measure 0, since for every n, x is contained in the interval $[x - \frac{1}{n}, x + \frac{1}{n}]$, and the length of the latter interval is $2\frac{1}{n}$, which tends to 0 if n tends to infinity.

These definitions suffice to do probability theory on $[0, 1]$, and to speak informally about picking reals "at random", but we now wish to define what it means for a *single* real x to be random. We can view any event of measure 0 as a "test for randomness", where the elements

not included in the event pass the test, and those in it fail. All the usual statistical laws, such as the law of large numbers, correspond to such tests. Now we would like to define x to be random if x passes all statistical tests, i.e. x is not in any set of measure 0. But, as we have just seen in the example above, every single real x has measure 0, hence in its full generality this definition is vacuous. (The reader may compare this to the situation we already encountered above in section 3 when we discussed Kollektiv's.)

However, as Martin-Löf observed, we obtain a viable definition if we restrict ourselves to a countable collection of measure 0 sets. More precisely, let us say that an event A has *effective measure 0* if there is a *computable* series of covers of A, with the measure of the covers in the series tending to 0. Phrased more informally: A has effective measure 0 if there is an *algorithm* witnessing that A has measure 0, by producing an appropriate series of covers for A. Now we can finally define:

Definition 6.1. A real x is *Martin-Löf random* if x is not contained in any event of effective measure 0.

It can be shown that with this modification random reals exist.[12] Moreover, *almost every real in* $[0, 1]$ *is random*, in the sense that the set of nonrandom reals is of effective measure 0.

Note the analogy between Definition 6.1 and the way that Church modified von Mises definition of Kollektiv, as described in section 3: There we restricted to the computable selection functions, here we restrict to the effective measure 0 events.

Identifying a real number x with its decimal expansion,[13] we have thus obtained a definition of randomness for infinite sequences. The question now immediately presents itself what the relation, if any, of this definition is with the definition of randomness of *finite* sequences from section 5. A first guess could be that an infinite sequence is random in the sense of Martin-Löf if and only if all of its finite initial segments are random in the sense of Kolmogorov, but this turns out to be false. A technical modification to Definition 5.1 is needed to make this work.

A string y is called a *prefix* of a string y' if y is an initial segment of y'. For example, the string 001 is a prefix of the string 001101. Let us now

[12]The proof runs as follows: There are only countably many algorithms, hence there are only countably many events of effective measure 0, and in measure theory a countable collection of measure 0 sets is again of measure 0.

[13]We ignore here that decimal expansions in general are not unique, for example $0,999\ldots = 1,000\ldots$, but this is immaterial.

impose the following restriction on descriptions: If $U(y) = x$, i.e. y is a description of x, and $U(y') = x'$, then we require that y is not a prefix of y'. This restriction may seem arbitrary, but we can motivate it as follows. Suppose that we identify persons by their phone numbers. It is then a natural restriction that no phone number is a prefix of another, since if the phone number y of x were a prefix of a phone number y' of x', then when trying to call x' we would end up talking to x. Indeed, in practice phone numbers are not prefixes of one another. We say that the set of phone numbers is *prefix-free*. We now require that the set of descriptions y used as inputs for the universal machine U in Definition 5.1 is prefix-free. Of course, this changes the definition of the complexity function $C(x)$: Since there are fewer descriptions available, in general the descriptive complexity of strings will be higher. The complexity of strings under this new definition is called the *prefix-free complexity*. The underlying idea of the prefix-free complexity is the same as that of Kolmogorov complexity, but technically the theory of it differs from Kolmogorov complexity in several important ways. For us, at this point of the discussion, the most important feature of it is the following landmark result. It was proven in 1973 by Claus-Peter Schnorr, one of the pioneers of the subject.

Theorem 6.2 (Schnorr [24])**.** *An infinite sequence X is Martin-Löf random if and only if there is a constant c such that every initial segment of X of length n has prefix-free complexity at least $n - c$.*

The reader should take a moment to let the full meaning and beauty of this theorem sink in. It offers no less than an equivalence between two seemingly unrelated theories. One is the theory of randomness for finite sequences, based on descriptive complexity, and the other is the theory of infinite sequences, based on measure theory. The fact that there is a relation between these theories at all is truly remarkable.

7. Martingales

Thus far we have seen three different formalizations of intuitions underlying randomness:

(i) Mises–Wald–Church randomness, formalizing unpredictability using selection functions,
(ii) Kolmogorov complexity, based on descriptive complexity,
(iii) Martin-Löf randomness, based on measure theory.

Theorem 6.2 provided the link between (ii) and (iii), and (i) was discussed in section 3. We already mentioned Ville, who showed that the notion in (i) was flawed in a certain sense. Ville also showed an

alternative way to formalize the notion of unpredictability of an infinite sequence, using the notion of a *martingale*, which we now discuss.[14] Continuing our game-theoretic discussion of section 3, we imagine that we are playing against an unknown infinite binary sequence X. At each stage of the game, we are shown a finite initial part

$$X(0), X(1), X(2), \ldots, X(n-1)$$

of the sequence X, and we are asked to bet on the next value $X(n)$. Suppose that at this stage of the game, we have a capital of d dollar. Now we may split the amount d into parts b_0 and b_1, and bet the amount b_0 that $X(n)$ is 0, and the amount b_1 that $X(n)$ is 1. After placing our bets, we receive a payoff $d_0 = 2b_0$ if $X(n) = 0$, and a payoff $d_1 = 2b_1$ if $X(n) = 1$. Hence the payoffs satisfy the equation

$$\frac{d_0 + d_1}{2} = d. \tag{1}$$

After placing our bets, we receive a payoff d_0 if $X(n) = 0$, and a payoff d_1 if $X(n) = 1$.

For example, we may let $b_0 = b_1 = \frac{1}{2}d$, in which case our payoff will be d, no matter what $X(n)$ is. So this is the same as not betting at all, and leaving our capital intact. But we can also set $b_0 = d$ and $b_1 = 0$. In this case, if $X(n) = 0$ we receive a payoff of $2d$, and we have doubled our capital. However, if it turns out that $X(n) = 1$, we receive 0, and we have lost everything. Hence this placement of the bets should be made only when we are quite sure that $X(n) = 0$. Any other placement of bets between these two extremes can be made, reflecting our willingness to bet on $X(n) = 0$ or $X(n) = 1$.

After betting on $X(n)$, the value $X(n)$ is revealed, we receive our payoff for this round, and the game continues with betting on $X(n+1)$.

Now the idea of Ville's definition is that we should not be able to win an infinite amount of money by betting on a random sequence. For a given binary string σ, let $\sigma\hat{}0$ denote the string σ extended by a 0, and $\sigma\hat{}1$ the string σ extended by a 1. Formally, a *martingale* is a function d such that for every finite string σ the martingale equality

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

[14]The word "martingale" comes from gambling theory, where it refers to the very dangerous strategy of doubling the stakes in every round of gambling, until a win occurs. With the stakes growing exponentially, if the win does not occur quickly enough, this may result in an astronomical loss for the gambler. In modern probability theory, the word "martingale" refers to a betting strategy in general.

holds. The meaning of this equation is that when we are seeing the initial segment σ, and we have a capital $d(\sigma)$, we can bet the amount $\frac{1}{2}d(\sigma\widehat{\ }0)$ that the next value will be a zero, and $\frac{1}{2}d(\sigma\widehat{\ }1)$ that the next value will be a one, just as above in equation (1). Thus the martingale d represents a particular *betting strategy*. Now for a random sequence X, the amounts of capital

$$d\big(X(0),\ldots,X(n-1)\big)$$

that we win when betting on X should not tend to infinity.[15]

As in the case of Mises–Wald–Church randomness and the case of Martin-Löf randomness, this definition only makes sense when we restrict ourselves to a countable class of martingales.[16] A natural choice would be to consider the *computable* martingales. The resulting notion of randomness was studied in Schnorr [23], and it turns out to be *weaker* than Martin-Löf randomness. However, *there exists another natural class of martingales, the so-called* c.e.-martingales,[17] *such that the resulting notion of randomness is equivalent to Martin-Löf randomness.*

Thus Ville's approach to formalizing the notion of unpredictability using martingales gives yet a third equivalent way to define the same notion of randomness.

8. Randomness and provability

By Gödel's incompleteness theorem (see section 4), in any reasonable formal system of arithmetic, there exist formulas that are true yet unprovable. A consequence of this result is that there is no algorithm to decide the truth of arithmetical formulas. It follows from the undecidability of the Halting Problem (see Definition 4.1) that the set of formulas that are *provable* is also undecidable.[18] However, the set of provable formulas is *computably enumerable*, meaning that there is an

[15]Ville showed that martingales provide an alternative, game-theoretic, formulation of measure theory: The sets of measure 0 are precisely the sets on which a martingale can win an infinite amount of money.

[16]Note that for every sequence X there is a martingale that wins an infinite amount of capital on X: just set $d(X(0)\ldots X(n-1)\widehat{\ }i) = 2d(X(0)\ldots X(n-1))$, where $i = X(n)$. However, in order to play this strategy, one has to have full knowledge of X.

[17]C.e. is an abbreviation of "computably enumerable". This notion is further explained in section 8.

[18]This follows by the method of arithmetization: Statements about Turing machines can be translated into arithmetic by coding. If the set of provable formulas were decidable, it would follow that the Halting Problem is also decidable.

algorithm that lists all the provable statements. Computably enumerable, or *c.e.*, sets, play an important role in computability theory. For example, the set H representing the Halting Problem is an example of a c.e. set, because we can in principle make an infinite list of all the halting computations.[19] The complement $\overline{H}$ of the set H, consisting of all nonconvergent computations, is *not* c.e. For if it were, we could decide membership in H as follows: Given a pair M and x, effectively list both H and its complement $\overline{H}$ until the pair appears in one of them, thus answering the question whether the computation $M(x)$ converges. Since H is not computable, it follows that $\overline{H}$ cannot be c.e. Because the set of all provable statements is c.e., it also follows that not all statements of the form

$$\text{“}M(x) \text{ does not halt”}$$

are provable. Hence there exist computations that do not halt, but for which this fact is not provable! Thus we obtain a specific example of a true, but unprovable statement. The same kind of reasoning applies if we replace H by any other noncomputable c.e. set.

Now consider the set R of all strings that are Kolmogorov random, and let non-R be the set of all strings that are not Kolmogorov random. We have the following facts:

(i) non-R is c.e. This is easily seen as follows: If x is not random, there is a description y shorter than x such that $U(y) = x$. Since the set of halting computations is c.e., it follows that non-R is also c.e.

(ii) R is not c.e. This is proved in Theorem B.1 in appendix B.

By applying the same reasoning as for H above, we conclude from this that there are statements of the form

$$\text{“}x \text{ is random”}$$

that are true, but not provable. This is Chaitin's version of the incompleteness theorem [5].[20]

[19]We can do this by considering all possible pairs of Turing machines M and inputs x, and running all of them in parallel. Every time we see a computation $M(x)$ converge, we add it to the list. Note, however, that we cannot list the converging computations in order, since there is no way to predict the running time of a converging computation. Indeed, if we could list the converging computations in order, the Halting Problem would be decidable.

[20]As for Gödel's incompleteness theorem, the statement holds for any reasonable formal system that is able to express elementary arithmetic. In fact, it follows from Theorem B.1 that any such system can prove the randomness of at most *finitely* many strings.

9. Other notions of randomness

Mises–Wald–Church random sequences were defined using computable selection functions, and Martin-Löf random sequences with computable covers, which in Ville's approach correspond to c.e.-martingales. As Wald already pointed out in the case of Kollektiv's, all of these notions can be defined relative to any countable collection of selection functions, respectively covers and martingales. Choosing computable covers in the case of Martin-Löf randomness gave the fundamental and appealing connection with Kolmogorov randomness (Theorem 6.2), but there are situations in which this is either too weak, or too strong. Viewing the level of computability of covers and martingales as a parameter that we can vary allows us to introduce notions of randomness that are either weaker or stronger than the ones we have discussed so far.

In his groundbreaking book [23], Schnorr discussed alternatives to the notion of Martin-Löf randomness, thus challenging the status of this notion (not claimed by Martin-Löf himself) as the "true" notion of randomness.[21]

In studying the randomness notions corresponding to various levels of computability, rather than yielding a single "true" notion of randomness, a picture has emerged in which every notion has a corresponding context in which it fruitfully can be applied. This ranges from low levels of complexity in computational complexity theory (see e.g. the survey paper by Lutz [19]), to the levels of computability (computable and c.e.) that we have been discussing in the previous sections, to higher levels of computability, all the way up to the higher levels of set theory. In studying notions of randomness across these levels, randomness has also served as a unifying theme between various areas of mathematical logic.

Chaitin also drew a number of dubious philosophical conclusions from his version of the incompleteness theorem, that were adequately refuted by van Lambalgen [17], and later in more detail by Ratikaainen, Franzen, Porter, and others. Unfortunately, this has not prevented Chaitin's claims from being widely publicized.

[21]After Martin-Löf's paper [20], the notion of Martin-Löf randomness became known as a notion of "computable randomness". As Schnorr observed, this was not quite correct, and for example the characterization with c.e.-martingales pointed out that is was more apt to think of it as "c.e.-randomness". To obtain a notion of "computable randomness", extra computational restrictions have to be imposed. Schnorr did this by basing his notion on Brouwer's notion of constructive measure zero set. The resulting notion of randomness, nowadays called Schnorr randomness, has become one of the standard notions in randomness theory, see [9].

The general theory also serves as a background for the study of specific cases. Consider the example of π. Since π is a computable real number, its decimal expansion is perfectly predictable, and hence π it is not random in any of the senses discussed above. However, the distribution of the digits $0, \ldots, 9$ in π appears to be "random". Real numbers with a decimal expansion in which every digit occurs with frequency $\frac{1}{10}$, and more general, every block of digits of length n occurs with frequency $\frac{1}{10^n}$, are called *normal* to base 10. Normality can be seen as a very weak notion of randomness, where we consider just one type of statistical test, instead of infinitely many as in the case of Martin-Löf randomness. It is in fact not known if π is normal to base 10, but it is conjectured that π is indeed "random" in this weak sense. For a recent discussion of the notion of normality, see Becher and Slaman [3].

10. Pseudorandom number generators and complexity theory

In many contexts, it is desirable to have a good source of random numbers, for example when one wants to take an unbiased random sample, in the simulation of economic or atmospheric models, or when using statistical methods to estimate things that are difficult to compute directly (the so-called Monte Carlo method). In such a case, one may turn to physical devices (which begs the question about randomness of physical sources), or one may try to generate random strings using a computer. However, the outcome of a deterministic procedure on a computer cannot be random in any of the senses discussed above. (By Theorem B.1 in appendix B, there is no purely algorithmic way of effectively generating infinitely many random strings, and it is easy to see that a Martin-Löf random set cannot be computable.) Hence the best an algorithm can do is to produce an outcome that is *pseudorandom*, that is, "random enough", where the precise meaning of "random enough" depends on the context. In practice this usually means that the outcome should pass a number of standard statistical tests. Such procedures are called *pseudorandom number generators*. That the outcomes of a pseudorandom number generator should not be taken as truly random was pointed out by the great mathematician and physicist John von Neumann, when he remarked that

> Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.[22]

[22]The Monte Carlo method was first used extensively in the work of Ulam and von Neumann on the hydrogen bomb.

Randomized algorithms are algorithms that employ randomness during computations, and that allow for a small probability of error in their answers. For example, the first feasible[23] algorithms to determine whether a number is prime were randomized algorithms.[24] An important theme in computational complexity theory is the extent to which it is possible to *derandomize* randomized algorithms, i.e. to convert them to deterministic algorithms. This is connected to fundamental open problems about the relation between deterministic algorithms, nondeterministic algorithms, and randomized computation.[25] Besides being of theoretical interest, this matter is of great practical importance, for example in the security of cryptographic schemes that are currently widely used. For an overview of current research we refer the reader to Arora and Barak [2]. It is also interesting to note that randomness plays an important part in many of the proofs of results about deterministic algorithms, that do not otherwise mention randomness.

11. Applications

As pointed out in section 5 and Corollary B.2, due to the undecidability of the Halting Problem, the notion of Kolmogorov complexity is inherently noncomputable. This means that there is no algorithm that, given a finite sequence, can compute its complexity, or decide whether it is random or not. Can such a concept, apart from mathematical and philosophical applications, have any *practical* applications? Perhaps surprisingly, the answer is "yes". A large number of applications, ranging from philosophy to physics and biology, is discussed in the monograph by Li and Vitányi [18]. Instead of attempting to give an overview of all applications, for which we do not have the space, we give an example of one striking application, namely the notion of information distance. Information distance is a notion built on Kolmogorov complexity that was introduced by Bennett et al. [4]. It satisfies the properties of a metric (up to constants), and it gives a well-defined

[23]In computational complexity theory, an algorithms is considered feasible if it works in *polynomial time*, that is, if on an input of length n it takes n^k computation steps for some fixed constant k.

[24]Since 2001 there also exist *deterministic* feasible algorithms to determine primality [1], but the randomized algorithms are still faster, and since their probability of error can be made arbitrary small, in practice they are still the preferred method.

[25]The question about derandomization is embodied in the relation between the complexity classes P and BPP, see [2]. This is a probabilistic version of the notorious P versus NP problem, which is about determinism versus nondeterminism. The latter is one of the most famous open problems in mathematics.

notion of distance between arbitrary pairs of binary strings. The computational status of information distance (and its normalized version) was unclear for a while, but as the notion of Kolmogorov complexity itself it turned out to be noncomputable [25]. However, it is possible to *approximate* the ideal notion using existing, computable, compressors. This gives a computable approximation of information distance, that can in principle be applied to any pair of binary strings, be it musical files, the genetic code of mammals, or texts in any language. By computing the information distance between various files from a given domain, one can use the notion to classify anything that can be coded as a binary string. The results obtained in this way are startling. E.g. the method is able to correctly classify pieces of music by their composers, animals by their genetic code, or languages by their common roots, purely on the basis of similarity of their binary encodings, and without any expert knowledge. Apart from these applications, the notion of information distance is an example of a *provably* intractable notion, which nevertheless has important practical consequences. This provides a strong case for the study of such theoretical notions.

Appendix A. Measure and probability

A *measure space* is a set X together with a function μ that assigns positive real values $\mu(A)$ to subsets A of X, such that the following axioms are satisfied:

(i) The empty set $\emptyset$ has measure 0.
(ii) If $A \cap B = \emptyset$, then $\mu(A \cup B) = \mu(A) + \mu(B)$. That is, if A and B are disjoint sets then the measure of their union is the sum of their measures.[26]

If also $\mu(X) = 1$ we can think of the values of μ as *probabilities*, and we call X a *probability space*, and μ a *probability measure*. If A is a subset of X, we think of $\mu(A)$ as the probability that a randomly chosen element of X will be in the set A. Subsets of X are also called *events*. In this setting the axioms (i) and (ii) are called the *Kolmogorov axioms* of probability. The axioms entail for example that if $A \subseteq B$, i.e. the event A is contained in B, that then $\mu(A) \leqslant \mu(B)$.

An important example of a probability space consists of the unit interval $[0, 1]$ of the real line. The *uniform* or *Lebesgue* measure on $[0, 1]$ is defined by assigning to every interval $[a, b]$ the measure $b - a$,

[26]It is in fact usually required that this property also holds for countably infinite collections.

i.e. the *length* of the interval. The measure of more complicated sets can be defined by considering combinations of intervals.[27]

Appendix B. The noncomputability of the complexity function

In Zvonkin and Levin [31] the following results are attributed to Kolmogorov.

Theorem B.1. *The set R of Kolmogorov random strings does not contain any infinite c.e. set.*[28] *In particular, R itself is not c.e.*

Proof. Suppose that A is an infinite c.e. subset of R. Consider the following procedure. Given a number n, find the first string a enumerated in A of length greater than n. Note that such a string a exists since A is infinite. Since a is effectively obtained from n, n serves as a description of a, and hence the Kolmogorov complexity $C(a)$ is bounded by the length of n, which in binary notation is roughly $\log n$ (plus a fixed constant c independent of n, needed to describe the above procedure), where log denotes the binary logarithm. So we have that $C(a)$ is at most $\log n$. But since a is random (because it is an element of A, which is a subset of R), we also have that $C(a)$ is at least the length of a, which we chose to be greater than n. In summary, we have $n \leqslant C(a) \leqslant \log n + c$, which is a contradiction for sufficiently large n. □

Corollary B.2. *The complexity function C is not computable.*

Proof. If C were computable, we could generate an infinite set of random strings, contradicting Theorem B.1. □

References

[1] M. Agrawal, N. Kayal, and N. Saxena, *PRIMES is in P*, Annals of Mathematics 160(2) (2004) 781–793.

[2] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*, Cambridge University Press, 2009.

[3] V. Becher and T. A. Slaman, *On the normality of numbers in different bases*, Journal of the London Mathematical Society 90(2) (2014) 472–494.

[27] The definition of a probability measure on the unit interval $[0, 1]$ that assigns a probability to *all* subsets of it is fraught with technical difficulties that we will not discuss here. This problem, the so-called *measure problem*, properly belongs to the field of set theory, and has led to deep insights into the nature of sets and their role in the foundation of mathematics (cf. Jech [13]).

[28] R itself is infinite, but by the theorem there is no way to effectively generate infinitely many elements from it. Such sets are called *immune*.

[4] C. H. Bennett, P. Gács, M. Li, P. M. B. Vitányi, W. Zurek, *Information distance*, IEEE Trans. Inform. Theory 44(4) (1998) 1407–1423.
[5] G. J. Chaitin, *Information-theoretic limitations of formal systems*, Journal of the ACM 21 (1974) 403–424.
[6] A. Church, *On the concept of a random sequence*, Bulletin of the American Mathematical Society 46 (1940) 130-135.
[7] B. J. Copeland, *The modern history of computing*, The Stanford Encyclopedia of Philosophy (Fall 2008 Edition).
[8] P. G. Doyle, *Maybe there's no such thing as a random sequence*, manuscript, 2011.
[9] R. G. Downey and D. R. Hirschfeldt, *Algorithmic Randomness and Complexity*, Springer-Verlag, 2010.
[10] R. Downey, D. R. Hirschfeldt, A. Nies, and S. A. Terwijn, *Calibrating randomness*, Bulletin of Symbolic Logic 12(3) (2006) 411–491.
[11] H. Freudenthal, *Realistic models in probability*, in: I. Lakatos (ed.), Problems in inductive logic, North-Holland, 1969.
[12] K. Gödel, *The Consistency of the Continuum-Hypothesis*, Princeton University Press, 1940.
[13] T. Jech, *Set Theory*, 3rd millennium edition, Springer-Verlag, 2003.
[14] A. N. Kolmogorov, *Grundbegriffe der Wahrscheinlichkeitsrechnung*, Springer, 1933.
[15] K. Kunen, *Set Theory: An Introduction to Independence Proofs*, North-Holland, 1980.
[16] M. van Lambalgen, *Random Sequences*, PhD thesis, University of Amsterdam, 1987.
[17] M. van Lambalgen, *Algorithmic information theory*, Journal of Symbolic Logic 54(4) (1989) 1389–1400.
[18] M. Li and P. Vitányi, *An Introduction to Kolmogorov Complexity and its Applications*, 3rd edition, Springer-Verlag, 2008.
[19] J. H. Lutz, *The quantitative structure of exponential time*, in: L. A. Hemaspaandra and A. L. Selman (eds.), Complexity Theory Retrospective II, Springer-Verlag, 1997, 225–254.
[20] P. Martin-Löf, *The definition of random sequences*, Information and Control 9 (1966) 602–619.
[21] A. Nies, *Computability and Randomness*, Oxford University Press, 2009.
[22] P. G. Odifreddi, *Classical Recursion Theory*, Vol. 1, Studies in Logic and the Foundations of Mathematics Vol. 125, North-Holland, 1989.
[23] C. P. Schnorr, *Zufälligkeit und Wahrscheinlichkeit*, Lecture Notes in Mathematics 218, Springer-Verlag, 1971.
[24] C. P. Schnorr, *Process complexity and effective random tests*, Journal of Computer and System Sciences 7 (1973) 376–388.
[25] S. A. Terwijn, L. Torenvliet, and P. M. B. Vitányi, *Nonapproximability of the normalized information distance*, Journal of Computer and System Sciences 77 (2011) 738–742.
[26] A. M. Turing. *On computable numbers with an application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society 42 (1936) 230–265. Correction in Proceedings of the London Mathematical Society 43 (1937) 544–546.

[27] J. Ville, *Étude critique de la notion de collectif*, Monographies des Probabilités, Calcul des Probabilités et ses Applications, Gauthier-Villars, 1939.
[28] R. von Mises, *Grundlagen der Wahrscheinlichkeitsrechnung*, Mathematische Zeitschrift 5 (1919) 52-99
[29] A. Wald, *Sur la notion de collectif dans la calcul des probabilités*, Comptes Rendus des Seances de l'Académie des Sciences 202 (1936) 180-183
[30] A. Wald, *Die Wiederspruchsfreiheit des Kollektivbegriffes der Wahrscheinlichkeitsrechnung*, Ergebnisse eines Mathematischen Kolloquiums 8 (1937) 38-72.
[31] A. K. Zvonkin and L. A. Levin, *The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms*, Russian Mathematical Surveys 25(6) (1970), 83–124.

(Sebastiaan A. Terwijn) Radboud University, Department of Mathematics, P.O. Box 9010, 6500 GL Nijmegen, the Netherlands.
E-mail address: terwijn@math.ru.nl