Notions and applications of algorithmic randomness

Stijn Vermeeren

Submitted in accordance with the requirements
for the degree of Doctor of Philosophy

The University of Leeds
School of Mathematics

March 2013

The candidate confirms that the work submitted is his own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Chapter 5 consists mostly of joint work with Laurent Bienvenu, Andrei Romashchenko, Alexander Shen and Antoine Taveneaux. The material will be published in [4]. Initial investigations on the topic were made by Shen. The research was then considerably widened when the other authors (including the candidate) got involved. Most core results were obtained by all authors together during a three week period of collaboration in France in November 2011. Afterwards, the candidate proved another theorem himself (Theorem 5.3.3), while the other authors contributed additional work as well.

Acknowledgements

Thanks to my parents for their continuous support for my studies away from home. Thanks to my supervisors, S. Barry Cooper and Andy E. M. Lewis, for being always available and helpful, while also leaving me enough freedom to discover and pursue my own interests. Thanks to my fellow PhD students for their friendship, knowledge, but most of all for the sense of not being *in it* alone. Thanks to Laurent Bienvenu, Andrei Romashchenko, Alexander Shen and Antoine Taveneaux for a fruitful three weeks of collaborating in France. Thanks to the University of Leeds for providing me with the University Research Scholarship that funded this PhD. Thanks to the School of Mathematics and to the Association for Symbolic Logic for providing me with funding to attend conferences all over the world. Thanks to my examiners Michael Rathjen and Wolfgang Merkle for their valuable corrections.

This thesis is dedicated to the memory of Graham Connell and to the Leeds University Union Hiking Club.

Abstract

Algorithmic randomness uses computability theory to define notions of *randomness* for infinite objects such as infinite binary sequences. The different possible definitions lead to a hierarchy of randomness notions. In this thesis we study this hierarchy, focussing in particular on Martin-Löf randomness, computable randomness and related notions. Understanding the relative strength of the different notions is a main objective. We look at proving implications where they exists (Chapter 3), as well as separating notions when the are not equivalent (Chapter 4). We also apply our knowledge about randomness to solve several questions about provability in axiomatic theories like Peano arithmetic (Chapter 5).

Contents

List of Figures

Chapter 1

Introduction

With some infinite sequences of zeroes and ones, such as

$$0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\ldots, \tag{1}$$

we immediately recognize that they satisfy a pattern or have a regularity. Other sequences appear to follow no pattern at all, and we would call them *random*. How can we turn this intuitive dichotomy into a rigorous mathematical notion of *randomness*?

It is important to realize that we are looking for a notion that is much stronger than incomputability. A sequence that is incomputable on its odd positions and has a zero on every even position is still incomputable. However, having a zero on every even position is a very strong pattern, so this sequence is certainly not random.

Randomness as it is used in statistics does not help us. Even though we will feel very suspicious when we see the sequence (1) appear as the result of a repeated coin toss (writing '0' for *heads* and '1' for *tails*), from a probabilistic point of view this sequence of outcomes isn't less probable

than any other. Statistical randomness is a notion that applies to variables or processes. However, it does not give us a sensible notion for randomness of individual sequences of zeroes and ones.

Computability theory provides the solution. Algorithmic randomness uses computability theory in various ways to come up with mathematical definitions of what exactly is a *regularity* in a sequence, i.e. which sequences are random and which ones are not. Some of these definitions turn out to be equivalent. However, often the notions defined by these definitions have (sometimes very subtle) differences between them. A whole hierarchy of different randomness notions appears. Many aspects of this hierarchy are not well understood yet. In this thesis, I have studied the properties of and the relations between randomness notions, focussing in particular on *computable randomness* and its variations. Additionally, the final chapter explores some fascinating interactions between randomness and provability.

The principal new results in this thesis are

- the notion of *weak Church stochasticity* as defined in Section 3.1 and further investigated in Section 3.7;
- the remarks on the problem of base-invariance of partial computable randomness in Section 3.5, in particular Theorem 3.5.3;
- the proof of Theorem 4.2.1, providing a direct construction of a sequence that is partial computably random but not total injection random;
- Chapter 5, which is joint work with Laurent Bienvenu, Andrei Romashchenko, Alexander Shen and Antoine Taveneaux. My most distinctive personal contribution to this work is Theorem 5.3.3.

Aside from presenting new results, I have also made an effort to give a

clear presentation of a good amount of background material, in particular on the randomness and stochasticity notions in Figures 6 and 9, and on the implications that exist between them. I hope that this will be of value, since these results tend to be rather scattered in the available books on algorithmic randomness. Some basic remarks, such as why *blind computable randomness* is not a sensible randomness notion (Remark 3.4.4), don't even appear in the literature. Surely this is not because nobody has thought about these questions; I rather suspect that people have just found these observations to elementary to include them in their research papers. Still, these remarks are certainly not trivial, so I've taken the opportunity to present them rigorously in my thesis. This will hopefully serve as a useful reference for future researchers in algorithmic randomness.

During the first years of my PhD, two books appeared on the subject of algorithmic randomness: *Computability and Randomness* by André Nies [48] and *Algorithmic Randomness and Complexity* by Rodney Downey and Denis Hirschfeldt [16]. These monographs have collected an invaluable amount of material that was previously scattered across many publications or not accessible at all. It is likely that I would not have started research on algorithmic randomness at all without these two fabulous resources available to ease my path into the subject. As both books will undoubtedly remain an essential resource for generations to come, I have included numerous references to them in this thesis. Where appropriate, I have provided results not only with a reference to their original publication, but also with references to the corresponding theorems or sections in Nies and/or Downey and Hirschfeldt.

Two more historical resources that are particularly interesting and deserve to be mentioned here are Jean Ville's 1939 PhD thesis [60] and Claus-Peter Schnorr's 1971 book *Zufälligkeit und Wahrscheinlichkeit* [53]. Both docu-

ments can be downloaded for free on the internet, if you know where to look for them; I've included links in the bibliography of this thesis.

In this thesis, I generally use the pronoun 'we', as if the reader and me are going through the mathematics together. For expressing my personal opinions and for explaining certain decisions, however, I use the pronoun 'I' (such as in this paragraph).

Chapter 2

Cantor space and computability theory

A basic knowledge of mathematical logic, computability theory and topology will be assumed in this thesis. This chapter introduces a lot of background material, but due to space constraints this can be no replacement for proper textbooks such as [40], [15] and [47]. The principal aim of this chapter is to establish terminology and notation. Some extra attention will be paid to a couple of results that will have a key role further on in this thesis.

2.1 Basic notation and terminology

$\mathbb{N} = \{0, 1, 2, \ldots\}$ is the set of nonnegative integers or natural numbers. $\mathbb{Q}$ is the set of rational numbers and $\mathbb{R}$ is the set of real numbers.

If A and B are sets, then B^A is the set of all **functions** with domain A and codomain B. If $f : A \to B$ is such a function and $C \subseteq A$, then $f\restriction_C : C \to B$ is the **restriction** of f to the domain C.

A **string** is a finite sequence of symbols, which are elements of a fixed

finite set. We will mostly work with binary strings, where the only symbols are 0 and 1. Indeed, when we don't specify anything to the contrary, *string* will always mean *binary string*. A string σ can be seen as a function $\{0, 1, \ldots, n-1\} \to \{0, 1\}$ from some finite initial segment of the natural numbers to the set of symbols. The symbol at position i ($i \in \operatorname{dom}\sigma$) is then $\sigma(i)$. The **length** $|\sigma|$ of a string σ is the number of symbols, i.e. the size of the domain of σ. There is a unique string of length 0, called the **empty string**, which we denote by λ.

Two strings σ and τ can be **concatenated** to form a longer string $\rho = \sigma\tau$, consisting of the symbols of σ followed by the symbols of τ. We say that σ is an **initial segment** or **prefix** of ρ and that ρ **extends** σ with **extension** τ. If ρ extends σ, then we also write $\sigma \preccurlyeq \rho$ and if moreover $\sigma \neq \rho$ then we write $\sigma \prec \rho$. This gives a partial order on the set of all strings. Two strings are called **comparable** if one extends the other, otherwise they are **incomparable**. A set of pairwise incomparable strings is called an **antichain** or a **prefix-free set of strings**. With σ^n we denote the concatenation of n copies of the string σ. For example, 0^n is the string consisting of n zeroes.

The term **sequence** will be used for infinite sequences of symbols, again usually 0 and 1. Hence, a sequence can be seen as function $\mathbb{N} \to \{0, 1\}$. A sequence Z **extends** a string σ, written $\sigma \prec Z$, if σ is an initial segment of Z. The set of all sequences that extend some string σ is written as $[\![\sigma]\!]$. Also, if X is a set of strings, then we write $[\![X]\!] = \cup_{\sigma \in X} [\![\sigma]\!]$.

If (σ_i) is a sequence of strings such that σ_j extends σ_i whenever $j \geq i$, and $\lim_{i\to\infty} |\sigma_i| = \infty$, then we define $\lim_{i\to\infty} \sigma_i$ to be the sequence Z with $Z(k) = \sigma_i(k)$ for i sufficiently large.

In axiomatic set theory, it is customary to define each natural number as the set of all smaller natural numbers, that is $n = \{0, 1, \ldots, n-1\}$. The

symbol ω is also used for the set of all natural numbers. We will use this convention to introduce a concise notation for our purposes. For example, when no confusion with the integer 2^n is possible, 2^n will signify $\{0,1\}^{\{0,1,\ldots,n-1\}}$, i.e. the set of all strings of length n. The set of all infinite sequences is written as 2^ω. The notation $2^{\leq n}$ is used for the set $\cup_{i=0}^{n} 2^i$ of all strings of length less than or equal to n. Likewise $2^{<\omega}$ is used for the set $\cup_{i\in\mathbb{N}} 2^i$ of all strings of any length. If x is a sequence or a string of length at least n, then $x\restriction_n$ is the restriction of x to the domain $\{0,1,\ldots,n-1\}$, i.e. is the initial segment of x of length n. The new sequence that we obtain by removing some initial segment from a sequence Z, i.e. a sequence of the form $Z\restriction_{[n,\infty)}$, is called a **tail** of Z.

2.2 Cantor space and measure theory

The set of all sequences 2^ω is known as **Cantor space**.

A fundamental lemma that applies to Cantor space is **König's lemma** [28]. This lemma says that for any infinite, downwards-closed (i.e. closed under taking prefixes) set of strings X, there is a sequence $Z \in 2^\omega$ such that every initial segment of Z is in X. Though in its general form König's lemma famously depends on the Axiom of Choice, this is not the case when we are just considering Cantor space.

Cantor space has a well-studied **standard topology**. The basic open sets or **open cylinders** are of the form $[\![\sigma]\!]$ for all strings σ. For more background on topology, see for example [47].

Cantor space also has a well-studied **standard measure**. On this topic, a more detailed introduction is appropriate. A measure on a set X assigns a nonnegative real number (or possibly infinity) to certain subsets of X,

representing their *size*. A measure μ must satisfy $\mu(\emptyset) = 0$ and must be countably additive, that is

$$\mu\left(\bigsqcup_{i\in\mathbb{N}} A_i\right) = \sum_{i\in\mathbb{N}} \mu(A_i)$$

for pairwise disjoint sets A_i on which the measure is defined. It might not be possible to assign a measure in a suitable way to every subset of X. Therefore, a measure is only defined on a certain σ-algebra Σ of subsets of X. (This means that Σ must contain X itself, and be closed under countable unions and complementation.) We will be interested in measures on Cantor space that are defined on the σ-algebra of Borel sets. A set is Borel if it can be obtained from open cylinders by taking complements, countable unions and countable intersections. This will include in particular all Σ^0_n and Π^0_n classes, as defined in the next section.

By the extension theorems of measure theory, a measure for all Borel sets can be defined by assigning a measure to every open cylinder in a countably additive way. In fact, since every open cylinder is compact, we only need to worry about finite additivity. Since we will need this result further on, I prove it here as a lemma.

Lemma 2.2.1.
Suppose $m : 2^{<\omega} \to \mathbb{R}_{\geq 0}$ *satisfies*

$$m(\sigma) = m(\sigma 0) + m(\sigma 1) \tag{2}$$

for every string σ*. Then there is a unique measure* μ *on the Borel sets such that* $\mu([\![\sigma]\!]) = m(\sigma)$ *for all* σ*.*

Proof. Let $\mathcal{A}$ be the class of all finite unions of pairwise disjoint open cylinders

(including $\emptyset$ as the *empty union*). As $\mathcal{A}$ contains $\emptyset$ and is closed under complements, finite intersections and finite unions, $\mathcal{A}$ is called an *algebra of sets*. We first prove that there exists a unique function $\mu_0 : \mathcal{A} \to \mathbb{R}_{\geq 0}$ (called a *pre-measure*) that is countably additive and satisfies $\mu_0(\emptyset) = 0$ and $\mu_0([\![\sigma]\!]) = m(\sigma)$ for all strings σ.

When $U = \bigsqcup_{i=0}^{n} [\![\sigma_i]\!]$ is a disjoint union of open cylinders, then we certainly must have

$$\mu_0(U) = \sum_{i=0}^{n} m(\sigma_i).$$

So it remains to prove that this μ_0 is indeed a well-defined and countably additive function. Suppose U can also be written as another disjoint union of open cylinders $\bigsqcup_{i=0}^{m} [\![\tau_i]\!]$. Let N be the maximal length of any σ_i or τ_i. Using (2) we have

$$\sum_{i=0}^{n} m(\sigma_i) = \sum_{\substack{\sigma \in 2^N \\ [\![\sigma]\!] \subseteq U}} m(\sigma) = \sum_{i=0}^{m} m(\tau_i),$$

so μ_0 is indeed well-defined. For countable additivity, suppose that

$$U, U_0, U_1, \ldots \in \mathcal{A}$$

and $U = \bigsqcup_{i \in \mathbb{N}} U_i$ in a disjoint union. Then all but finitely many U_i must be the empty set, as U is compact. So we only need to prove finite additivity, which μ_0 satisfies by definition.

Finally, the σ-algebra generated by $\mathcal{A}$ is the σ-algebra of Borel sets. So by the extension theorem from measure theory (often called either *Hahn-Kolmogorov theorem* or *Carathéodory's extension theorem*; see for example [22, Section 13, Theorem A]) there is a unique measure μ on the Borel sets such that $\mu(U) = \mu_0(U)$ for all $U \in \mathcal{A}$. So this is also the unique measure on

the Borel sets that satisfies $\mu([\![\sigma]\!]) = m(\sigma)$ for every string σ, as required. □

From now on, μ will always refer to the **standard measure** that satisfies $\mu([\![\sigma]\!]) = 2^{-|\sigma|}$ for every σ.

Cantor space and the unit interval $[0, 1]$ of the real line are similar in many ways. The function $2^\omega \to [0, 1]$ that maps a sequence Z to the real number with binary expansion $0.Z$ is a continuous and measure preserving surjection. Moreover, only the dyadic rationals have two different binary expansions. If $x \in [0, 1]$, then by $x\restriction_n$ we denote the string that contains the first n digits of the binary expansion of x (taking by convention an expansion with infinitely many zeroes if we have the choice). So, for example, we have

$$x = \lim_{n\to\infty} 0.(x\restriction_n)$$

for $x \in [0, 1]$.

2.3 Computability theory

I will assume that the reader is familiar with the basics of computability theory, including oracle computations, Turing degrees and the arithmetical hierarchy. This section merely serves to fix the notation that will be used in this thesis, and to mention some important results that will be used further on. A complete introduction to computability theory can be found in [15].

A **computable order** is a (total) computable function $h : \mathbb{N} \to \mathbb{N}$ that is nondecreasing and with $\lim_{n\to\infty} h(n) = \infty$. Often, to obtain the notion of a *computable rate of convergence*, we will divide by a computable order. In that case, we will implicitly assume that the order is nowhere equal to 0.

We fix an effective enumeration $\phi_0, \phi_1, \phi_2, \ldots$ of all partial computable

functions (possibly and indeed necessarily with repetitions). We write $\phi(x)\uparrow$ if the function ϕ is undefined on input x, and $\phi(x)\downarrow$ if ϕ is defined on input x. Equality will be used to mean that either both sides are undefined, or both sides are defined and have the same value. We also use the shorthand notation $\phi(x)\downarrow = y$ to mean that $\phi(x)\downarrow$ and $\phi(x) = y$. We use $[s]$ to indicate that all computations are only approximated up to a certain stage s, e.g. $\phi(x)[s]$ might be undefined if the computation for $\phi(x) = y$ takes more than s steps.

An important lemma is the Fixed Point Theorem.

Lemma 2.3.1 (Fixed Point Theorem)**.**
For every computable function $f : \mathbb{N} \to \mathbb{N}$ there exists an $n \in \mathbb{N}$ such that $\phi_{f(n)} = \phi_n$.

A proof can be found in [15, 4.4.1]. Note that the Fixed Point Theorem implies, by e.g. taking $f(n) = n + 1$, that the enumeration (ϕ_i) must have repetitions, as mentioned before.

By letting $W_i = \text{range}(\phi_i)$ for all $i \in \mathbb{N}$ we get an effective enumeration (W_i) of all computable enumerable (c.e.) sets.

In some contexts, especially when defining Kolmogorov complexity, it is customary to speak about (Turing) **machines** rather than about partial computable functions. These machines simply execute a fixed algorithm using a given input and possibly producing an output. Hence machines and computable functions are essentially the same concept. Two machines M and N are called *equivalent* ($M \equiv N$) if they compute the same partial function.

There exists a computable bijection $\langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ that encodes every pair of natural numbers m, n as a single natural number $\langle m, n \rangle$. For example $\langle m, n \rangle = 2^m(2n + 1) - 1$ defines such a **pairing function**. We fix some pairing function $\langle \cdot, \cdot \rangle$ from now on. This also gives us computable encoding

for n-tuples for $n \geq 2$, by defining inductively

$$\langle n_0, \ldots, n_k, n_{k+1} \rangle = \langle \langle n_0, \ldots, n_k \rangle, n_{k+1} \rangle.$$

The preorder $\leq_T$ of Turing reducibility induces an equivalence relation $\equiv_T$ on all the subsets of $\mathbb{N}$. The equivalence classes are called **Turing degrees**. There is a minimal Turing degree $\mathbf{0}$ that contains exactly all computable sets. The Turing degree of the halting problem is denoted by $\mathbf{0}'$ (pronounced **zero prime** or **zero jump**). The n**th jump of the zero degree** is denoted by $\mathbf{0}^{(n)}$. These satisfy

$$\mathbf{0} <_T \mathbf{0}' <_T \mathbf{0}^{(2)} <_T \mathbf{0}^{(3)} <_T \ldots.$$

From this, the Turing degrees might appear to be a simple linear order, but in fact, it is a very complicated non-linear structure. There are many Turing degrees in between $\mathbf{0}$ and $\mathbf{0}'$.

A weak truth table reduction is a Turing reduction with computably bounded use. Hence weak truth table reducibility ($\leq_{wtt}$) is a stronger reduction than Turing reducibility. It induces wtt-degrees that are subsets of the Turing degrees.

The c.e. subsets of $\mathbb{N}$ are also called the Σ^0_1 **sets**. They are exactly the sets of the form

$$\{n \in \mathbb{N} : \exists m \, \phi(m, n)\}$$

where ϕ is a computable predicate, i.e. a total computable function that outputs either 0 for *false* or 1 for *true*. In other words, Σ^0_1 sets can be defined by an existential formula or Σ^0_1 formula. The Π^0_1 **sets** are the complements of the Σ^0_1 sets. These can be defined by a Π^0_1 formula that has just a universal

quantifier in front of a computable predicate. In general, a set is Σ^0_n or Π^0_n is it is definable using a formula with n alternating quantifiers, the first of which is an $\exists$ or a $\forall$ respectively, followed by a computable predicate. Equivalently, $\Sigma^0_{n+1}/\Pi^0_{n+1}$ sets are the Σ^0_1/Π^0_1 sets relative to $\mathbf{0}^{(n)}$, i.e. the predicate in the defining formula is allowed to be $\mathbf{0}^{(n)}$-computable. This hierarchy of sets is called the **arithmetical hierarchy**. It can equally be applied to sets of strings, rational numbers, and so on.

A similar hierarchy can be defined for subsets of Cantor space. A Σ^0_1 **class** is a subset of 2^ω of the form

$$\{Z \in 2^\omega \,:\, \exists m\, \phi(Z\restriction_m)\},$$

where ϕ is again a computable predicate. The Σ^0_1 classes are also called the **effectively open subsets** of Cantor space, and indeed they are open in the topology of Cantor space. A class $\mathcal{A} \subseteq 2^\omega$ is Σ^0_1 if and only if there is a c.e. set of strings X such that $\mathcal{A} = [\![X]\!]$. We say that $\mathcal{A}$ is **generated by** X and that X is a **set of generators** for $\mathcal{A}$.

Complements of Σ^0_1 classes are Π^0_1 **classes**, also called **effectively closed subsets** of Cantor space. More generally, a Σ^0_n **class** is a set of the form

$$\{Z \in 2^\omega \,:\, \exists m_1 \forall m_2 \ldots \phi(Z\restriction_{m_1}, Z\restriction_{m_2}, \ldots, Z\restriction_{m_n})\},$$

where there are n alternating quantifiers and ϕ is a computable predicate. A Π^0_n **class** is defined similarly but with a $\forall$-quantifier in front. It is important to note that, in contrast to Σ^0_1 and Π^0_1 *sets*, not every Σ^0_{n+1} *class* is a Σ^0_1 class relative to $\mathbf{0}^{(n)}$. Indeed, every Σ^0_1 class (relative to whatever oracle) is

topologically open, but for example the Σ^0_2 class

$$\{Z \in 2^\omega \ : \ \exists m_1 \forall m_2 \, (Z\restriction_{m_2} = 0^{m_2})\}$$

is not. The difference is that every Σ^0_1 class relative to $\mathbf{0}^{(n)}$ is of the form

$$\{Z \in 2^\omega \ : \ \exists m_1 \forall m_2 \ldots \phi(Z\restriction_{m_1}, m_2, \ldots, m_n)\},$$

where only the first quantifier is allowed to apply to the length of the initial segment that is considered. (See [16, p. 76], though note that they state the implication the wrong way around.)

A special class of Turing degrees that will appear on several occasions in this thesis are the **PA degrees**, i.e. the degrees that contain a complete extension of Peano Arithmetic (see [48, p.156] or [16, Section 2.2.1]). Being a PA degree is a *highness property*, i.e. a PA degree either computes the halting problem or it is in some sense close to computing the halting problem. A PA degree can compute a member of any nonempty Π^0_1 class. Also, the class of complete extensions of Peano arithmetic is itself a Π^0_1 class.

An **index** for a partial computable function ϕ is a code for an algorithm that computes ϕ, i.e. a number e such that $\phi = \phi_e$. An index for a Σ^0_n/Π^0_n set or class is an index for the computable predicate that defines it. An index for a finite set is a number that encodes the finite string that lists all the elements of the set. Hence from an index for a finite set, we can compute its cardinality, which would not be the case if we would define an index for a finite set to be an index for its characteristic function.

An infinite binary sequence is computable if it is computable as a function $\mathbb{N} \to \{0, 1\}$. Computability for real numbers can be approached in two different ways. Either a real number x is computable if its binary expansion

is computable as a binary sequence. (Note that if x has two different binary expansions, then one only has finitely many zeroes and the other has only finitely many ones, so both are computable.) Other bases than 2 can be used with equivalent result as well. A second equivalent approach is to say that a real number x is computable if for every n we can compute (uniformly in n) a rational number q_n (given by its numerator and denominator, i.e. as a pair of natural numbers) with $|q_n - x| < 2^{-n}$. If we do computations with computable real numbers, than we really do computations with an index of an approximation (q_n). Consequently, the relation $<$ on computable real numbers is c.e., but not computable. Indeed, two approximations can appear to converge to the same real number, but we can never be certain that they won't diverge at a later stage.

2.4 Kolmogorov complexity

Kolmogorov complexity was introduced in the mid-1960s, independently by Ray Solomonoff [56, 57] and Andrey Kolmogorov [26]. That we use the name Kolmogorov complexity, and not Solomonoff complexity, could be due to the fact that Solomonoff only used it as an auxiliary concept in the study of *a priory probability*, whereas Kolmogorov investigated the complexity for its own sake [33, section 1.6].

Kolmogorov complexity formalizes the following idea. It's very easy to give a relatively short description of the string 0^{10^6}, i.e.

$$\underbrace{0000\ldots00}_{10^6 \text{ zeroes}}.$$

It is simply "the string consisting of one million zeroes". On the other hand,

when we toss a fair coin 10^6 times, writing "0" for heads and "1" for tails, then we generate a string of length 10^6 for which we probably don't have any short description. There is probably no regularity in the digits of the string, so we can't give any shorter description than laboriously listing every single digit of the string. This description isn't any shorter than the string itself, so we can say that the string is quite complex.

The Kolmogorov complexity of strings will bring us a first taste of *randomness*. Strings with a pattern in their digits tend to have a low complexity (i.e. significantly less than the length of the string). On the other hand, strings with a high complexity (i.e. close to the length of the string) will appear random. However, the actual values of Kolmogorov complexity are somewhat arbitrary, as they depend on a choice of *universal machine*. By picking a suitable universal machine, we can give a fixed string any complexity we want. Therefore it doesn't make much sense to talk about randomness of individual strings in this way. However, studying how Kolmogorov complexity behaves in the limit as we go from strings to sequences *will* lead to robust notions of randomness for infinite sequences. Studying the complexity of initial segments of random sequences will also expose differences between different notions of randomness. Finally, the simple formalization of Kolmogorov complexity into axiomatic systems such as Peano Arithmetic, will make it an indispensable tool to investigate interactions between randomness and proof theory in Chapter 5. As Kolmogorov complexity is an essential concept throughout this thesis, I will state and proof the basic results in this section in considerable detail.

Plain complexity

Descriptions in English, such as "the string consisting of one million zeroes" in the above example, can be ambiguous and can give rise to paradoxes such as the Berry paradox ("The smallest positive integer not definable in under eleven words."). For this reason, we use binary strings as descriptions, and we fix a machine M to decode these descriptions. That is: a string τ describes the output $M(\tau)$, if this computation halts. The M-complexity of a string σ is the length of the shortest description τ that produces σ when given to M.

Definition.
*The M-**complexity** of a string σ is defined as*

$$C_M(\sigma) = \min\{|\tau| \ : \ M(\tau) = \sigma\},$$

where we let the minimum be ∞ if the set is empty.

Of course, different machines decode descriptions differently, and therefore they have different complexity functions. However, there are certain *optimal* or *universal* machines, whose complexities are lower than the complexities of than any other machine, up to an additive constant.

Definition.
*A machine $\mathbb{V}$ is **universal** if for every machine M there is a constant $c_M \in \mathbb{N}$ such that*

$$C_{\mathbb{V}}(\sigma) \leq C_M(\sigma) + c_M$$

for every $\sigma \in 2^{<\omega}$.

A universal machine can be constructed as follows: let

$$\mathbb{V}(\underbrace{00\ldots0}_{e \text{ zeroes}}1\tau) = \phi_e(\tau)$$

and $\mathbb{V}$ is undefined on any inputs that don't contain any ones. If a string σ has a ϕ_e-description τ of length n, then $\underbrace{00\ldots0}_{e \text{ zeroes}}1\tau$ is a $\mathbb{V}$-description of σ of length $n+e+1$. Therefore $\mathbb{V}$ is universal with $c_{\phi_e} = e+1$. The constant c_M provides room to include the program of M into the $\mathbb{V}$-descriptions. Because of this, c_M is called the **coding constant** of M.

Definition.

*Fix a universal machine $\mathbb{V}$. The **plain complexity** $C(\sigma)$ of a string σ is $C_{\mathbb{V}}(\sigma)$.*

Immediately from the definition of universal machine, it follows that the difference between the complexities for two different universal machines is bounded. For most of our results about Kolmogorov complexity, it will not matter which universal machine is used. If it does matter (particularly in Chapter 5), we will explicitly investigate the issue.

We can approximate C by the time-bounded complexity C^s, for which descriptions must be decoded in at most s steps in order to be considered.

Definition.

*Let $s \geq 0$. The **time-bounded complexity** C^s is defined as*

$$C^s(\sigma) = \min\{|\tau| \ : \ \mathbb{V}(\tau)[s] = \sigma\},$$

where again the minimum is equal to ∞ if the set is empty.

The functions C^s are uniformly computable upper bounds for C. More-

over, for every string σ, $(C^s(\sigma))_{s\in\mathbb{N}}$ is a decreasing sequence in $\mathbb{N}\cup\{\infty\}$ that eventually assumes the constant value $C(\sigma)$. We say that C is **computably approximated from above** by the functions C^s.

The complexity function C satisfies the **counting condition**

$$\#\{x\,:\,C(x)<k\}<2^k \tag{3}$$

for every $k\in\mathbb{N}$. This is immediate from the fact that there are only 2^k-1 strings of length less than k, so there exist only 2^k-1 possible descriptions of length less than k.

In fact, $C(\sigma)\leq D(\sigma)+O(1)$ for every function D that is computably approximable from above and satisfies the counting condition (3). This can be used to give a machine-independent definition of C, up to an additive constant [48, 2.1.16]. (The inequality above uses the *big O notation*. Essentially $O(1)$ can be read as "an additive constant". See for example [16, p.3–4] for more explanations.)

The counting condition (3) implies that for every natural number n, there is at least one string of length n with complexity at least n. Such a string is called **incompressible**. The complexity of a string cannot get significantly bigger than its own length. Indeed, for the machine Id that implements the identity function, every string is a description of itself. Hence

$$C_{Id}(\sigma)=|\sigma|$$

and

$$C(\sigma)\leq|\sigma|+O(1) \tag{4}$$

for all strings σ.

We can also define the complexity of a natural number to be the complexity of its binary form. Then (4) becomes

$$C(n) \leq \log(n) + O(1).$$

Complexity of initial segments of a sequence

We now investigate the complexity of initial segments of a sequence. For a computable sequence Z, we expect that the initial segments have a relatively low complexity. Indeed, if Z is computable, then there is a machine that maps every natural number n to the initial segments $Z\restriction_n$ of length n. Therefore

$$C(Z\restriction_n) \leq C(n) + O(1) \leq \log(n) + O(1). \tag{5}$$

We would like to apply the notion of complexity for strings to define randomness of infinite sequences. A definition for randomness might be that a sequence is random if all initial segments have high complexity, i.e. a complexity close to their length. However, it turns out that all sequences have **complexity dips** for C:

Theorem 2.4.1 (Martin-Löf, see also [16, 3.1.4] and [48, 2.2.1])**.**
For every sequence Z, the difference

$$n - C(Z\restriction_n)$$

is unbounded.

For the proof of this theorem, we consider a computable bijection between strings and natural numbers. We then say that any string *encodes* the corresponding number. For example, we could let σ encode $n-1$ where n is the number with binary form 1σ.

Proof. Consider the machine M that maps any string σ to the binary encoding of $|\sigma|$ concatenated with σ itself. For any $i \in \mathbb{N}$, let m_i be the number with binary encoding $Z\restriction_i$. Then

$$M(Z\restriction_{[i,m_i+i)}) = Z\restriction_{m_i+i},$$

where $Z\restriction_{[i,m_i+i)}$ is the substring of Z with length m_i and starting at position i. So

$$C(Z\restriction_{m_i+i}) \leq m_i + O(1),$$

and

$$m_i + i - C(Z\restriction_{m_i+i}) \geq i - O(1).$$

Therefore, $n - C(Z\restriction_n)$ is unbounded, as required. □

There are still sequences Z for which

$$\liminf (n - C(Z\restriction_n)) < \infty,$$

that is, there exists a $c \in \mathbb{N}$ such that

$$C(Z\restriction_n) \geq n - c$$

for infinitely many n. In Chapter 3, these sequences will be called the 2-random sequences. However, in defining randomness, it will be more fruitful to consider a slightly different notion of complexity, for which most sequences don't have these *complexity dips*: prefix-free complexity.

Prefix-free complexity

Prefix-free complexity arises when we only allow a limited class of machines to decode descriptions: we only allow machines with a prefix-free domain.

That is: if M is a **prefix-free machine** and σ is an initial segment of τ, then we cannot have both $M(\sigma)\downarrow$ and $M(\tau)\downarrow$.

We can use an effective enumeration of all Turing machines to get an effective enumeration of all prefix-free machines. Any program is executed as usual, but when on input σ the instruction to halt comes, the prefix-free machine only actually halts if the program hasn't already halted at an earlier stage on any initial segment of σ or on any string that extends σ. (We use the convention that computations with an input of length s never halt before stage s, so this procedure is computable.)

This enumeration gives us the ability to build a universal prefix-free machine.

Definition.

*A prefix-free machine $\mathbb{U}$ is **universal** (for prefix-free machines) if for every prefix-free machine M there is a constant c_M such that*

$$C_{\mathbb{U}}(\sigma) \leq C_M(\sigma) + c_M$$

for every string $\sigma \in 2^{<\omega}$.

We can construct a universal prefix-free machine $\mathbb{U}$ in an identical way to our construction of $\mathbb{V}$: let

$$\mathbb{U}(\underbrace{00\ldots0}_{e \text{ zeroes}}1\tau) = M_e(\tau)$$

where M_e is the e'th prefix-free machine. Then it is immediate that $\mathbb{U}$ is prefix-free itself, and universal as well.

For prefix-free complexity, the letter K is usually used instead of C. In particular, we define prefix-free complexity as follows:

Definition.

Fix a universal prefix-free machine $\mathbb{U}$*. The* ***prefix-free complexity*** $K(\sigma)$ *of a string* σ *is* $C_{\mathbb{U}}(\sigma)$*.*

Again, for most of our results, it will not matter which universal machine is used. If it does matter, we will explicitly investigate the issue.

Just like plain complexity, prefix-free complexity is computably approximable from above by the time-bounded complexity functions K^s.

The counting condition for C is replaced by the **weight condition** for K:

$$\sum_{\sigma \in 2^{<\omega}} 2^{-K(\sigma)} \leq 1. \tag{6}$$

This follows immediately from the fact that the open cylinders $[\![\tau]\!]$, for τ in the domain of a prefix-free function, are disjoint, so their measures in Cantor space cannot add up to more than 1.

In fact, $K(\sigma) \leq D(\sigma) + O(1)$ for every function D that is computably approximable from above and satisfies the weight condition (6). This can be used to give a machine-independent definition of K, up to an additive constant [48, 2.2.19].

The identity function is not prefix-free, so the inequality

$$C(\sigma) \leq |\sigma| + O(1)$$

does *not* hold for K. Strings of different lengths can be prefixes of each other, but their descriptions for prefix-free complexity are not allowed to be prefixes of each other. So we can try to describe all strings by prepending the digits of any string with a prefix-free description of its length. Indeed, the machine M that on input σ tries to decompose $\sigma = \rho\tau$, with $\mathbb{U}(\rho)\downarrow = |\tau|$,

and if successful outputs τ, is prefix-free and decodes these descriptions. So we have

$$K(\sigma) \leq K(|\sigma|) + |\sigma| + O(1). \tag{7}$$

Now, if the number n has binary form $\alpha_0\alpha_1 \ldots \alpha_{n-1}\alpha_n$, then

$$0\alpha_0 0\alpha_1 \ldots 0\alpha_{n-1}1\alpha_n \tag{8}$$

is a prefix-free description of it (in the sense that these descriptions for all natural numbers can be decoded by a prefix-free machine). Since a number n has a binary form of length approximately $\log n$, we have

$$K(n) \leq 2\log(n) + O(1). \tag{9}$$

Putting (7) and (9) together, we get

$$K(\sigma) \leq |\sigma| + 2\log(|\sigma|) + O(1).$$

By putting more digits of the binary form in between the zeroes in (8), we can even obtain that for every $\epsilon > 0$

$$K(n) \leq (1+\epsilon)\log(n) + O(1).$$

But we cannot replace the constant by 1, as

$$\sum_{n=0}^{\infty} 2^{-\log(n)} = \sum_{n=0}^{\infty} \frac{1}{n} = \infty,$$

so this would violate the weight condition (6).

Weak truth table completeness of Kolmogorov complexity

Given a string σ, at any stage s, we don't know if the approximation $C^s(\sigma)$ is correct, or if some shorter description of σ will appear at a later stage. It seems that we need the halting problem to compute the complexity function. This is indeed the case. Kolmogorov complexity, in both its plain and its prefix-free forms, is Turing complete, and even weak truth table complete.

Theorem 2.4.2 (See also [48, 2.1.28])**.**
The functions C and K are wtt-complete.

Proof. We give a proof for C. The proof for K is a straight-forward adaptation of it.

Let $n \in \mathbb{N}$ be given. Let σ_n be the lexicographically first string of length n with $C(\sigma_n) \geq n$. Let s_n be the first stage at which $C^{s_n}(\sigma) = C(\sigma)$ for all strings σ of length n. Given n and any $s \geq s_n$, we can compute σ_n as the lexicographically first string σ of length n with $C^s(\sigma) \geq n$.

Note that we can concatenate prefix-free descriptions of n and plain descriptions of s to get plain descriptions of the pairs $\langle n, s \rangle$. So

$$C(\sigma_n) \leq K(n) + C(s) + O(1),$$

and hence

$$\begin{aligned} C(s) &\geq C(\sigma_n) - K(n) - O(1) \\ &\geq n - O(\log n) \end{aligned} \tag{10}$$

for all $s \geq s_n$.

If a program halts, then the program itself is a description of the number of steps it takes to halt. So if a program of length m halts in exactly s steps, then

$$C(s) \leq m + O(1). \tag{11}$$

For n large enough, (10) and (11) give $s \leq s_n$. How large n needs to be can be computed from m. Moreover, the value of s_n can be obtained with bounded use of the oracle for C. Subsequently, we know that a program of length m halts in at most s_n steps, or doesn't halt at all. Hence the halting problem is wtt-reducible to C, as required. □

Even stronger than Theorem 2.4.2, Kummer [30] proved that the set of *incompressible strings*

$$\{\sigma \,:\, C(\sigma) \geq |\sigma|\}$$

is truth table complete (tt-complete).

Conditional Kolmogorov complexity

Intuitively, the conditional complexity of σ given τ is the length of the shortest description of σ, where we can use the value of τ *for free* as auxiliary information when decoding the description.

More formally, we have the following definitions, in which we consider machines that take a pair of strings as input.

Definition.

For any machine M we define the M-complexity of σ given τ as

$$C_M(\sigma|\tau) = \min\{|\rho| \,:\, M(\rho, \tau) = \sigma\}$$

where we let the minimum be ∞ if the set is empty.

*A machine $\mathbb{V}^2$ is **universal** if for every machine M there is a constant $c_M \in \mathbb{N}$ such that*

$$C_{\mathbb{V}^2}(\sigma|\tau) \leq C_M(\sigma|\tau) + c_M$$

for all strings σ, τ.

A universal machine $\mathbb{V}^2$ can be constructed in a similar way to the universal machines for (unconditional) Kolmogorov complexity that we constructed before.

Definition.
*Fix a universal machine $\mathbb{V}^2$. The **plain conditional complexity** $C(\sigma|\tau)$ of a string σ given τ is $C_{\mathbb{V}^2}(\sigma|\tau)$.*

Like before, there is a also a prefix-free version of conditional complexity, where we only consider machines N such that

$$M(\cdot, \tau) : \sigma \mapsto M(\sigma, \tau)$$

is a function with a prefix-free domain for every string τ. The **prefix-free conditional complexity** of σ given τ is written as $K(\sigma|\tau)$.

Chapter 3

Notions of randomness

In this chapter we will look at the different possible answers to the question that we posed in the introduction: how to define randomness for individual infinite binary sequences. In Section 3.1 we look at the earliest attempts at a solution, initiated by Von Mises around 1919. His notion is nowadays called *stochasticity*. In Section 3.2, we encounter the first modern randomness notion, namely *Martin-Löf randomness*. Variations on the idea of a Martin-Löf test also lead to other notions such as *Schnorr randomness* and 2-*randomness*. In Section 3.3 we will see that an equivalent definition of Martin-Löf randomness can be obtained by considering the Kolmogorov complexity of initial segments of a sequence. In Section 3.4 we study a third approach to defining randomness, involving *betting strategies* and *martingales*. Again, Martin-Löf randomness can be defined in this way, but moreover a number of new interesting notions appear, namely *computable randomness* and its variations.

The three different points of view in Sections 3.2–3.4 (*typicality*, *incompressibility* and *unpredictability*) are traditionally called the three *paradigms* for defining randomness. More recently however, some completely new ap-

proaches to randomness have been uncovered. In Section 3.5 we discuss the relations between randomness notions and differentiability of computable real functions. This also leads to some interesting remarks on base-invariance of computable randomness. In Section 3.6 we briefly look at some interactions between Martin-Löf randomness and ergodic theory. Finally, in Section 3.7 we revisit the notion of stochasticity and investigate how it relates to the randomness notions that are defined in this chapter.

There are still a lot of other randomness notions that are not mentioned in this chapter. Moreover, I completely ignore major topics such as defining randomness relative to non-standard measures of Cantor space (see e.g. [51]). A full treatment requires the space of an entire book. Hence I can only refer to the excellent monographs of Nies [48] and Downey and Hirschfeldt [16] for more information.

3.1 Stochasticity

The first attempt at defining randomness for infinite sequences goes back to 1919. Richard von Mises [61], while trying to build rigorous foundations for probability theory, defined the notion of a *Kollektiv*. Nowadays we use use the name *stochastic sequence* instead of Kollektiv. Von Mises tried to characterize random sequences by looking at the law of large numbers: the frequency of zeroes and the frequency of ones must approach $\frac{1}{2}$ in a random sequence. This in itself is not enough: the very regular sequence

$$010101010101\ldots$$

also satisfies the law of large numbers. However, it is easy to select a subsequence of this sequence that does not satisfy the law of large numbers, for

example by taking all digits with an odd position in the sequence. Hence, a sequence is stochastic if every subsequence *that we can select* satisfies the law of large numbers. However, Von Mises could not make this idea of a *selectable* subsequence into a rigorous mathematical notion.

Around 1936, Wald [62] made significant efforts to turn stochasticity into a notion free from contradictions. He explicitly stated the need for a restriction to some countable collection of selection rules, and suggested that they should be "*computable in a finite number of steps*"[1]. In 1940, Alonzo Church [14] made this exact using the recently formalized notion of computable function: "*Thus a* Spielsystem *[selection rule] should be represented mathematically, not as a function, or even as a definition of a function, but as an effective algorithm for the calculation of the values of a function.*" This finally gives us the definitions of stochasticity that we will use below.

Even so, stochasticity has not been accepted as a genuine randomness notion. The most well-known objection is Ville's Theorem, which is stated below as Theorem 3.1.2. With regards to the foundations of probability theory, Kolmogorov's axioms became generally favoured over Von Mises' approach. (For more historical details, see e.g. Van Lambalgen's thesis [32].) Nonetheless, stochasticity is still an interesting notion in itself.

Stochastic sequences

Definition.

A ***selection rule*** *is a function* $s : 2^{<\omega} \to \{\text{YES}, \text{NO}\}$. *Given a string or infinite sequence* x, *the* ***set of selected positions in*** x *is*

$$pos_s(x) = \{i \in \text{dom}(x) \ : \ s(x\restriction_i) = \text{YES}\}.$$

[1] In German: "*in endlich vielen Schritten berechnet*"

If $pos_s(x)$ is finite, say $pos_s(x) = \{i_1, i_2, \ldots, i_n\}$ with $i_1 < i_2 < \ldots < i_n$, then s selects the substring

$$s[x] = x(i_1)x(i_2)\ldots x(i_n)$$

from Z.

If $pos_s(x)$ is infinite, then s selects a subsequence $s[x]$ from x in a similar way, such that

$$s[x] = \lim_{n\to\infty} s[x\restriction_n].$$

We also consider partial selection rules (selection rules that are partial functions). As soon as the selection rule is undefined on some initial segment of a string or sequence x, the set of selected positions in x, and the selected substring or subsequence of x become undefined.

Definition.

For any string σ, let $\mathrm{zeroes}(\sigma)$ *be the number of zeroes in σ, i.e.*

$$\mathrm{zeroes}(\sigma) = |\{i \in \mathrm{dom}(\sigma) \ : \ \sigma(i) = 0\}|.$$

An infinite sequence Z satisfies the ***law of large numbers*** *if*

$$\lim_{n\to\infty} \left(\frac{\mathrm{zeroes}(Z\restriction_n)}{n} \right) = \frac{1}{2}.$$

Definition.

A sequence Z is ***Mises-Wald-Church stochastic*** *if $s[Z]$ satisfies the law of large numbers for all partial computable selection rules s such that $s[Z]$ is defined and infinite.*

A sequence Z is ***Church stochastic*** *if $s[Z]$ satisfies the law of large numbers for all (total) computable selection rules s such that $s[Z]$ is infinite.*

A sequence Z is ***weakly Church stochastic*** *if $s[Z]$ satisfies the law of large numbers for all computable selection rules s such that $s[Y]$ is infinite for all sequences Y.*

The notions of Mises-Wald-Church stochasticity and Church stochasticity are well studied. The notion of weak Church stochasticity is new as far as I'm aware. I will argue that it is an interesting notion, by proving in Section 3.7 that it is implied by Schnorr randomness, whereas Church stochasticity is not.

Remark 3.1.1. *In defining a selection rule, it is necessary to be able to select differently depending on the values of the preceding digits. Indeed, suppose we don't allow this. That is, suppose we work with restricted selection rules that are functions $s : \mathbb{N} \to \{\text{YES}, \text{NO}\}$, and we just select digits in positions i such that $s(i) = \text{YES}$. Let's call a sequence* blindly stochastic *if every subsequence selected by such a restricted selection rule satisfies the law of large numbers. We claim that this approach fails, because it leaves us with some blindly stochastic sequences that follow a glaringly obvious pattern.*

Indeed, let Z be any blindly stochastic sequence, and let $\overline{Z}$ be the sequence obtained from Z by taking two copies of every digit. For example, if

$$Z = 0\ \ 1\ \ 1\ \ 0\ \ 1\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1\ldots$$

then

$$\overline{Z} = 00\ 11\ 11\ 00\ 11\ 11\ 00\ 00\ 11\ 00\ 11\ldots.$$

We claim that $\overline{Z}$ will also be blindly stochastic, in spite of the very obvious

pattern that every digit is repeated.

Suppose for contradiction that $\overline{Z}$ *is not blindly stochastic. Then we have a computable selection rule* $\overline{s} : \mathbb{N} \to \{\text{YES}, \text{NO}\}$ *that selects a subsequence from* $\overline{Z}$ *that does not satisfy the law of large numbers. We need to define a computable selection rule* s *that selects a subsequence from the original sequence* Z *that doesn't satisfy the law of large numbers. Let* $\overline{s}_0$ *be the computable selection rule that selects just the even positions that* $\overline{s}$ *selects. Let* $\overline{s}_1$ *be the computable selection rule that selects just the odd positions that* $\overline{s}$ *selects.*

We claim that at least one of these two selection rules must also select a subsequence of $\overline{Z}$ *that does not satisfy the law of large numbers. Indeed, if either* $\overline{s}_0$ *or* $\overline{s}_1$ *selects only finitely many positions, then the other trivially selects a subsequence that still does not satisfy the law of large numbers. So suppose for contradiction that both* $\overline{s}_0$ *and* $\overline{s}_1$ *select an infinite subsequence from* $\overline{Z}$ *that does satisfy the law of large numbers. Let* $\epsilon > 0$*. Abbreviate* $\sigma_i^n = \overline{s}_i \left[\overline{Z}\restriction_n\right]$ *and take* N *large enough such that*

$$\left|\frac{\text{zeroes}(\sigma_i^n)}{|\sigma_i^n|} - \frac{1}{2}\right| < \epsilon$$

for $i \in \{0, 1\}$ *and for all* $n > N$*. Given* $n > N$*, we then have*

$$\begin{aligned}
&\left|\frac{\text{zeroes}(\overline{s}[\overline{Z}\restriction_n])}{|\overline{s}[\overline{Z}\restriction_n]|} - \frac{1}{2}\right| \\
&= \left|\frac{\text{zeroes}(\sigma_0^n) + \text{zeroes}(\sigma_1^n)}{|\sigma_0^n| + |\sigma_1^n|} - \frac{1}{2}\right| \\
&= \left|\frac{|\sigma_0^n|}{|\sigma_0^n| + |\sigma_1^n|}\left(\frac{\text{zeroes}(\sigma_0^n)}{|\sigma_0^n|} - \frac{1}{2}\right) + \frac{|\sigma_1^n|}{|\sigma_0^n| + |\sigma_1^n|}\left(\frac{\text{zeroes}(\sigma_1^n)}{|\sigma_1^n|} - \frac{1}{2}\right)\right| \\
&< \frac{|\sigma_0^n|}{|\sigma_0^n| + |\sigma_1^n|}\epsilon + \frac{|\sigma_1^n|}{|\sigma_0^n| + |\sigma_1^n|}\epsilon \\
&= \epsilon,
\end{aligned}$$

contradicting that $\overline{s}$ selected a subsequence from $\overline{Z}$ not satisfying the law of large numbers.

Suppose without loss of generality that $\overline{s}_0$ selects a subsequence that does not satisfy the law of large numbers. Define $s'(i) = \overline{s}_0(2i)$. Then s' selects exactly the same subsequence from Z that $\overline{s}_0$ selects from $\overline{Z}$. Hence, Z is not blindly stochastic, in contradiction with our assumptions.

Ville's Theorem

The most important objection against stochasticity being a proper randomness notion, was given in 1939 by Jean Ville. He proved that just looking at the law of large numbers for certain subsequences, always fails to capture certain regularities. This cannot be solved by considering a larger class than just (partial) computable selection rules. Indeed, Ville's result applies to any countable collection of selection rules.

Theorem 3.1.2 (**Ville's Theorem** [60])**.**
Let S be a countable collection of selection rules. There exists a sequence Z such that

- *for every $s \in S$ such that $s[Z]$ is defined and infinite, $s[Z]$ satisfies the law of large numbers, and*
- *the inequality*

$$\frac{\text{zeroes}(Z\restriction_n)}{n} \geq \frac{1}{2}$$

holds for every n.

That is, the sequence Z is stochastic with respect to the given collection of selection rules, but the limit frequency of zeroes is always approached from above. This is not something we expect from a random sequence.

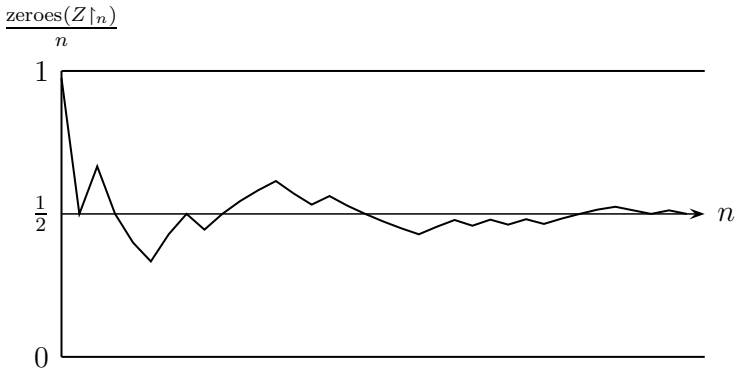

Figure 1: Typical graph of the frequency of zeroes in the initial segments of a random sequence Z. The frequency oscillates around the limit frequency of $\frac{1}{2}$. Compare with Figure 2.

For a proof of Ville's theorem, see [34] or [16, 6.5.1].

By showing that certain regularities don't manifest themselves in a failure of the law of large numbers for certain subsequences, Ville's theorem exposed a fundamental flaw in Von Mises' approach. It would take several decades for a suitable alternative approach to defining randomness to arise. Subsequently, however, soon many different plausible definitions were proposed. Some of these definitions turned out to be equivalent, but others led to different notions. The definitions mostly took one of three different approaches, which are discussed in the following three sections. However, these three paradigms (*typicality*, *incompressibility* and *unpredictability*) are not exhaustive. Ever more new approaches to randomness are being discovered, including links with computable analysis and ergodic theory. I'll also briefly discuss these.

Ville's theorem does not mean that stochasticity is a worthless notion. Stochasticity and its relation to proper randomness notions is still an interesting topic. This is why we will revisit stochasticity in Section 3.7, once we

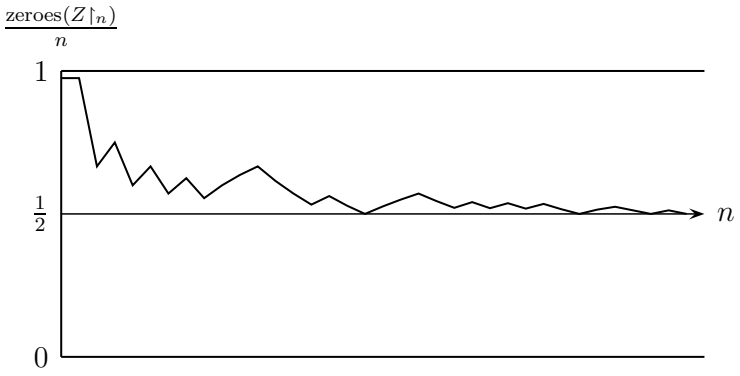

Figure 2: Graph of the frequency of zeroes in the initial segments of a sequence Z as constructed in Ville's Theorem. The frequency is always greater or equal than the eventual limit of $\frac{1}{2}$. Compare with Figure 1.

have defined the relevant randomness notions.

3.2 The typicality paradigm

Ville's theorem showed that stochasticity is not strong enough to be a real randomness notion: there are regularities that cannot be discovered by just considering the convergence of the frequencies of zeroes and ones in a sequence, no matter what countable collection of selection rules we use. In the words of the Swedish mathematician Per Martin-Löf [39]: "*Not even such an intuitively appealing property as the oscillative behavior of the relative frequencies necessarily holds for sequences which are random in [Von Mises'] sense.*"

Martin-Löf himself proposed the first improved approach to defining randomness in 1966. We've seen in Chapter 2 that Cantor Space has a well-studied standard measure, as well as well-studied computability notions for sets of sequences. Martin-Löf combined these to define the notion of an *ef-*

fective null class: a collection of sequences that satisfy a rare (measure zero), effective property and are hence to be considered nonrandom. These effective null classes are stronger than Von Mises' selection rules. In particular, the sequences constructed in Ville's Theorem all lie in an effective null class, and are hence nonrandom in Martin-Löf's sense.

Martin-Löf randomness

Martin-Löf's way to formally define an effective null class was to consider the intersection of an effective sequence of Σ_1^0 classes, with the measure of the intersection converging to 0 at a computable rate. Such a sequence of Σ_1^0 classes is called a Martin-Löf test.

Definition (Martin-Löf [38])**.**
A ***Martin-Löf test*** *is a sequence (U_i) of Σ_1^0 classes (which we call the* ***levels*** *of the test), whose indices can be effectively obtained from i, with $\mu(U_i) < 2^{-i}$. A sequence Z* ***passes*** *the test if $Z \notin \cap_{i\in\mathbb{N}} U_i$. If on the other hand $Z \in \cap_{i\in\mathbb{N}} U_i$, then we say that Z* ***fails*** *the test, or that the test* ***captures*** *Z. A sequence is* ***Martin-Löf random*** *if it passes every Martin-Löf test.*

Remark 3.2.1. *Interestingly, there exists a single* ***universal Martin-Löf test****, such that if a sequence Z passes the universal test, then it passes* every *Martin-Löf test. The universal Martin-Löf test (V_i) is constructed as follows. We can take an effective enumeration of all Martin-Löf tests $(U_i^0), (U_i^1), (U_i^2) \ldots$ and define $V_i = \cup_{j\in\mathbb{N}} U_{j+i+1}^j$. An effective union of Σ_1^0 classes is still a Σ_1^0 class and we have*

$$\mu(V_i) \leq \sum_{j\in\mathbb{N}} \mu\left(U_{j+i+1}^j\right) = \sum_{j\in\mathbb{N}} 2^{-j-i-1} = 2^{-i},$$

so (V_i) is indeed a Martin-Löf test. Furthermore, if $Z \in \cap_{i\in\mathbb{N}} U_i^j$ for some j, then we also have $Z \in \cap_{i\in\mathbb{N}} V_i$. ([38], see also [16, 6.2.5] or [48, 3.2.4].)

Remark 3.2.2. *We can loosen the requirement $\mu(U_i) < 2^{-i}$ to $\mu(U_i) < \frac{1}{h(i)}$ for some fixed computable order $h : \mathbb{N} \to \mathbb{N}$. In other words: the measures $\mu(U_i)$ should converge to 0 at a computable rate. Indeed, if we have $\mu(U_i) < \frac{1}{h(i)}$ for all i, then we can effectively take a subsequence (U_{n_i}) of (U_i) that does satisfy $\mu(U_{n_i}) < 2^{-i}$. Then (U_{n_i}) is a Martin-Löf test in the original sense which captures every sequence captured by (U_i).*

Remark 3.2.3. *Another* test notion *that captures exactly the same class of sequences is the* ***Solovay test****. A Solovay test is a uniformly computable sequence (U_i) of Σ_1^0 classes with $\sum_{i\in\mathbb{N}} \mu(U_i) < \infty$. A sequence Z passes the test if $Z \in U_i$ for at most finitely many values i. A sequence is Martin-Löf random if and only if it passes every Solovay test (attributed by [16] to unpublished work of Solovay; also proven by Shen [55]; see also [16, 6.2.8] or [48, 3.2.19]).*

Martin-Löf randomness is also sometimes called 1**-randomness**. More generally, if we work with Σ_{n+1}^0 classes instead of Σ_1^0 classes in the definition of a Martin-Löf test, then we get the notion of $(n+1)$**-randomness**. Equivalently, we can allow the use of an oracle $\emptyset^{(n)}$ [25, Lemma II.1.5] [16, Corollary 6.8.5]. This equivalence is not trivial, since not every Σ_{n+1}^0 class is also a $\Sigma_1^{0,\mathbf{0}^{(n)}}$ class.

The notion of n-randomness gets stronger as n grows larger.

Schnorr randomness

In the years after Martin-Löf proposed his definition of randomness, German mathematician Claus-Peter Schnorr formulated a number of objections

against the notion. One of his main objections was that Martin-Löf tests are not sufficiently computable. In his 1971 book [53, p.35-36] he wrote: *"Let* (U_i) *be a Martin-Löf test. Given* $\sigma \in 2^{<\omega}$ *and* $i \in \mathbb{N}$*, the value*

$$2^{|\sigma|}\mu([\![U_i]\!] \cap [\![x]\!])$$

expresses that probability that an infinite sequence starting with σ *lies in the effectively open neighbourhood* U_i *of the effectively null class* $\mathcal{U} = \cap_{i\in\mathbb{N}} U_i$*. This value thus indicates to some extent how much the initial segment* σ *conforms with the almost-everywhere property defined by* $2^\omega \setminus \mathcal{U}$*. If the value is high, then* σ *conforms relatively little with the property. In the definition of Martin-Löf test, however, we did not require in any way that this value has to be computable. Indeed, as we will see, this is generally, and in particular for a universal Martin-Löf test, not the case."*[2]

Schnorr got around this objection by requiring that the measures $\mu(U_i)$ of the levels of a test (U_i) are computable real numbers, uniformly in i. This gives rise to the notion of Schnorr randomness.

Definition.

A ***Schnorr test*** *is a uniformly computable sequence* (U_i) *of* Σ^0_1 *classes such that* $\mu(U_i)$ *is a computable real number uniformly in* i *with* $\mu(U_i) <$

[2]Translated from German using a more modern notation. The original quote is as follows: *"Sei* $Y \subset N \times X^\star$ *ein rekursiver Sequentialtest. Zu* $x \in X^\star$ *und* $i \in N$ *bedeutet der Wert*

$$2^{|x|}\overline{\mu}([Y_i] \cap [x])$$

die Wahrscheinlichkeit dafür, daß eine unendliche Folge, die mit x *beginnt, in der r.o. Ungebung* $[Y_i]$ *der rekursiven Nullmenge* $\mathfrak{Y}_Y = \cap_{i\in N}[Y_i]$ *liegt. Dieser Wert sagt also etwas darüber aus, inwieweit die Anfangsfolge* x *den zu* $\mathfrak{Y}_Y$ *zugehörige FÜG [Fastüberallgesetz] entspricht. Ist der Wert hoch, dann entspricht* x *eben diesem FÜG in geringem Maße. Nun haben wir aber bei der Definition rekursiver Sequentialtests keinerwegs gefordert, daß man diese Werte effektiv berechnen kann. Tatsächlich ist dies, wie wir noch sehen werden, im allgemeinen und insbesonders für einen universellen rekursiven Sequentialtest auch nicht der Fall."*

2^{-i}. *A sequence Z is* ***passes*** *the test if $Z \notin \cap_{i\in\mathbb{N}} U_i$. A sequence is* ***Schnorr random*** *if it passes every Schnorr test.*

Remark 3.2.4. *Just like with Martin-Löf tests, we don't strictly need $\mu(U_i) < 2^{-i}$ in a Schnorr test, as long as $\mu(U_i) \to 0$ at some computably rate.*

Remark 3.2.5. *A useful fact for Schnorr tests, is that a sequence Z is already certainly not Schnorr random when $Z \in U_i$ for infinitely many values of $i \in \mathbb{N}$ and for some Schnorr test (U_i). Indeed, if this is the case, then we can define another Schnorr test (V_i) by*

$$V_i = \bigcup_{j\in\mathbb{N}} U_{i+j+1}$$

that succeeds on Z in the conventional sense that $Z \in \cap_{i\in\mathbb{N}} V_i$. (See [48, 3.5.10], or [16, 7.1.10] for a slightly stronger result.)

Kurtz randomness and weak n-randomness

We can pose even stricter requirements on our tests. A Kurtz test is a Martin-Löf test where the U_i are now uniformly Δ^0_1, instead of just Σ^0_1. Equivalently, every U_i is a finite union of basic open sets $\cup_{\sigma\in D_i} [\![\sigma]\!]$, where the indices of the finite sets D_i are uniformly computable in i.

Definition (Kurtz [31] and Wang [63])**.**
A ***Kurtz test*** *(U_i) is sequence of Δ^0_1 sets, whose indices are uniformly computable in i, with $\mu(U_i) < 2^{-i}$. A sequence Z* ***passes*** *the test if $Z \notin \cap_{i\in\mathbb{N}} U_i$. A sequence that passes every Kurtz test is called* ***Kurtz random****.*

A Kurtz random sequence can also be defined as a sequence that is contained in no null (measure zero) Π^0_1 class.

Kurtz randomness is also called **weak 1-randomness**, which generalizes as follows:

Definition.

*A sequence Z is **weakly n-random** if $Z \notin A$ for every null Π^0_n class.*

It should be noted that weak $(n+1)$-randomness is *not* the same as weak 1-randomness relativized to the oracle $\emptyset^{(n)}$. This is because, as we mentioned in Chapter 2, not every Π^0_{n+1} class is a Π^0_1 class relative to $\mathbf{0}^{(n)}$. (See also [16, p. 76 and p. 286].)

Weak $(n+1)$-randomness implies n-randomness. Indeed, null Π^0_{n+1} classes are exactly the uniform intersections of sequences of Σ^0_n classes with measures converging to 0. So if we take Σ^0_n Martin-Löf tests but don't require any computable bound on how quickly the measures of the sets U_i converge to zero, then we obtain tests for weak $(n+1)$-randomness.

Moreover, justifying the name of weak n-randomness, n-randomness implies weak n-randomness [25, II.5.1].

There are Kurtz random sequences that do not satisfy the law of large numbers. In particular, this applies to any weakly 1-generic sequence [48, 3.5.3–3.5.5]. One can argue that therefore, Kurtz randomness is really too weak to be a genuine randomness notion. However, Kurtz randomness (weak 1-randomness) relates to Martin-Löf randomness (1-randomness) just like weak 2-randomness relates to 2-randomness, and so on. Because the notion fits nicely into the hierarchy of randomness notions, the name Kurtz randomness still applies.

Randomness and Turing completeness

As we will see in the next section, some Martin-Löf random sequences are Turing complete, for example Chaitin's Ω. Even stronger: the Kučera-Gács theorem ([29], [20], see also [16, 8.3.2]) says that there exists a Martin-Löf random above any given Turing degree. In a sense however, these sequences are not typical for Martin-Löf randomness. One might even argue that randomness and strong computational power should not go together: how can a sequence be random, and still be able to compute nontrivial information like the halting problem?

This issue becomes even more pressing when we observe that there is a clear dichotomy between the Martin-Löf random sequences that are Turing complete and those that aren't. The latter aren't even close to computing the halting problem, in the sense that they can't even have a PA degree ([58], see also [48, 4.3.5] or [16, 8.8.4]).

Downey and Hirschfeldt [16, footnote 4 p.229] make a cunning analogy between passing a Martin-Löf test and passing an *ignorance test*. You can pass an ignorance test either by being genuinely ignorant, or by being so smart that you can successfully impersonate an ignorant person. In a similar way, Martin-Löf tests are passed by computationally weak as well as computationally strong sequences.

When we move to stronger notions such as weak 2-randomness and 2-randomness, then no Turing-complete sequence is random anymore. We can even define a test notion that gives us a randomness definition which includes exactly the Martin-Löf randoms that are not Turing complete ([19], see also [16, 7.7.4]). This notion is called **difference randomness**, as a test for difference randomness is the *difference* of two Martin-Löf tests, in the sense that every level of the difference test is the set-theoretic difference of

the corresponding levels of the respective Martin-Löf tests. Difference randomness is strictly stronger than Martin-Löf randomness but strictly weaker than weak 2-randomness.

So is difference randomness, or 2-randomness, in some sense a *better* notion than Martin-Löf randomness? If so, are still stronger notions even more preferable? Even Per Martin-Löf himself, just a few years after defining the notion of Martin-Löf randomness, proposed to define randomness instead using the much stronger hyperarithmetical null classes, rather than Martin-Löf tests [39].

However, strength is not everything. Martin-Löf randomness has very useful properties and applications, and many alternative characterizations, that cannot be found for stronger notions of randomness. In short, Martin-Löf randomness interacts better with other areas of computability theory (and even with proof theory, as we will see in chapter 5) than most other randomness notions. A possible explanation is that the Martin-Löf test has a good balance between capturing power and simplicity. Consequently, knowing that a particular sequence is not random (i.e. that it fails some Martin-Löf test) is very useful information. Stronger notions have more complicated tests, which makes it harder to find interesting consequences of the fact that a given sequence is not random. Can we call a sequence regular (i.e. non-random) if the pattern that it satisfies is so complex that we can't do anything with it?

Whatever your personal view, it is unlikely that there will ever be a single notion with the status of *only sensible randomness notion*. To our current knowledge, Martin-Löf randomness is probably the most robust and well-behaved of all notions. However, many other notions have their own appeal

and interest. Since randomness has so many different *faces*, studying the differences between the many notions is essential in the quest to understand the concept of randomness in general. This is one of the main objectives of this thesis.

3.3 The incompressibility paradigm

Kolmogorov complexity provides a second approach to defining randomness. We expect that we cannot give a much shorter description for the first n digits of a random sequence, than just listing these digits one by one, giving a description of approximate length n. Hence randomness corresponds to incompressibility of initial segments. We can only require this up to an additive constant, if the notion is to be independent of the choice of universal machine, and also to allow for a finite number of initial segments of a random sequence to have very low complexity, as long as the sequence as a whole is random. We also need the correct notion of Kolmogorov complexity. As we saw in Theorem 2.4.1, there are no sequences Z for which there exists a constant c such that

$$C(Z\upharpoonright_n) > n - c \tag{12}$$

for all $n \in \mathbb{N}$. It is still possible to change equation (12) to obtain definitions of randomness using plain complexity, for example by replacing the expression on the right hand side with a function that grows more slowly in n. However, the most elegant definition of randomness using Kolmogorov complexity is obtained by using prefix-free complexity.

Theorem 3.3.1 (Schnorr [54], see also [48, 3.2.9] and [16, 6.2.3])**.** *A sequence Z is Martin-Löf random if and only if there exists a constant c such that*

$$K(Z\restriction_n) > n - c \tag{13}$$

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

Proof. First suppose that Z is not Martin-Löf random. Hence there exists a Martin-Löf test (U_i) such that $Z \in \cap_{i\in\mathbb{N}} U_i$. Suppose every U_i is generated by the prefix-free c.e. set X_i. Construct a prefix-free machine M as follows: for every string σ enumerated in some X_{2i}, provide an M-description of length $|\sigma| - i + 1$. As the measure of U_{2i} is at most 2^{-2i}, the M-descriptions for the elements of X_{2i} contribute at most weight 2^{-i+1} to the domain of M. So the descriptions that we want M to have, do not a priory violate the weight condition (6). The actual existence of such a machine M is guaranteed by a result known as the Kraft-Chaitin Theorem, or KC-theorem, or Machine Existence Theorem ([11], [48, 2.2.17], [16, 3.6.1]). If the machine M has coding constant d, and $Z\restriction_{n_i}$ is the initial segment of Z that is enumerated in X_{2i}, then

$$K(Z\restriction_{n_i}) \leq K_M(Z\restriction_{n_i}) + d \leq n_i - i + 1 + d$$

for all $i \in \mathbb{N}$. Hence (13) cannot hold for any constant c.

For the other direction, suppose that for every $c \in \mathbb{N}$ we can find some initial segment x_c of Z such that $K(x_c) \leq |x_c| - c$. Given i, let X_i be the prefix-free set of minimal (for the order $\prec$) strings σ with complexity $K(\sigma) \leq |\sigma| - i$. These sets X_i are uniformly c.e.. Furthermore, $[\![X_i]\!]$ has measure at most 2^{-i}, otherwise the weight condition (6) would be violated. So $([\![X_i]\!])$ is a Martin-Löf test, and it succeeds on Z. Therefore Z is not Martin-Löf random, as required. □

Another way of putting Theorem 3.3.1 is to say that a sequence Z is Martin-Löf random if and only if

$$\liminf_{n\to\infty}(K(Z\restriction_n) - n) > -\infty,$$

i.e. the lengths of initial segments do not grow more quickly than their complexities. But in fact, if Z is Martin-Löf random, something stronger holds:

$$\lim_{n\to\infty}(K(Z\restriction_n) - n) = \infty.$$

Hence, the prefix-free complexities of initial segments never grow at the same rate as their lengths. Either the complexities grow strictly faster, in which case the sequence is Martin-Löf random, or the lengths grow strictly faster, in which case the sequence is not random. This was proven in 1987 by Chaitin [12] as an application of Solovay tests (see also [48, 3.2.21]). We give a direct proof of a more recent stronger result, called the *Ample Excess Lemma*.

Theorem 3.3.2 (**Ample Excess Lemma**. Miller and Yu [45], see also [16, 6.6.1])**.**
A sequence Z is Martin-Löf random if and only if

$$\sum_{n\in\mathbb{N}} 2^{n-K(Z\restriction_n)} < \infty.$$

Proof. First, suppose that Z is not Martin-Löf random. By Theorem 3.3.1, the difference $(n - K(Z\restriction_n))$ has no upper bound. Therefore

$$\sum_{n\in\mathbb{N}} 2^{n-K(Z\restriction_n)} = \infty.$$

For the other direction, define

$$U_i = \left\{ Y \in 2^\omega \ : \ \sum_{n \in \mathbb{N}} 2^{n - K(Y\upharpoonright_n)} > 2^i \right\}.$$

By approximating $K(Y\upharpoonright_n)$ from above, we can approximate the terms $2^{n-K(Y\upharpoonright_n)}$ from below. Moreover, all terms are positive, so we can approximate the whole sum $\sum_{n\in\mathbb{N}} 2^{n-K(Y\upharpoonright_n)}$ from below. Therefore the U_i's are Σ^0_1 classes uniformly in i. We claim that $\mu(U_i) \leq 2^{-i}$, i.e. that (U_i) is a Martin-Löf test.

Suppose for contradiction that $\mu(U_i) > 2^{-i}$. Then also for some $m \in \mathbb{N}$ the measure of

$$\left\{ Y \in 2^\omega \ : \ \sum_{n=0}^{m} 2^{n - K(Y\upharpoonright_n)} > 2^i \right\}$$

must be greater than 2^{-i}. (Indeed, U_i is the countable intersection of these sets, so its measure is the limit of the measures of these sets.) However, for any $m \in \mathbb{N}$ we have

$$\begin{aligned} \sum_{\sigma \in 2^m} \sum_{n=0}^{m} 2^{n - K(\sigma\upharpoonright_n)} &= \sum_{\tau \in 2^{\leq m}} 2^{m - |\tau| 2^{|\tau| - K(\tau)}} \\ &= 2^m \sum_{\tau \in 2^{\leq m}} 2^{-K(\tau)} \\ &\leq 2^m, \end{aligned}$$

where the last line is due to the weight condition (6). Therefore, less than a fraction 2^{-i} of strings σ of length m can satisfy

$$\sum_{n=0}^{m} 2^{n - K(\sigma\upharpoonright_n)} > 2^i.$$

This is a contradiction. So we have proven that $\mu(U_i) \leq 2^{-i}$.

As (U_i) is a Martin-Löf test that captures every sequence Z which does not satisfy

$$\sum_{n\in\mathbb{N}} 2^{n-K(Z\restriction_n)} < \infty,$$

no such sequence is Martin-Löf random, as required. □

Corollary 3.3.3.

A sequence Z is Martin-Löf random if and only if

$$\lim_{n\to\infty} (K(Z\restriction_n) - n) = \infty.$$

Proof. One direction is weaker than Theorem 3.3.1. For the other direction, suppose that Z is Martin-Löf random. By the Ample Excess Lemma (Theorem 3.3.2),

$$\sum_{n\in\mathbb{N}} 2^{n-K(Z\restriction_n)} < \infty.$$

This can only happen if

$$\lim_{n\to\infty} (n - K(Z\restriction_n)) = -\infty,$$

as required. □

Chaitin's Ω

For every prefix-free machine M we can consider the measure Ω_M of the Σ^0_1 class generated by its domain:

$$\Omega_M = \sum_{\sigma\in \mathrm{dom}\, M} 2^{-|\sigma|}.$$

This is called the **halting probability** of the machine, since if we take an infinite sequence Z at random, Ω_M is exactly the probability that M halts on some initial segment of Z.

If M is a universal prefix-free machine, then we denote the halting probability simply by Ω. Of course, the exact value of Ω depends on the choice of universal machine, but this will not matter for the properties of Ω that are relevant to us. The real number Ω is also called **Chaitin's constant**, after Gregory Chatin's 1975 paper [11].

Ω is a left-c.e. real number, which means that there is a computable nondecreasing sequence of rational numbers that converges to Ω_M. Indeed, the approximations $\Omega[s] = \Omega_{\mathbb{U}[s]}$ provide exactly such a sequence. However, Ω is far from being computable.

Theorem 3.3.4 (Calude and Nies [8], see also [16, 6.1.2] or [48, 3.2.30])**.** *Ω is wtt-complete.*

Proof. To compute Ω to an accuracy of 2^{-n} it suffices to ask to oracle for the halting problem if the approximations $\Omega[s]$ ever become greater than $k2^{-n}$ for all $k \in \{0, 1, \ldots, 2^n\}$. Hence $\Omega \leq_{wtt} \mathbf{0}'$.

Conversely, to prove $\mathbf{0}' \leq_{wtt} \Omega$, consider the prefix-free machine M that halts on 0^n1 with output s if $\phi_n(n)$ halts at stage s. Let c be the coding constant for M. If $\phi_n(n)$ halts at stage s, then

$$K(s) \leq K_M(s) + c \leq n + 1 + c,$$

so descriptions of s contribute at least 2^{-n-1-c} to the universal halting probability Ω. With bounded use of the oracle for Ω we can find all such stages s, and check if $\phi_n(n)$ halts at any of them. If not, then it must be that $\phi_n(n)\uparrow$. □

Before proving that Ω is Martin-Löf random, we need the following lemma.

Lemma 3.3.5.
For every partial computable function f there exists a constant c such that if $f(\sigma)$ is defined, then

$$K(f(\sigma)) \leq K(\sigma) + c.$$

Proof. Consider the prefix-free machine M that on input σ computes $f(\mathbb{U}(\sigma))$, if this is defined. Let c be the coding constant of M. Suppose that $f(\sigma)$ is defined. Every $\mathbb{U}$-description of σ is an M-description of $f(\sigma)$, so we have

$$K(f(\sigma)) \leq K_M(f(\sigma)) + c \leq K(\sigma) + c$$

as required. □

Theorem 3.3.6 (Chaitin [11, Theorem 4.3b], see also [16, 6.1.3] or [48, 3.2.11])**.**
Ω is Martin-Löf random.

Proof. We claim that there is a constant c such that $K(\Omega\restriction_n) \geq n - c$ for all n.

Consider the partial function f that on input σ tries to enumerate $\mathrm{dom}(\mathbb{U})$ until its measure is at least $0.\sigma$ at some stage s, and if successful outputs some string $f(\sigma)$ with $K^s(f(\sigma)) > |\sigma|$. Note that $f(\Omega\restriction_n)$ is defined and not only $K^s(f(\Omega\restriction_n)) > n$, but also $K(f(\Omega\restriction_n)) > n$, since any $\mathbb{U}$-description of length at most n must appear in $\mathrm{dom}(\mathbb{U})$ before $\mu(\mathrm{dom}(\mathbb{U}[s])) = \Omega[s]$ reaches $0.\Omega\restriction_n$. By the lemma above there exists a constant c such that

$$n < K(f(\Omega\restriction_n)) \leq K(\Omega\restriction_n) + c$$

for every n, as required. □

Corollary 3.3.7.

There exists a Martin-Löf random that is left-c.e. and Turing complete.

As discussed at the end of Section 3.2, being left-c.e. and computing the halting problem are properties that we might not expect any random sequences to have. Indeed, stronger notions such as 2-randomness have no left-c.e. or Turing-complete random sequences. Hence Ω could be regarded as a somewhat atypical random number.

3.4 The unpredictability paradigm

The third way to go about defining randomness of infinite sequences comes from the idea that the digits of a random sequence should be unpredictable. This idea is somewhat similar to Von Mises' definition of stochasticity, but we should allow for more ways to *predict* patterns in a sequence, rather then just selecting a subsequence that doesn't satisfy the law of large numbers. We do this by using *betting strategies*. That name is appropriate since a betting strategy works like a gambler playing roulette in a casino, repeatedly betting on either red or black numbers. He doesn't have to tip the dealer and there is no number zero on the roulette wheel, so he is playing a fair game. The casino had better make certain that the sequence of reds and blacks that appears is random. If there is a regularity in the outcomes, the gambler will be able to come up with a betting strategy that exploits this and makes his profits grow unboundedly.

Betting on the digits of an infinite sequence works just like this. The gambler starts with a certain initial capital, and then has the option to wager a certain fraction of his capital on the first digit of the sequence being a zero, or on it being a one. If he is correct, then he doubles the money that he

has risked. If he is wrong however, he loses the money that he has wagered. Next, he can place a new bet on the second digit of the sequence with his new capital, and so on. If his profits grow unboundedly whilst betting on the sequence, then the gambler has discovered a pattern, and hence it is a non-random sequence.

This type of gambling game, formalized using the concept of *martingales*, was already investigated by Ville in 1939 [60] as a way to provide a definition of randomness where Ville's Theorem does not pose an objection. Unfortunately, it appears that Ville was unaware of the recent developments in computability theory. Therefore, he did not think that there was a canonical class of martingales that could be used to define random sequences. Ville concluded: "*[. . .] the definition of randomness by martingales is relative; it supposes a prior choice of properties (of measure zero) to be excluded. If, in some sense, it solves the question of randomness more completely than the definition of Wald, it does not succeed in providing an arithmetical model of a sequence that has* all *the characteristics of a randomly generated sequence; this last problem is considered by us to be insolvable, and on this point we yield to the opinion of many mathematicians, among whom E. Borel, Fréchet and P. Lévi.*"[3]. It is rather astonishing that Ville was so close to giving the current definition of computable randomness, but gave up and deemed the problem unsolvable.

It would take 30 more years before Claus-Peter Schnorr took up Ville's idea again. Schnorr provided the missing link by suggesting to consider

[3]Original text in French: "*Mais la condition d'irrégularité par la martingale est relative; elle suppose un choix préalable des propriétés (de probabilité nulle) à exclure. Si, dans un certain sens, elle résout la question d'irrégularité plus complètement que la condition de M. Wald, elle ne parvient pas à donner un modèle arithmétique d'une suite présentant* tous *les caractères d'une suite prise au hasard; ce dernier problème est considéré par nous comme insoluble, et nous nous soumettons sur ce point à l'opinion de nombreux mathématiciens, parmi lesquels MM. E. Borel, Fréchet, P. Lévi.*"

just computable betting strategies, and thus defined computable randomness. This provided Schnorr with another argument against the notion of Martin-Löf randomness. Indeed, when defining randomness using betting strategies, computable randomness is a much more natural notion than Martin-Löf randomness. Moreover, Schnorr randomness appears again, when we require that the profits of a betting strategy grow faster than some computable order.

More on the history of martingales can be found in a dedicated issue of the *Journal Electronique d'Histoire des Probabilités et de la Statistique*, in particular in the article by Bienvenu, Shafer and Shen [5].

Martingales and computable randomness

A betting strategy can be represented by its capital function B. This function maps each string σ to the amount of money that the better strategy has after betting on the digits of σ. For example, suppose a gambler starts with one unit of money, i.e. $B(\lambda) = 1$. He might suspect that the first digit of the sequence is a zero, without being willing to risk all his money on this bet. Consequently, his betting strategy might bet $\frac{1}{2}$ on the first digit being a one. If this is correct, then the gambler wins $\frac{1}{2}$, giving him a new capital of $B(0) = \frac{3}{2}$. If the first digit is a one however, he loses $\frac{1}{2}$ and remains with only $B(1) = \frac{1}{2}$.

Next the gambler has to bet on the second digit. If the first digit turned out to be a zero, then he might be convinced that the second digit will be a zero as well. His betting strategy might then risk his entire capital on the next digit being a zero. If this is correct, then he doubles his money and obtains a capital of $B(00) = 3$. If the next digit is a one however,

then $B(01) = 0$, i.e. the gambler has lost all his money and cannot bet any further. If the first digit turned out to be a one, then the gambler might bet differently. For example, he might not have any clue what the next digit is going to be. In this case, he could not bet anything at all on the second digit. Consequently, he keeps the same capital, whatever the value of the second digit, i.e. $B(1) = B(10) = B(11) = \frac{1}{2}$. We also call this *betting evenly*, as the same result can be accomplished by betting an equal amount of money on either outcome.

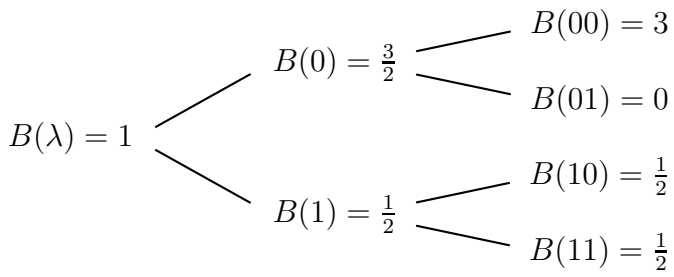

Figure 3: The first few values of the martingale B, which is the capital function of the example betting strategy in the text.

The first few values of the capital function B of this betting strategy are shown in Figure 3. This is a function that satisfies

$$B(\sigma) = \frac{B(\sigma 0) + B(\sigma 1)}{2} \tag{14}$$

for all strings σ. Such a function is called a **martingale**. Equation (14) is a **fairness condition**: the expected value of the new capital must be equal to the initial capital. Any betting strategy, as above, gives rise to a martingale, and every martingale corresponds to a betting strategy. Hence we will use the terms *betting strategy* and *martingale* interchangeably.

Definition.

*A **martingale** is a function*

$$B : 2^{<\omega} \to \mathbb{R}_{\geq 0}$$

that satisfies

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

*for all strings σ. The martingale B **succeeds** on a sequence Z if*

$$\limsup_{n\to\infty} B(Z\restriction_n) = \infty. \tag{15}$$

*A sequence Z is **computably random** if* no computable *martingale succeeds on Z.*

The terminology can be slightly confusing here. *Passing* a Martin-Löf test is an indication of randomness, whereas *success* for a martingale is a proof of non-randomness.

Remark 3.4.1. *The fairness condition (14) serves to stop the gambler from obtaining more money than he should deserve from his bets. However, there is no cheating in throwing some money away (tips to the dealer, donations to charity, some good Belgian beers; whatever you like). Therefore we can loosen the requirement to an inequality:*

$$B(\sigma) \geq \frac{B(\sigma 0) + B(\sigma 1)}{2}. \tag{16}$$

*Any function $B : 2^{<\omega} \to \mathbb{R}_{\geq 0}$ satisfying this inequality is called a **supermartingale**. For every (computable) supermartingale there exists a (computable) martingale that succeeds on the same (and possibly more) sequences,*

simply by saving any money that the supermartingale throws away. Therefore it does not matter whether we define computable randomness using martingales or supermartingales.

Remark 3.4.2. *We formalized* infinite profits *by requiring that the lim sup of the capital is infinity. One might call this jokingly the* American concept for success*: it does not matter if you lose almost all of your money repeatedly, because in the* land of opportunity *you'll always have the possibility to grow rich again. A more* European condition for success *would be to require a more steady growth of capital, without repeated bankruptcies, i.e. the* limit *(and not just the lim sup) of the capital should be infinity. So why didn't we use a limit to formalize* infinite profits*? In fact it does not matter. If a (computable) martingale succeeds on a sequence Z in the American sense, then we can construct another (computable) martingale B' that succeeds on Z in the European sense. Even stronger, we can require that the martingale B' satisfies the so-called* ***savings property****:*

$$B'(\tau) \geq B'(\sigma) - 2 \tag{17}$$

for all strings σ, τ where τ extends σ. This result is called the savings lemma *or the* savings trick *(see e.g. [7, Lemma 2.3], [16, 6.3.8] or [48, 7.1.14]). We construct B' by splitting the capital in a* wallet*, which initially contains the whole initial capital, and a* savings account*. Whenever there are more than 2 units of money in the wallet, we leave just 1 unit in the wallet, and move the rest to the savings account. B' bets on zeroes and ones in the same proportions as B, but using only the money in the wallet and never touching the money in the savings account. Hence, the profits of B' will grow more slowly compared to those of B. But even if B loses a dramatic amount of*

money in some series of bets, B' loses at most what is in the wallet, so at most 2. This is illustrated in Figure 4.

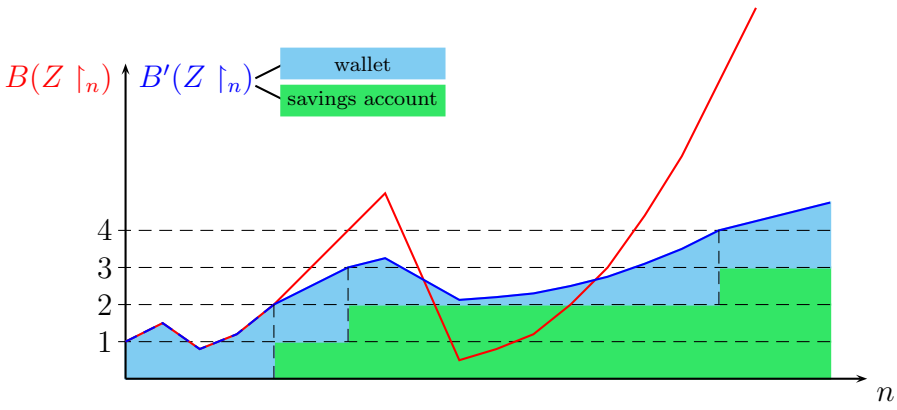

Figure 4: The savings lemma.

Remark 3.4.3. *We can restrict ourselves to (super)martingales with values in the rational numbers (or even dyadic rational numbers), where we represent a rational number as a pair of natural numbers (numerator and denominator). Indeed we can effectively approximate each computable (super)martingale $B : 2^{<\omega} \to \mathbb{R}_{\geq 0}$ with a computable (super)martingale $D : 2^{<\omega} \to \mathbb{Q}_{\geq 0}$ such that whenever B succeeds on Z, then also D succeeds on Z. The important advantage of this is that equality of rational numbers is a computable relation. ([53, 9.3]; see also [16, 7.1.2] or [48, 7.3.8])*

Remark 3.4.4. *It is essential that a betting strategy is allowed to read the values of previous digits in a sequence, before having to decide on how to bet on the next digit. Indeed, suppose that we don't allow this, and restrict out attention to betting strategies $b : \mathbb{N} \to [-1, 1]$ that bet a fraction $b(n)$ of their current capital on the value of the n'th digit being a zero. So if $b(n) = -1$, then the betting strategy puts all its money on the n'th digit being a one. If $b(n) = 0$, then the betting strategy bets evenly on the n'th digit.*

Generally, if the n'th digit of the sequence is a zero, then the capital of the betting strategy gets multiplied by a factor $(1 + b(n))$*. If the n'th digit of the sequence is a one, then the capital of the betting strategy gets multiplied by a factor* $(1 - b(n))$*. Let's call a sequence* blindly computably random *if no such restricted computable betting strategy* $b : \mathbb{N} \to [-1, 1]$ *succeeds on it. This does not give us a suitable randomness notion, for the following reason.*

Let Z *be any blindly computably random sequence, and let* $\overline{Z}$ *be the sequence obtained from* Z *by taking two copies of every digit. For example, if*

$$Z = 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \ldots$$

then

$$\overline{Z} = 00\ 11\ 11\ 00\ 11\ 11\ 00\ 00\ 11\ 00\ 11 \ldots.$$

We claim that $\overline{Z}$ *will also be blindly computably random, in spite of the very obvious pattern that every digit is repeated.*

Suppose for contradiction that there is a blind computable betting strategy $\overline{b} : \mathbb{N} \to [-1, 1]$ *that succeeds on* $\overline{Z}$*. Then the sequence of profits*

$$\prod_{i=0}^{n} \left(1 + (-1)^{\overline{Z}(i)} \overline{b}(i)\right)$$

is unbounded as $n \to \infty$*. So at least one of the sequences*

$$\prod_{i=0}^{n} \left(1 + (-1)^{\overline{Z}(2i)} \overline{b}(2i)\right)$$

or

$$\prod_{i=0}^{n} \left(1 + (-1)^{\overline{Z}(2i+1)} \overline{b}(2i+1)\right)$$

must also be unbouded as $n \to \infty$*. So either*

$$b\colon \mathbb{N} \to [-1,1]\colon i \mapsto \overline{b}(2i)$$

or

$$b\colon \mathbb{N} \to [-1,1]\colon i \mapsto \overline{b}(2i+1)$$

defines a blind computable betting strategy that succeeds on the original sequence Z*. Hence* Z *is not blindly computably random, in contradiction with our assumptions.*

It is even possible to construct a single *blind computable betting strategy* b *from* $\overline{b}$*, such that whenever* $\overline{b}$ *succeeds on a sequence* $\overline{Z}$*, then* b *succeeds on the original sequence* Z*. For this, note that the square root of the sequence of profits*

$$\prod_{i=0}^{n} \sqrt{\left(1+(-1)^{Z(i)}\overline{b}(2i)\right)\left(1+(-1)^{Z(i)}\overline{b}(2i+1)\right)}$$

is unbounded as $n \to \infty$*. By the inequality of arithmetic mean and geometric mean, we have*

$$\begin{aligned}
&\sqrt{(1+\overline{b}(2i))(1+\overline{b}(2i+1))} + \sqrt{(1-\overline{b}(2i))(1-\overline{b}(2i+1))} \\
\leq\ & \frac{1+\overline{b}(2i)+1+\overline{b}(2i+1)+1-\overline{b}(2i)+1-\overline{b}(2i+1)}{2} \\
\leq\ & 2
\end{aligned}$$

so we can define a blind computable betting strategy $b : \mathbb{N} \to [-1,1]$ *that when betting on the* i*'th digit of* Z*, multiplies its capital by at least*

$$\sqrt{(1+\overline{b}(2i))(1+\overline{b}(2i+1))}$$

when the digit is a zero, and multiplies its capital by at least

$$\sqrt{(1-\overline{b}(2i))(1-\overline{b}(2i+1))}$$

when the digit is a one. Hence the profits that b makes when betting on the original sequence Z are unbounded, as required.

In conclusion, it is absolutely necessary that betting strategies are allowed to look at the context of the digits that they are betting on, before deciding on how to bet. Otherwise we do not get a suitable randomness notion.

Lemmas about martingales

Before studying the relation of computable randomness with other notions of randomness, we prove a few useful lemmas.

The first lemma generalizes the fairness conditions (14) and (16). It is also related to *Doob's optional stopping theorem* from probability theory. It will be used to prove that a Σ^0_1 class generated by strings on which a martingale makes a lot of profit, must have small measure.

Lemma 3.4.5.

- *Let B be a martingale and let A be a prefix-free set of strings that covers Cantor space (that is: $[\![A]\!] = 2^\omega$). Then*

$$\sum_{\tau\in A} 2^{-|\tau|}B(\tau) = B(\lambda).$$

- *Let B be a (super)martingale and let A be any prefix-free set of strings. Then*

$$\sum_{\tau\in A} 2^{-|\tau|}B(\tau) \leq B(\lambda).$$

Proof.

- If B is a martingale, let $m(\sigma) = 2^{-|\sigma|}B(\sigma)$. By Lemma 2.2.1 there is a unique measure μ_m on Cantor space such that $m(\sigma) = \mu_m([\![\sigma]\!])$ for all strings σ. Hence

$$\begin{aligned}\sum_{\tau\in A} 2^{-|\tau|}B(\tau) &= \sum_{\tau\in A}\mu_m[\![\tau]\!] \\ &= \mu_m([\![A]\!]) \\ &= \mu_m(2^\omega) \\ &= B(\lambda),\end{aligned}$$

  as required.

- If B is a supermartingale, then we can define a martingale B' such that $B'(\sigma) \geq B(\sigma)$ for any string σ. Let $m(\sigma) = 2^{-|\sigma|}B'(\sigma)$. Again, by Lemma 2.2.1 there is a unique measure μ_m on Cantor space such that $m(\sigma) = \mu_m([\![\sigma]\!])$ for all strings σ. Hence

$$\begin{aligned}\sum_{\tau\in A} 2^{-|\tau|}B(\tau) &\leq \sum_{\tau\in A} 2^{-|\tau|}B'(\tau) \\ &= \sum_{\tau\in A}\mu_m[\![\tau]\!] \\ &= \mu_m([\![A]\!]) \\ &\leq \mu_m(2^\omega) \\ &= B(\lambda),\end{aligned}$$

  as required.

□

The next lemma shows how we can combine countably many martingales into one new martingale, that succeeds on a sequence Z whenever *any* of the

original martingales succeeds on Z.

Lemma 3.4.6.

Let $\alpha_0, \alpha_1, \ldots$ be uniformly computable real numbers. Let $B_0, B_1, \ldots$ be uniformly computable martingales such that $\sum_{i\in\mathbb{N}} \alpha_i B_i(\lambda)$ is a computable real number. Then

$$\sum_{i\in\mathbb{N}} \alpha_i B_i$$

is a computable martingale.

Proof. Write

$$B = \sum_{i\in\mathbb{N}} \alpha_i B_i.$$

Since $\sum_{i\in\mathbb{N}} \alpha_i B_i(\lambda)$ is finite, B is a well-defined real function. The linear sum of martingales also preserves the martingale equality (14), so B is a martingale. In order to compute a value $B(\sigma)$, observe that if

$$\sum_{i=k}^{\infty} \alpha_i B_i(\lambda) < 2^{-n-|\sigma|},$$

then

$$\sum_{i=k}^{\infty} \alpha_i B_i(\sigma)$$

contributes at most 2^{-n} to the value of $B(\sigma)$. As $\sum_{i\in\mathbb{N}} \alpha_i B_i(\lambda)$ is computable, we can find such a value for k effectively. Hence B is a computable martingale. □

Finally, Lemma 3.4.7 shows how to construct a martingale that makes a fixed amount of profit on a Σ^0_1 class. The amount of profit that can be made is inversely proportional to the measure of the Σ^0_1 class.

Lemma 3.4.7.

Let W be a prefix-free c.e. set such that $\mu[\![W]\!]$ is a computable real number. There exists a computable martingale B_W with initial capital $\mu[\![W]\!]$ such that

$$B_W(\sigma) = 1$$

for every σ that extends some element of W.

Proof. For any string σ, let B_σ be the martingale that starts with initial capital $2^{-|\sigma|}$ and bets everything on σ being an initial segment of the sequence, achieving a capital of 1 on all strings extending σ.

Let $\{\sigma_0, \sigma_1, \ldots\}$ be an effective enumeration of W. (Essentially the same argument works is W is finite.) Note that

$$\sum_{i\in\mathbb{N}} B_{\sigma_i}(\lambda) = \sum_{i\in\mathbb{N}} 2^{-|\sigma_i|} = \mu[\![W]\!].$$

So by Lemma 3.4.6

$$B_W = \sum_{i\in\mathbb{N}} B_{\sigma_i}$$

is a computable martingale that satisfies the requirements. □

Relation with Martin-Löf, Schnorr and Kurtz randomness

Theorem 3.4.8 (see [48, 7.3.2])**.**

Computable randomness implies Schnorr randomness.

Proof. Suppose that the sequence Z is not Schnorr random, as witnessed by the Schnorr test (U_i). Let V_i be a prefix-free c.e. set of generators of U_i.

Consider

$$B = \sum_{i \in \mathbb{N}} B_{V_i}.$$

By definition of a Schnorr test, the measures $\mu(U_i)$ are uniformly computable and $\mu(U_i) < 2^{-i}$, so

$$\sum_{i \in \mathbb{N}} B_{V_i}(\lambda) = \sum_{i \in \mathbb{N}} \mu(U_i)$$

is a computable real number. By Lemma 3.4.6, B is a computable martingale. As every B_{V_i} achieves a capital of 1 along Z, B succeeds on Z. Therefore Z is not computably random. □

Note that the computability of the measures $\mu(U_i)$ is needed in order to apply Lemma 3.4.6. Therefore the same argument does not work for Martin-Löf randomness instead of Schnorr randomness. Indeed, computable randomness does not imply Martin-Löf randomness, but the other direction does hold.

Theorem 3.4.9.

Martin-Löf randomness implies computable randomness.

Proof. Suppose that the sequence Z is not computably random, as witnessed by the martingale B. Suppose without loss of generality that $B(\emptyset) = 1$. We construct a Martin-Löf test (U_i) that succeeds on Z. The sets

$$U_i = \left\{ Y \in 2^\omega \ : \ \exists n \text{ such that } B(Y\upharpoonright_n) > 2^i \right\}$$

are Σ^0_1 classes uniformly in i. Moreover, we claim that $\mu(U_i) < 2^{-i}$. Indeed, by Lemma 3.4.5 we have

$$\sum_{\sigma \in X} 2^{-|\sigma|} B(\sigma) \leq B(\lambda) = 1$$

for every prefix-free set of strings X. Taking X to be the set of minimal strings σ (for the prefix order) such that $B(\sigma) > 2^i$, we get

$$\mu(U_i) = \sum_{\sigma \in X} 2^{-|\sigma|} < \frac{\sum_{\sigma \in X} 2^{-|\sigma|} B(\sigma)}{2^i} \leq 2^{-i}.$$

Consequently (U_i) is a Martin-Löf test, which by construction succeeds on Z. □

Though the above theorem only works in one direction, Martin-Löf randomness *can* be characterized using martingales. We just need to use the larger class of **c.e. martingales**. A c.e. martingale is a martingale B which is the limit of an increasing sequence of uniformly computable martingales. That is, $B = \lim B_s$ where B_s are computable martingales uniformly in s, with $(B_s(\sigma))$ an increasing sequence for all strings σ. Equivalently, a c.e. martingale is a martingale whose values are uniformly left-c.e..

Theorem 3.4.10 (Schnorr [53], see also [16, 6.3.4] or [48, 7.2.6])**.** *A sequence is Martin-Löf random if and only if no c.e. martingale succeeds on it.*

Proof. The proof of Theorem 3.4.9 still works when B is an c.e. martingale. Furthermore, the construction of Lemma 3.4.7 and the proof of Theorem 3.4.8 naturally adapt to give a c.e. martingale when starting from a Martin-Löf test instead of a Schnorr test. □

Computable martingales can succeed very slowly on non-random sequences. If we require that martingales make profits at least as quickly as some computable order, then we obtain weaker randomness notions. If we require that

$$\limsup_{n \to \infty} \frac{B(Z\restriction_n)}{h(n)} = \infty \tag{18}$$

for some computable order h, then we obtain Schnorr randomness. If we require that

$$\lim_{n\to\infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty \tag{19}$$

for some computable order h, then we obtain a different notion, namely Kurtz randomness.

Remark 3.4.11. *The fact that Schnorr randomness and Kurtz randomness are different notions, means that the savings trick (Remark 3.4.2) no longer works in this context. Indeed, suppose we try to apply the savings trick anyway to a martingale B and a computable order h with*

$$\limsup_{n\to\infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty. \tag{20}$$

We want to construct a new martingale B' and find a computable order h' such that

$$\lim_{n\to\infty} \frac{B'(Z\upharpoonright_n)}{h'(n)} = \infty.$$

To obtain the latter, we need to have a computable lower bound on how quickly the money in the savings account grows. We know from (20) that at some point we will have $B(Z\upharpoonright_n) > 2$, at which point we put 1 unit of money in the savings account, i.e. $B'(Z\upharpoonright_m) \geq 1$ for all $m \geq n$. Similarly, at some point we have $B(Z\upharpoonright_n) > 4$ and hence $B'(Z\upharpoonright_m) \geq 2$ for all $m \geq n$. And so on. However, the computability of the order h does not help to find a computable lower bound on how quickly the money in the savings account grows, as it doesn't tell us at which position we will have $B(Z\upharpoonright_n) > 2$, etc. Indeed, if the set of positions n where $B(Z\upharpoonright_n) \geq h(n)$ is sparse enough (certainly hyperimmune, i.e. the principal function, that maps k to the k'th element of this set, must not be dominated by any computable function), then the money

in the savings account might also grow more slowly than any computable order. Hence we do not get any suitable computable order h'.

We now give proofs that (18) indeed characterizes Schnorr randomness and that (19) indeed defines Kurtz randomness.

Theorem 3.4.12 (Schnorr [53], see also [16, 7.1.7] or [48, 7.3.3])**.** *A sequence is Schnorr random if and only if there does not exists a computable martingale B and a computable order h such that*

$$\limsup_{n\to\infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty.$$

Proof. If there exists a martingale B and a computable order h such that

$$\limsup_{n\to\infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty,$$

then we can adapt the proof of Theorem 3.4.9. The definition of U_i becomes

$$U_i = \left\{ Y \in 2^\omega \,:\, \exists n \text{ such that } \frac{B(Y\upharpoonright_n)}{h(n)} > 2^i \right\}.$$

In this case, $\mu(U_i)$ is computable, since values of n with $h(n) \geq 2^k$ can only contribute at most measure 2^{-i-k} to $\mu(U_i)$, by the same argument as in the proof of Theorem 3.4.9. Hence (U_i) is a Schnorr test that succeeds on Z.

For the other direction, suppose that (U_i) is a Schnorr test that succeeds on Z. Suppose $U_i = [\![W_i]\!]$, where the W_i are uniformly c.e. prefix-free sets of generators. Let $W = \cup_{i\in\mathbb{N}} W_i$. Given $k \in \mathbb{N}$, we can compute an $f(k)$ such that generators of length greater than $f(k)$ contribute a computable measure smaller than 2^{-2k} to all of the U_i together. That is: let

$$V_k = \{\sigma \in W \,:\, |\sigma| > f(k)\}.$$

Then

$$\sum_{\sigma \in V_k} 2^{-|\sigma|} < 2^{-2k}.$$

The martingales B_{V_k} from Lemma 3.4.7 make a capital 1 on every $\sigma \in V_k$. By Lemma 3.4.6, $B = \sum_{k \in \mathbb{N}} 2^k B_{V_k}$ is a computable martingale. If σ is an initial segment of Z with $\sigma \in W$ and $k \in \mathbb{N}$ is such that $f(k) < |\sigma| \leq f(k+1)$, then

$$B(\sigma) \geq 2^k B_{V_k} = 2^k.$$

As we can take σ arbitrarily long, we have

$$\limsup_{n \to \infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty,$$

for e.g. a computable order h that grows at the same rate as f^{-1}. □

Theorem 3.4.13 (Wang [63], see also [16, 7.2.13])**.**
A sequence is Kurtz random if and only if there does not exists a computable martingale B and a computable order h such that

$$\lim_{n \to \infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty.$$

Proof. First, suppose there exists a martingale B and a computable order h such that

$$\lim_{n \to \infty} \frac{B(Z\upharpoonright_n)}{h(n)} = \infty.$$

Let n_0 be such that $\frac{B(Z\upharpoonright_n)}{h(n)} \geq 1$ for all $n \geq n_0$. Given i, pick an $n \geq n_0$ such that $h(n) \geq 2^i$. Define

$$U_i = \left[\!\left[\{ \sigma \in 2^n \ : \ B(\sigma) \geq 2^i \} \right]\!\right].$$

Then (U_i) is a Kurtz test that succeeds on Z.

Conversely, suppose that (U_i) is a Kurtz test that succeeds on Z. Suppose $U_i = [\![D_i]\!]$ where the D_i are uniformly computable finite prefix-free sets of generators. As $\mu(U_i) < 2^i$, Lemma 3.4.6 gives that

$$B = \sum_{i \in \mathbb{N}} B_{D_i}$$

is a computable martingale. Define the computable order h by $h(n) = k$ where k is the least integer such that there are is a strings of length greater than n in D_k. If $h(n) > 0$, then all of $B_{D_0}, \ldots, B_{D_{h(n)-1}}$ achieve a capital of 1 on $Z\restriction_n$, so

$$\frac{B(Z\restriction_n)}{h(n)} \geq 1.$$

By picking a computable order h' that grows a little more slowly than h, we get

$$\lim_{n \to \infty} \frac{B(Z\restriction_n)}{h'(n)} = \infty,$$

as required. □

Partial and nonmonotonic computable randomness

The notion of computable randomness can be strengthened in two different directions. On the one hand we can allow *partial* betting strategies, giving rise to the notion of partial computable randomness. On the other hand, we can be flexible about the order in which we bet on the digits of a sequence, which gives us non-monotonic variations of computable randomness.

Partial computable randomness

Partial computable randomness is defined just like computable randomness, except that the martingales used are allowed to be *partial* computable functions $2^{<\omega} \to \mathbb{R}$. The martingale equality (14) only applies when all terms involved are defined. For a partial computable martingale B we also require by convention that $B(\sigma 0)\downarrow$ if and only if $B(\sigma 1)\downarrow$, and this can only be the case if already $B(\sigma)\downarrow$ at some earlier stage.

If a partial martingale B is to succeed on a sequence Z, then B must certainly be defined on all initial segments of Z (we also say: B *is defined along* Z). However, B may be undefined on some other strings. Intuitively, a partial computable betting strategy can be forever undecided on certain bets. For example, this allows us to make bets along the lines of: "wait until England wins another football World Cup, then bet on the next digit according to the final score of the final." As this might never happen, we cannot do this in a total betting strategy. In a partial betting strategy however, we can do this. Hence we have ways of betting that where not available to us before. That partial computable randomness is indeed a strictly stronger notion than (total) computable randomness will be proven in Theorem 4.1.1.

Non-monotonic computable randomness

Non-monotonicity (first introduced in the context of stochasticity by Kolmogorov [27] and Loveland [36] [35]) means that we are flexible in the order in which we bet on the digits of a sequence. We might for example bet on the second digit before betting on the first digit. Then how we bet on the first digit can depend on the value of the second digit. It is not important that we bet place a bet on all digits of a sequence, as we can ignore certain

digits anyway by betting evenly one them.

The order in which to bet on the digits can be fixed in advance, for example by a computable permutation or a computable injection. A (partial) permutation/injection betting strategy is then a pair $\langle f, B\rangle$ of a computable permutation/injection $f : \mathbb{N} \to \mathbb{N}$, which maps n to the position for the n'th bet, and a (partial) computable martingale B. The non-monotonic betting strategy $\langle f, B\rangle$ succeeds on a sequence Z if

$$\limsup_{n\to\infty} B\left((Z \circ f)\restriction_n\right) = \infty.$$

(Note that $(Z \circ f)$ conveniently gives us the sequence of digits that we bet upon in the correct order.) The resulting notions are **partial/total permutation randomness** and **partial/total injection randomness**. Total permutation randomness is in fact equivalent to computable randomness (see Theorem 3.4.14 below).

We can take non-monotonicity even further by not fixing the order in advance. Instead, we can allow the next position that we bet on to depend on the outcomes of the previous bets. For example, we might bet on the second position first. If the second digit is a zero, we might go back to bet on the first digit, while if the second digit turns out to be a one, then we might to on to bet on the tenth digit next. And so on, as long as we pick the next position in a computable way, and we don't bet on the same position more than once. The resulting notion is called **Kolmogorov-Loveland randomness**. Kolmogorov-Loveland betting strategies can be partial in the martingale as well as in the rule used to select the next position, but none of this matters, as the total and partial variations of Kolmogorov-Loveland randomness are equivalent (see Theorem 3.4.15 below).

Remarks 3.4.1-3.4.4 generalize to all partial and non-monotonic notions

of randomness as well. Moreover, the proof of Theorem 3.4.9 can be adapted to prove that Martin-Löf randomness not only implies computable randomness, but it even implies the strongest variation of computable randomness, namely Kolmogorov-Loveland randomness. It is an open question whether this implication is strict or not.

Two equivalences

As mentioned above, there are two equivalences involving the variations of computable randomness that we have introduced. Firstly, total permutation randomness is equivalent with (total) computable randomness. In other words, computable randomness is closed under computable permutations. Closure under computable permutations is trivial for randomness notions defined using measure theoretic tests such as Martin-Löf randomness, but not for variations of computable randomness. Indeed, partial computable randomness will turn out to be not closed under computable permutations. But first we prove that computable randomness is closed under computable permutations.

Theorem 3.4.14 (Buhrman et al. [7, Section 4], see also [24, section 2.3] and [48, 7.6.24])**.**
Total permutation randomness and computable randomness are equivalent.

Proof. For the non-trivial direction, suppose that (B, f) is a total permutation betting strategy that succeeds on a sequence Z. We can assume without loss of generality that B satisfies the savings property (17). We construct a

computable martingale $\overline{B}$ that succeeds on Z. Define for all strings σ

$$\overline{B}(\sigma) = \sum_{\substack{\tau \succ \sigma \\ |\tau| = l}} 2^{-(|\tau| - |\sigma|)} B(\tau \circ f\restriction_n)$$

where $l, n \in \mathbb{N}$ have values such that

$$\{0, \ldots, |\sigma| - 1\} \subseteq \{f(0), \ldots, f(n-1)\} \subseteq \{0, \ldots, l-1\}.$$

That is: in the first n bets the permutation betting strategy bets on all the first $|\sigma|$ digits, but on no other than the first $|\tau|$ digits, for the strings τ extending σ that we consider. $\overline{B}(\sigma)$ can be seen as the expected capital of the permutation betting strategy over long enough strings extending σ. It can be verified using induction that $\overline{B}(\sigma)$ is independent of the particular choice of l and n, and that $\overline{B}$ is a total computable martingale. Furthermore, there is for all $i \in \mathbb{N}$ an $n_i \in \mathbb{N}$ and an initial segment σ_i of Z such that $B(\sigma_i \circ f\restriction_{n_i}) > i+1$. Using the savings property (17), we get that $B(\tau \circ f\restriction_n) > i$ for all strings τ extending σ_i and suitable values of n. Consequently

$$\overline{B}(\sigma_i) > i$$

for all i. So $\overline{B}$ is a computable martingale that succeeds on Z, and Z is not computably random. □

Secondly, an elegant argument by Wolfgang Merkle shows that the total and partial versions of Kolmogorov-Loveland randomness are equivalent.

Theorem 3.4.15 (Merkle [41, Remark 6], see also [16, 7.5.4] and [48, 7.6.25])**.**
The partial and total versions of Kolmogorov-Loveland randomness are equivalent.

Proof. For the non-trivial direction, suppose that some partial Kolmogorov-Loveland betting strategy succeeds on a sequence Z. Then the betting strategy also succeeds by betting in the same way on either just the odd positions of Z and simply reading the even positions of Z, or the other way around. Suppose the former is the case. We define a total Kolmogorov-Loveland betting strategy that succeeds on Z as follows. Read successive even positions of the sequence, while trying to compute the next bet that the partial betting strategy prescribes. If this is a bet on an even position, then simply read that digit (if not already read) and continue as before. If it is a bet on an odd position, then do the same bet and continue as before. Note that if the partial betting strategy is not defined on some sequence Y, then the new betting strategy still keeps on reading more and more even positions, so it is in fact total. The total betting strategy bets in the same way as the partial betting strategy on the odd positions of Z, and by assumption this is enough for the betting strategy to succeed on Z. □

3.5 Randomness and differentiability

Another, more recent approach to defining randomness, is to define non-random real numbers as the points of differentiability of computable real functions. Here we are mainly speaking about the randomness of real numbers, as opposed to randomness of binary sequences. This difference is easily overcome by identifying a real number with the sequence containing its binary expansion. The only real numbers with two different binary expansions are rational numbers, and then both expansions are non-random (even computable), so there is no issue with this.

There are different ways of going about defining computability for real

functions. A common definition [50] goes as follows:

Definition.

A function $f : [0,1] \to \mathbb{R}$ *is* ***computable*** *if*

1. *there is a sequence* (x_i) *of uniformly computable real numbers that is dense in* $[0,1]$*, such that* $(f(x_i))$ *is uniformly computable, and*
2. f *is* uniformly effectively continuous*, i.e. there is a computable function* $h : \mathbb{N} \to \mathbb{N}$ *such that* $|f(y) - f(x)| < 2^{-n}$ *whenever* $x, y \in [0,1]$ *with* $|y - x| < 2^{-h(n)}$*.*

The sequence in point 1. of the definition could for example be an enumeration of all rational numbers, or all dyadic rational numbers.

For different classes of computable functions $[0,1] \to \mathbb{R}$, we can argue that is only possible to construct such a function that is not differentiable at a real number $x \in [0,1]$, if there is some regularity in x that can guide us in the construction. Different classes of functions lead to different notions of randomness. The larger the class of computable functions, the stronger the corresponding randomness notion. However, the class of *all* computable functions is too large, by the following theorem.

Theorem 3.5.1.

There exists a computable function $[0,1] \to \mathbb{R}$ *which is nowhere differentiable.*

Proof. We use the construction of the blancmange function[4], a well-known example of a continuous function that is nowhere differentiable, and we show that this function is computable.

[4]The function is named after the dessert *blancmange* because the shape of its curve resembles the shape of the dessert. It is also known as the Takagi curve, after the Japanese mathematician Teiji Takagi who first defined it [59].

The blancmange function is an infinite sum of *sawtooth* functions. Specifically, letting $s : \mathbb{R} \to \mathbb{R}$ be the function mapping every real to the distance to the closest integer, we use the sawtooth functions

$$\begin{aligned} f_n : [0,1] &\to \mathbb{R} \\ x &\mapsto \frac{s(2^n x)}{2^n} \end{aligned}$$

and define the blancmange function as

$$g = \sum_{i \in \mathbb{N}} f_i.$$

We also define the partial sums

$$g_n = \sum_{i=0}^{n} f_i$$

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

We first prove that g is not differentiable at any point $z \in [0,1]$. First suppose that z is a dyadic rational, in particular that $z = \frac{i}{2^n}$ where i is odd. (This supposes $z \notin \{0,1\}$, but these cases can be treated similarly.) Now f_i is differentiable at z for $i \leq n-2$. The sum $f_{n-1} + f_n$ is constant (and hence also differentiable) at z. Hence differentiability of g at z is equivalent with differentiability of

$$h_n = g - g_n = \sum_{i=n+1}^{\infty} f_i$$

at z. A direct calculation shows that $h_n(z) = 0$ and

$$h_n\left(z + \frac{1}{2^{n+m+1}}\right) = \frac{m}{2^{n+m+1}}$$

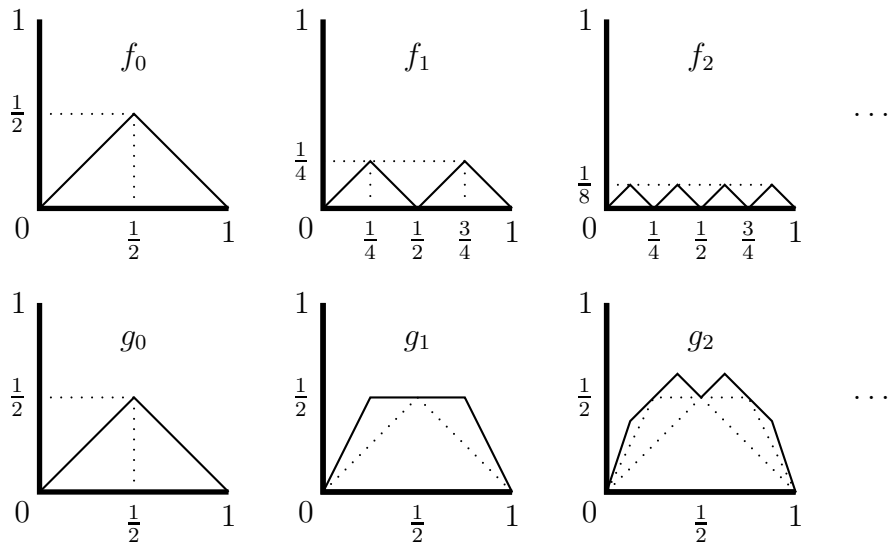

Figure 5: The sawtooth functions f_n and partial sums g_n used to define the blancmange function.

for all $m \in \mathbb{N}$. So the slopes

$$\frac{h_n\left(z + \frac{1}{2^{n+m+1}}\right) - h_n(z)}{\frac{1}{2^{n+m+1}}} = m$$

diverge for $m \to \infty$, contradicting differentiability of h_n at z. Therefore g is not differentiable at z.

Secondly, suppose that z is not a dyadic rational. Then for each n, there is an $i_n \in \mathbb{N}$ such that $z \in \left(\frac{i_n}{2^n}, \frac{i_n+1}{2^n}\right)$. If g were differentiable at z, then the slopes

$$\frac{g\left(\frac{i_n+1}{2^n}\right) - g\left(\frac{i_n}{2^n}\right)}{2^n} \tag{21}$$

must converge to the derivative at z for $n \to \infty$. However, we have

$$\begin{aligned}\frac{g\left(\frac{i_n+1}{2^n}\right) - g\left(\frac{i_n}{2^n}\right)}{2^n} &= \frac{g_{n-1}\left(\frac{i_n+1}{2^n}\right) - g_{n-1}\left(\frac{i_n}{2^n}\right)}{2^n} \\ &= g'_{n-1}(z),\end{aligned}$$

the first equality because f_i is 0 on multiples of $\frac{1}{2^n}$ for $i \geq n$, the second equality because $f_0, \ldots, f_{n-1}$ are linear on the intervals $\left[\frac{i}{2^n}, \frac{i+1}{2^n}\right]$ for any i. Moreover,

$$\left|g'_n(z) - g'_{n-1}(z)\right| = |f'_n(z)| = 1$$

for all n, so $(g'_n(z))$ is not a Cauchy sequence, contradicting the convergence of (21). So g is not differentiable at z.

It remains to show that g is computable as a real function $[0,1] \to \mathbb{R}$. It is certainly uniformly computable on dyadic rationals, where only a finite sum is involved. For uniform effective continuity, let $\epsilon > 0$ be given. Pick some n such that $2^{-n} < \frac{\epsilon}{3}$, and hence $|g(x) - g_n(x)| \leq 2^{-n} < \frac{\epsilon}{3}$ for all $x \in [0,1]$. Take $\delta = \frac{\epsilon}{3(n+1)}$ and take any $x, y \in [0,1]$ with $|x - y| < \delta$. Because g_n consists of line segments whose slopes have absolute values of at most $n+1$, we have $|g_n(x) - g_n(y)| \leq (n+1)\delta = \frac{\epsilon}{3}$. So finally,

$$|g(x) - g(y)| \leq |g(x) - g_n(x)| + |g_n(x) - g_n(y)| + |g_n(y) - g(y)| < \epsilon,$$

as required. □

By considering smaller classes of computable functions, we obtain new definitions for different notions of randomness.

Theorem 3.5.2 (Brattka, Miller and Nies [6])**.**

a. A real number $x \in [0,1]$ is computably random if and only if every nondecreasing computable function $[0,1] \to \mathbb{R}$ is differentiable at x.

b. A real number $x \in [0,1]$ is Martin-Löf random, if and only if every computable function $[0,1] \to \mathbb{R}$ of bounded variation is differentiable at x, if and only if every absolutely continuous computable function $[0,1] \to \mathbb{R}$ is differentiable at x.

c. A real number $x \in [0,1]$ is weakly 2-random if and only if every almost everywhere differentiable computable function $[0,1] \to \mathbb{R}$ is differentiable at x.

For exact definitions and proofs, see [6]. We limit ourselves to providing a few remarks about the result for computable randomness. Suppose we have a computable martingale B, such that there is no sequence Z such that

$$\lim_{n\to\infty} \frac{B(Z\restriction_n)}{2^n} > 0.$$

This extra condition is necessary to make the function f below effectively uniformly continuous, and is no real restriction, since such a sequence Z is always computable, being an isolated path of a Π^0_1-class. We define a nondecreasing computable function f by setting

$$f(x) = \sum_{\sigma \in X} 2^{-|\sigma|} B(\sigma)$$

where X is some prefix-free set of strings that generate the class of reals in $[0,1]$ which are less than x. It is straight-forward to verify that if B succeeds on x, then f is not differentiable at x. Indeed, if $x \in [q, q + 2^{-n}]$, where q is

a integer multiple of 2^{-n}, and $B(x\restriction_n) > k$, then we have

$$\frac{f(q + 2^{-n}) - f(q)}{2^{-n}} = B(x\restriction_n) > k,$$

so the slopes of f around x are unbounded.

Conversely, from a nondecreasing computable function such that $f(0) = 0$, one can reconstruct the corresponding martingale. Still, this is not enough to prove the other direction of Theorem 3.5.2a, because even when a martingale B does not succeed on x, it could still be that the corresponding function f is not differentiable at x. Details on how to get around this are in [6].

Base-invariance of computable randomness

Up to now, we have only considered randomness in base 2, where reals in $[0, 1]$ correspond to infinite sequences of zeroes and ones. We could also consider expansions in other bases and therefore define randomness for infinite sequences in 3^ω, 4^ω, ... The definitions for Martin-Löf randomness, computable randomness, and most other randomness notions translate in a natural way to other bases.

Definition.

A randomness notion which can be defined in any base is ***base-invariant*** *if for every $k, l \geq 2$ and every real number $x \in [0, 1]$, the base k expansion of x is random (among sequences in k^ω) if and only if the base l expansion of x is random (among sequences in l^ω).*

For notions defined using measure theory, like Martin-Löf randomness, the base that is used does not influence the randomness of the expansion of a real number. Hence Martin-Löf randomness is base-invariant. Computable

randomness however, is defined using martingales, which directly use the digits of the expansion rather than its value as a real number. Therefore there is no immediate way of proving that computable randomness is base-invariant. Theorem 3.5.2a provides the only known proof of this fact, as it gives a characterization of computable randomness using differentiability of real functions, which does not depend on the base used.

The base-invariance of other variants of computable randomness, like partial computable randomness and Kolmogorov-Loveland randomness, remains an open problem. There is also an additional complication in defining base k partial computable randomness, for $k > 2$. When considering partial computable martingales

$$B : k^{<\omega} \to [0, \infty),$$

do we require that whenever $B(\sigma i)$ is defined for some $\sigma \in k^{<\omega}$ and some $i \in \{0, \ldots, k-1\}$, then also $B(\sigma i)$ is defined for all other values of $i \in \{0, \ldots, k-1\}$? In terms of betting strategies: whenever we make a decision on how much money we want to make on one outcome for the next digit, do we immediately have to decide on how much to bet on the other outcomes as well? Let's call the resulting notion **weak base k partial computable randomness**. Alternatively, do we allow ourselves to think a little longer about how to bet on the other outcomes, potentially never coming to a decision about this at all? Let's call this **strong base k partial computable randomness**. Note that this last notion can only be formalized using supermartingales rather than martingales, using the condition that

$$B(\sigma) \geq \frac{1}{k} \left(\sum_{\substack{i \in \{0,1,\ldots,k-1\} \\ B(\sigma i)\downarrow}} B(\sigma i) \right) \tag{22}$$

for all strings σ such that $B(\sigma)\downarrow$.

Let $(B^k_n)_{n\in\mathbb{N}}$ be an effective enumeration of all weak base k partial computable supermartingales, and $(\overline{B}^k_n)_{n\in\mathbb{N}}$ an effective enumeration of all strong base k partial computable supermartingales.

Theorem 3.5.3.

Let $k > 2$. There is a strong base k partial computable supermartingale $\overline{B}$ that succeeds on a different set of sequences than any weak base k partial computable supermartingale.

Proof. For any $n \in \mathbb{N}$, we will use strings that extend 0^n1 to diagonalize against the n'th weak base k partial computable supermartingale B^k_n. We will make sure that $\overline{B}$ succeeds on some sequence that extends 0^n1 on which B^k_n does not succeed.

For any n, set $\overline{B}(0^n) = 1$ and $\overline{B}(0^n1) = 1$. Furthermore, once $\overline{B}$ is defined on some string $\sigma = 0^n1\tau$, say $\overline{B}(\sigma) = x$, set

$$\overline{B}(\sigma 0) = \frac{3}{2}x.$$

Then wait until $B^k_n(\sigma 0)$ halts (if ever). Because B^k_n is a weak supermartingale, we then also have $B^k_n(\sigma i)\downarrow$ for all $i \in \{0, \ldots, k-1\}$. For at least one such i, we have $B^k_n(\sigma i) \leq B^k_n(\sigma)$. Pick the least such i, and set

$$\overline{B}(\sigma i) = \frac{3}{2}x.$$

(If $i = 0$, this was already defined before.) As $k \geq 3$, $\overline{B}$ satisfies the supermartingale inequality in the sense of (22). So $\overline{B}$ is a strong base k partial computable supermartingale.

Take any $n \in \mathbb{N}$. We now claim that $\overline{B}$ succeeds on some sequence on which B_n^k does not succeed. Consider the sequence Z with

$$Z\restriction_{n+1} = 0^n 1$$

and which is further defined inductively as follows:

$$Z(m) = \begin{cases} \text{the least } i \text{ such that } B_n^k(Z\restriction_m i) \leq B_n^k(Z\restriction_m) & \text{if } B_n^k(Z\restriction_m 0)\downarrow, \\ 0 & \text{otherwise,} \end{cases}$$

for any $m \geq n+1$. By definition, B_n^k does not succeed on this sequence. However, $\overline{B}$ does succeed on this sequence, as

$$\overline{B}(Z\restriction_{n+1+m}) = \left(\frac{3}{2}\right)^m$$

for any $m \in \mathbb{N}$. □

It might still be possible that the strong and weak notions of base k partial computable randomness coincide. That is: even though the individual success sets of the different types of martingales are different, the union of the success sets might still be the same for both types. This is an open problem.

Question 3.5.4.

For $k > 2$, are the notions of weak and strong base k partial computable randomness equivalent?

3.6 Randomness and ergodic theory

The following theorem is implicitly due to Kučera ([29], Lemma 3):

Theorem 3.6.1.
Let U be a Σ^0_1 class of measure less than 1. If Z is a Martin-Löf random sequence, then some tail of Z is not in U.

Proof. We prove the contrapositive: if all tails of Z are in U, then Z is not Martin-Löf random.

As U is a Σ^0_1 class, there is a prefix-free c.e. set W such that $U = [\![W]\!]$. Let

$$W^n = \{\sigma_0\sigma_1\ldots\sigma_{n-1} \,:\, \sigma_0, \sigma_1, \ldots, \sigma_{n-1} \in W\}$$

and

$$U_n = [\![W^n]\!].$$

By induction $Z \in U_n$ for every n. The U_n are Σ^0_1 classes uniformly in n, and because W is prefix-free we have

$$\mu(U_n) = \mu(U)^n.$$

Now pick an n_0 such that $\mu(U)^{n_0} < \frac{1}{2}$ and let

$$V_n = U_{n_0 \cdot n}.$$

Then (V_n) is a Martin-Löf test that succeeds on Z. So Z is not Martin-Löf random. □

Corollary 3.6.2.

Let (U_n) be a universal Martin-Löf test and let n be a positive integer. The following are equivalent:

1. *Z is Martin-Löf random*
2. *for any Σ^0_1 class U of measure less than 1, some tail of Z is not in U.*
3. *for any Σ^0_1 class U of measure less than 1, infinitely many tails of Z are not in U.*
4. *infinitely many tails of Z are not in U_n.*
5. *some tail of Z is not in U_n.*

Proof. 1. $\implies$ 2. is Theorem 3.6.1.

For 2. $\implies$ 3., note that any tail of a Martin-Löf random sequence is itself Martin-Löf random. So we can apply 1. $\implies$ 2. to any tail of Z, to find infinitely many different tails of Z that are not in U.

3. $\implies$ 4. is immediate as U_n is a Σ^0_1 class of measure less than 1.

For 5. $\implies$ 1., note that (U_n) is a universal test, so any sequence that is not Martin-Löf random, is contained in U_n. As some tail of Z is not in U_n, that tail must be Martin-Löf random, so Z itself must be as well. □

4. and 5. are interesting because they give us characterizations of Martin-Löf randomness that only involve one particular Σ^0_1 class, instead of infinitely many as in a Martin-Löf test.

2. and 3. are interesting because they give us a characterization of Martin-Löf randomness in terms of ergodic theory. Ergodic theory deals with ergodic transformations. Those are measure-preserving transformations T of some

space X, such that whenever $T^{-1}(E) = E$, E has measure 0 or measure 1. Intuitively, ergodic transformations are transformations that mix up the whole space, without keeping any two sizeable subsets of X separated. The shift operator, that removes the first digit from a sequence, is an ergodic transformation of the Cantor space by Kolmogorov's 0-1 law (see [16, 1.2.4] or [48, 1.9.12]). This starts off an interaction between algorithmic randomness and ergodic theory. For example, 3. can be reformulated using terminology from ergodic theory to say that Z is Martin-Löf random if and only if Z is a Poincaré point for the shift operator with respect to the collection of all Π^0_1 classes [18]. See also [2] and [1] for more results connecting randomness and ergodic theory.

3.7 Comparison of stochasticity and randomness

Now we have rigorously defined the necessary notions of randomness, we can study the relation between stochasticity and randomness in detail. First, we show how to turn a selection rule into a martingale. This proves that partial computable randomness implies Mises-Wald-Church stochasticity, that computable randomness implies Church stochasticity and that Schnorr randomness implies weak Church stochasticity. In particular, sequences that are random for Schnorr randomness or stronger notions, always satisfy the law of large numbers. (This does *not* hold for Kurtz randomness.) Next, we study how randomness tests can be constructed directly from selection rules. This provides an alternative way to prove some of the results from the preceding subsection. Finally, we show that Ville's theorem does not apply to any of the randomness notions that we have defined, not even to the weakest notion of Kurtz randomness.

From selection rules to martingales

Theorem 3.7.1.

Partial computable randomness implies Mises-Wald-Church stochasticity.

Proof (adapted from [16, p. 302]). Suppose a sequence Z is not Mises-Wald-Church stochastic. Without loss of generality we suppose that there is a partial computable selection rule s such that $s(Z)$ is an infinite sequence and $\frac{\text{zeroes}(s[Z]\restriction_n)}{n} > \frac{1}{2} + \epsilon$ for infinitely many n, where $\epsilon > 0$ is a fixed constant. For every computable real $x \in (0, 1)$, we define a partial computable betting strategy B^x, such that B^x succeeds on Z for any x that is small enough.

Given $B^x(\sigma)$, the betting strategy tries to compute $s(\sigma)$. If $s(\sigma)\downarrow = \text{NO}$, then $B^x(\sigma 0) = B^x(\sigma 1) = B^x(\sigma)$. If $s(\sigma)\downarrow = \text{YES}$, then $B^x(\sigma 0) = (1+x)B^x(\sigma)$ and $B^x(\sigma 1) = (1-x)B(\sigma)$. Now, let n be such that

$$\frac{\text{zeroes}\left(s[Z]\restriction_n\right)}{n} > \frac{1}{2} + \epsilon,$$

and $m(n)$ such that $|s(Z\restriction_{m(n)})| = n$. Then

$$B^x(Z\restriction_{m(n)}) > (1+x)^{n\left(\frac{1}{2}+\epsilon\right)}(1-x)^{n\left(\frac{1}{2}-\epsilon\right)},$$

so

$$\log(B^x(Z\restriction_{m(n)})) > n\left(\left(\frac{1}{2}+\epsilon\right)\log(1+x) + \left(\frac{1}{2}-\epsilon\right)\log(1-x)\right).$$

The function

$$h : [0,1) \to \mathbb{R} : x \mapsto \left(\frac{1}{2}+\epsilon\right)\log(1+x) + \left(\frac{1}{2}-\epsilon\right)\log(1-x)$$

has a derivative $h'(x) = \frac{\left(\frac{1}{2}+\epsilon\right)}{1+x} - \frac{\left(\frac{1}{2}-\epsilon\right)}{1-x}$ which satisfies $h'(0) = 2\epsilon > 0$. Since $h(0) = 0$, we have $h(x) > 0$ for $x \in (0,1)$ small enough. For such an x that is computable, we have

$$\log(B^x(Z\restriction_{m(n)})) > n\, h(x)$$

and since we have infinitely many choices for n, we have

$$\limsup_{m\to\infty} B^x(Z\restriction_m) = \infty.$$

So B^x is a partial computable betting strategy that succeeds on Z, as required. □

Theorem 3.7.2.

Computable randomness implies Church stochasticity.

Proof. The construction in the proof of Theorem 3.7.1 can be copied exactly, noting that if s is a total computable selection rule, then every B^x is a total computable martingale. □

Theorem 3.7.3.

Schnorr randomness implies weak Church stochasticity.

Proof. The construction in the proof of Theorem 3.7.1 can again be used. Indeed, suppose s is a computable selection rule such that $s(Y)$ is defined and infinite for every Y. Let $g(m) = \min_{\sigma\in 2^m} |s[\sigma]|$. This is a computable order by König's Lemma. Let g' be any computable order that grows more slowly than $\exp(g)$, Then

$$\log(B^x(Z\restriction_{m(n)})) > n\, h(x) \geq g(m(n))\, h(x)$$

for infinitely many values of n, so

$$\limsup_{m\to\infty} \frac{B^x(Z\upharpoonright_m)}{g'(m)} = \infty,$$

as required by Theorem 3.4.12. □

From selection rules to randomness tests

In this section, we directly construct Martin-Löf tests that capture sequences that are not Mises-Wald-Church stochastic, and Schnorr tests that capture sequences that are not weakly Church stochastic. There results are implied by the previous section. However, I feel that the different approach has its own merit.

We first prove that any sequence that every Schnorr random sequence satisfies the law of large numbers.

Theorem 3.7.4.

Every Schnorr random satisfies the law of large numbers.

Proof. Let Z be a sequence that does not satisfy the law of large numbers. Without loss of generality, we suppose that

$$\frac{\text{zeroes}(Z\upharpoonright_n)}{n} > \frac{1}{2} + \epsilon$$

for some fixed $\epsilon > 0$ and for infinitely many values of n. Define

$$V_i = \left\{ [\![\sigma]\!] \,:\, \sigma \in 2^i \text{ and } \tfrac{\text{zeroes}(\sigma)}{i} > \tfrac{1}{2} + \epsilon \right\}$$

and

$$U_i = \bigcup_{j=i}^{\infty} V_j.$$

(We suppose $i > 0$ throughout this proof.) The classes U_i are uniformly Σ^0_1 and by definition $Z \in \cap_{i\in\mathbb{N}} U_i$. To bound and compute the measure of U_i, we cam use some *concentration inequality* from probability theory. For example, Hoeffding's [23] inequality gives that

$$\mu(V_i) < e^{-2\epsilon^2 i}.$$

(A similar bound can be obtained using related inequalities like the Bernstein inequalities or the Chernoff bound [13].) Hence

$$\begin{aligned}
\mu(U_i) &\leq \sum_{j=i}^{\infty} \mu(V_j) \\
&< \sum_{j=i}^{\infty} e^{-2\epsilon^2 i} \\
&< \int_{i-1}^{\infty} e^{-2\epsilon^2 i}\, di \\
&= \frac{1}{2\epsilon^2} e^{-2\epsilon^2 (i-1)}.
\end{aligned}$$

This provides a computable bound on how fast $\mu(U_i)$ converges to 0, so (U_i) is a Martin-Löf test. Moreover, as $\mu(V_j)$ is computable uniformly in j, and $\sum_{j=i}^{\infty} \mu(V_j)$ converges to 0 at a computable rate as $i \to \infty$, the measures $\mu(U_i)$ are computable uniformly in i. So (U_i) is a Schnorr test. As $Z \in \cap_{i\in\mathbb{N}} U_i$, Z is not Schnorr random. □

This proof does not produce a Kurtz test. Indeed, remember that there are Kurtz random sequences that do not satisfy the law of large numbers.

More generally, when we require the law of large numbers to hold not just for the sequence itself, but for any subsequence obtained by some computable selection rule, then we obtain stochasticity. To what extent can the proof of Theorem 3.7.4 be adapted to show that randomness implies stochasticity?

We first prove a lemma.

Lemma 3.7.5. *1. Let s be a partial selection rule. Consider the set $s^{-1}[[\![\sigma]\!]]$ of sequences Z such that $s(Z)$ is an infinite sequence that starts with σ. Also consider the set $[\![s^{-1}[\sigma]]\!]$ of sequences Z such that $s(\tau) = \sigma$ for some initial segment τ of Z. Then*

$$\mu(s^{-1}[[\![\sigma]\!]]) \leq \mu([\![s^{-1}[\sigma]]\!]) \leq \mu([\![\sigma]\!]) = 2^{-|\sigma|}.$$

2. Let s be a selection rule such that $s(Z)$ is infinite for every sequence Z. Then

$$\mu(s^{-1}[[\![\sigma]\!]]) = \mu([\![s^{-1}[\sigma]]\!]) = \mu([\![\sigma]\!]) = 2^{-|\sigma|}.$$

Proof. 1. The first inequality is trivial since $s^{-1}[[\![\sigma]\!]] \subseteq [\![s^{-1}[\sigma]]\!]$. For the second inequality, define a partial computable betting strategy B that starts with an initial capital of $2^{-|\sigma|}$. When betting on a sequence Z, B computes s along Z, and on the first $|\sigma|$ positions that are selected, bets everything on the digits being the corresponding digits of σ. Hence, B makes a capital of 1 on every element of $[\![s^{-1}[\sigma]]\!]$. Let S be the prefix-free set of minimal strings in $s^{-1}[\sigma]$. By Lemma 3.4.5,

$$\begin{aligned}
\mu([\![s^{-1}[\sigma]]\!]) &= \sum_{\sigma \in S} 2^{-|\sigma|} \\
&= \sum_{\sigma \in S} 2^{-|\sigma|} B(\sigma) \\
&\leq B(\lambda) \\
&= 2^{-|\sigma|},
\end{aligned}$$

proving the second inequality.

2. If $s(Z)$ is total for every Z, then $s^{-1}[[\![\sigma]\!]] = [\![s^{-1}[\sigma]]\!]$, proving the first equality. Also, let S be the set of minimal strings τ such that s selects a string of length $|\sigma|$ from τ. Then S is a prefix-free set of strings that covers Cantor space. Define B as above, we have for $\tau \in S$ that

$$B(\tau) = \begin{cases} 1 & \text{if } s(\tau) = \sigma, \\ 0 & \text{otherwise.} \end{cases}$$

By Lemma 3.4.5

$$\begin{aligned} \mu([\![s^{-1}[\sigma]]\!]) &= \sum_{\substack{\tau \in S \\ s(\tau)=\sigma}} 2^{-|\tau|} \\ &= \sum_{\tau \in S} 2^{-|\tau|} B(\tau) \\ &= B(\lambda) \\ &= 2^{-|\sigma|}, \end{aligned}$$

as required.

□

Since open cylinders generate the Borel σ-algebra, this lemma shows that $\mu(s^{-1}[\mathcal{A}]) \leq \mu(\mathcal{A})$ for any Borel set $\mathcal{A} \subseteq 2^\omega$.

Moreover, if $s(Z)$ is infinite for every sequence Z, then $\mu(s^{-1}[\mathcal{A}]) = \mu(\mathcal{A})$ for any Borel set $\mathcal{A}$. That is, $\mathcal{A} \mapsto s[\mathcal{A}]$ is a measure preserving transformation of Cantor space.

We are now ready to give an alternative proof of the following corollary of Theorem 3.7.1.

Theorem 3.7.6.

Martin-Löf randomness implies Mises-Wald-Church stochasticity.

Proof. Define

$$V_i = \left\{ [\![\sigma]\!] \,:\, \sigma \in 2^i \text{ and } \tfrac{\text{zeroes}(\sigma)}{i} > \tfrac{1}{2} + \epsilon \right\}$$

as in Theorem 3.7.4. If s is a partial computable selection rule then we also define

$$\begin{aligned} V_i^s &= \left\{ [\![\tau]\!] \,:\, \tau \in 2^{<\omega} \text{ and } |s[\tau]| = i \text{ and } \tfrac{\text{zeroes}(s[\tau])}{i} > \tfrac{1}{2} + \epsilon \right\} \\ &= \left\{ s^{-1}[[\![\sigma]\!]] \,:\, \sigma \in 2^i \text{ and } \tfrac{\text{zeroes}(\sigma)}{i} > \tfrac{1}{2} + \epsilon \right\}. \end{aligned}$$

By Lemma 3.7.5 we have $\mu(V_i^s) \leq \mu(V_i)$, so we get a Martin-Löf test (U_i^s) as in the proof of Theorem 3.7.4. Every sequence Z such that $s(Z)$ is infinite and does not satisfy the law of large numbers, fails this test, as required. □

Contrary to the proof of Theorem 3.7.4, we do not get a Schnorr test here. This is because an inequality in $\mu(V_i^s) \leq \mu(V_i)$ might make the measures $\mu(V_i^s)$ incomputable. This is even the case if we only allow total computable selection rules. Indeed, Yongge Wang proved in his PhD thesis [63] (see also [16, p. 330]) that Schnorr randomness does not imply Church stochasticity.

However, we already proved in Theorem 3.7.3 that Schnorr randomness *does* imply *weak* Church stochasticity. We give an alternative proof of this, using our current approach.

Theorem 3.7.7.

Schnorr randomness implies weak Church stochasticity.

Proof. We proceed exactly like in the proof of Theorem 3.7.6 above. However, we now only need to consider selection rules s that select an infinite sequence $s[Z]$ from every sequence Z. By Lemma 3.7.5 we know that such a selection

rule satisfies $\mu(s^{-1}[[\sigma]]) = \mu([[\sigma]])$ for every string σ. Consequently we have an equality in $\mu(V_i^s) = \mu(V_i)$ and we get a Schnorr test just like in the proof of Theorem 3.7.4. □

Randomness versus stochasticity: Summary

The relations between randomness and stochasticity notions are summarized in Figure 6. No additional implications hold between these notions, other than the ones implied by Figure 6. Indeed, we will prove that partial computable randomness is strictly stronger than computable randomness in Theorem 4.1.1. Nies, Stephan and Terwijn proved that every high Turing degree contains a sequence that is Schnorr random but not computably random (see Theorem 4.3.1 below). Kurtz randomness does not imply any other notion because it is the only notion with sequences that do not satisfy the law of large numbers. Ville's theorem prohibits any implications from stochasticity notions to randomness notions (see next subsection). As mentioned above, Schnorr randomness does not imply Church stochasticity as proven by Wang. Finally, Klaus Ambos-Spies proved that computable randomness does not imply Mises-Wald-Church stochasticity (see [16, 7.4.7]).

We can now also justify our newly defined notion of weak Church stochasticity. We have proven that it is different from the other stochasticity notions, in that it is implied by Schnorr randomness where the others are not. Weak Church stochasticity seems to fit into Figure 6 naturally, corresponding to Schnorr randomness, just like Church stochasticity corresponds to computable randomness and Mises-Wald-Church stochasticity corresponds to partial computable randomness. This correspondence can be further extended by defining non-monotonic notions of stochasticity in analogy with the non-monotonic versions of computable randomness.

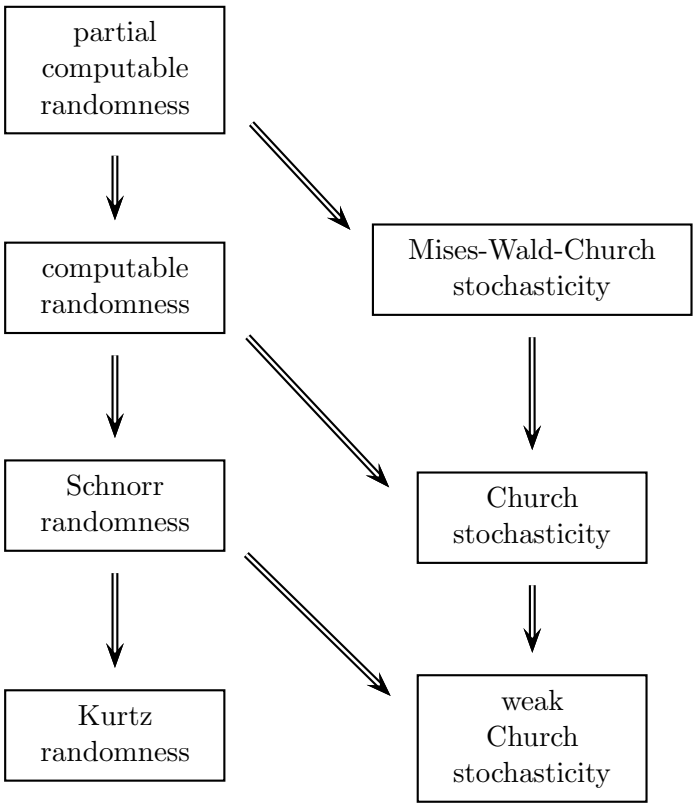

Figure 6: The relations between randomness and stochasticity notions. No additional implications hold between these notions.

Randomness and Ville's theorem

Remember that Ville's theorem (Theorem 3.1.2) showed that there are always some sequences with at least as many zeroes as ones in every initial segment, that are nonetheless stochastic. We now show that such sequences cannot be random, not even for the weakest notion of Kurtz randomness. Hence, Ville's theorem does not pose an objection to any of our definitions of randomness.

Lemma 3.7.8.

Let A_n be the set of strings of length n such that every initial segment has at least as many zeroes as ones.

$$|A_n| = \binom{n}{\lfloor \frac{n}{2} \rfloor}.$$

I give an elegant combinatorial proof, which is attributed by Feller [17] to E. Nelson. The method is similar to the *reflection method* solution to Bertrand's Ballot Problem (see e.g. [52]).

Proof. First suppose n is even.

Instead of counting strings, we will use a more visual approach. We will count the number of walks on the nonnegative integers (i.e. on the graph whose vertices are the nonnegative integers, and whose edges connect exactly the consecutive integers) of length n and starting at 0. The strings that we need to count in the theorem are in one-on-one correspondence with these walks, by letting the digit 0 correspond to a step to the next integer, and the digit 1 correspond to a step to the previous one. We will put these walks on the nonnegative integers in a one-on-one correspondence with the walks on the integers of length n that start at 0 and also end at 0. Of these there are exactly $\binom{n}{\frac{n}{2}}$, because to define such a walk, we need just to specify exactly which $\frac{n}{2}$ of the n steps will be to the next integer.

To make the required one-on-one correspondence between the two classes of walks, we represent each walk by the graph of the function $\{0, \ldots, n\} \to \mathbb{Z}$ that maps each $i \in \{0, \ldots, n\}$ to the position after exactly i steps of the walk.

Suppose that we are given a walk on the nonnegative integers of length n, starting at 0, and ending at some position m (where m must be even since n is even). To find the corresponding walk on the integers that starts and ends at 0, find the last step at which the given walk is at position $\frac{m}{2}$. Reflect the part of the graph to the right of this step around a vertical reflection axis, and put it in front of the other part of the graph, like in Figure 7. The resulting graph represents a walk on the integers of length n that starts and ends at 0.

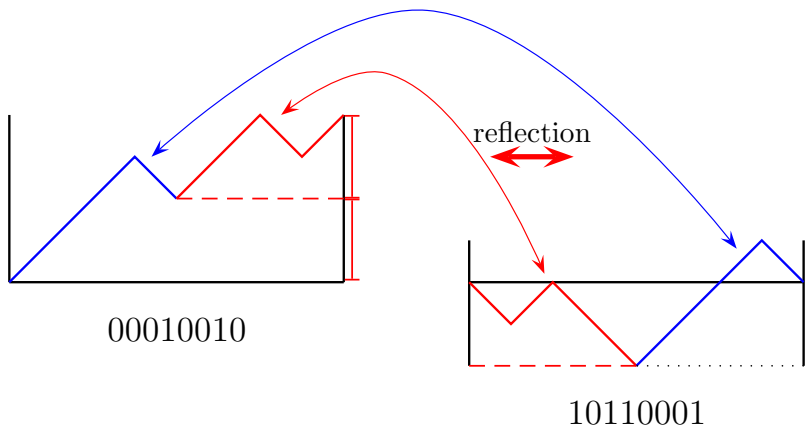

Figure 7: The one-on-one correspondence between walks on the nonnegative integers of length n starting at 0, and the walks on the integers of length n both starting and ending at 0.

In the other direction, suppose that we are given a walk on the integers of length n that starts and ends at 0. We can recover the corresponding walk on the nonnegative integers by finding the first step at which the new walk reaches its minimal position, reflecting the part of the graph to the left of this step, and putting it behind the other part of the graph.

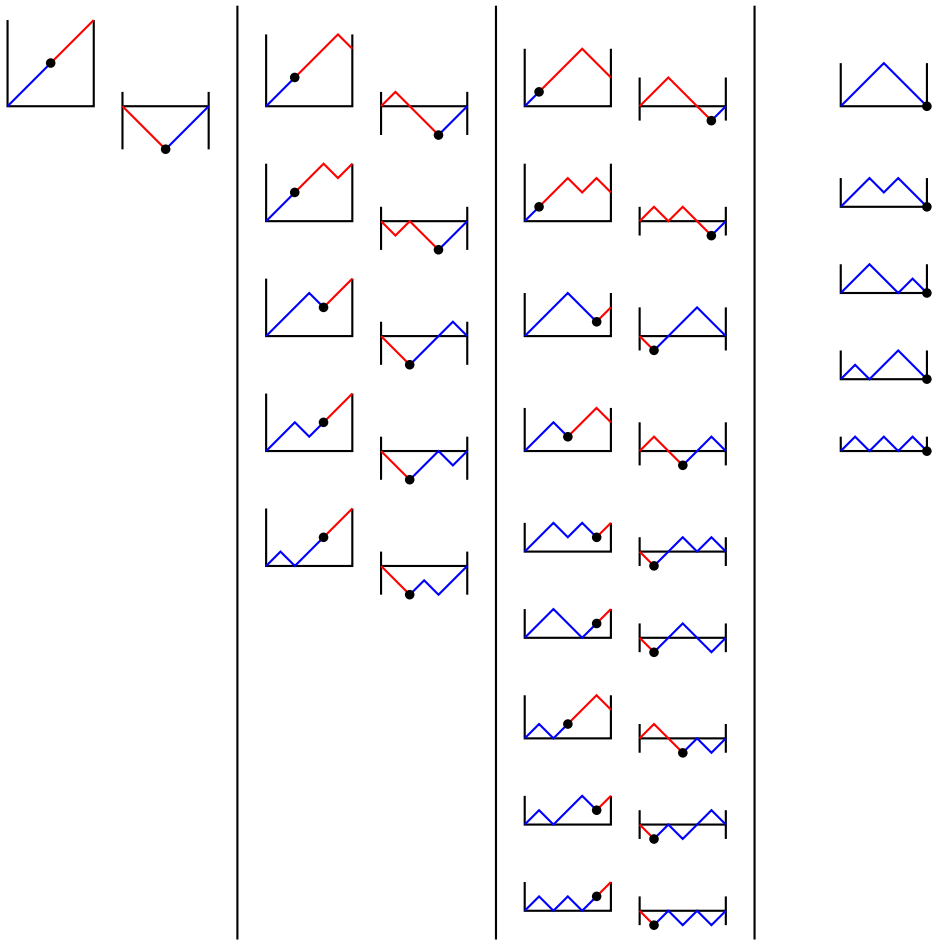

Figure 8: Graphs of walks on the integers of length 6 starting at 0, the nonnegative walks are to the left of the corresponding walks that end at 0. Walks that are a member of both classes correspond to themselves.

In case n is odd (say $n = 2m - 1$) we could give a similar proof, making a correspondence between the walks on the nonnegative integers of length n starting at 0, and the walks on the integers of length n starting at 0 and ending at 1. Alternatively, we can derive the odd case from the even case, by observing that each walk of length $2m - 1$ on the nonnegative integers starting at 0, can always be extended in exactly two ways to give such a walk of length $2m$, and moreover $2\binom{2m-1}{m} = \binom{2m}{m}$. □

Theorem 3.7.9.

There is a Kurtz test that captures every sequence with at least as many zeroes as ones in every initial segment.

Proof. Let A_n be the set of strings of length n such that every initial segment has at least as many zeroes as ones, as in the above Lemma. Note that

$$2^{-2n}\binom{2n}{n} = \frac{(2n)\cdot(2n-1)\cdot\cdots\cdot 2\cdot 1}{(2n)\cdot(2n-2)\cdot\cdots\cdot 2\cdot(2n)\cdot(2n-2)\cdot\cdots\cdot 2} = \prod_{i=1}^{n}\frac{2i-1}{2i}.$$

This gives

$$(2n+1)\left[2^{-2n}\binom{2n}{n}\right]^2 = \prod_{i=1}^{n}\frac{(2i-1)(2i+1)}{(2i)^2} < 1,$$

therefore

$$\frac{\binom{2n}{n}}{2^{2n}} < \frac{1}{\sqrt{2n+1}}$$

and by Lemma 3.7.8

$$\mu\left([\![A_{2n}]\!]\right) < \frac{1}{\sqrt{2n+1}}.$$

Hence, a suitable subsequence of $([\![A_{2n}]\!])$ gives a Kurtz test that captures exactly all the sequences with at least as many zeroes as ones in every initial segment. □

Chapter 4

Separating randomness notions

We have now introduced a fair variety of randomness notions and we have also discussed all known implications between them. We have arrived at a hierarchy of randomness notions. Still, there could be more implications that we haven't seen yet, collapsing different randomness notions into one. To prove that this does not happen, one needs to construct sequences that are random for one notion, but not for the other. This is called *separating* the notions. By separating notions we also get to understand the particular behaviour of each notion and the differences between the notions.

We will focus on the variations of computable randomness, lying in between Schnorr randomness and Martin-Löf randomness. The results from the previous chapter about these notions are summarized in Figure 9.

We will see that no additional implications hold between these notions, with one possible exception: it could be the case that partial injection randomness implies Kolmogorov-Loveland randomness, or that Kolmogorov-Loveland randomness implies Martin-Löf randomness, but not both at the same time.

We can separate notions by a direct construction or less directly, for

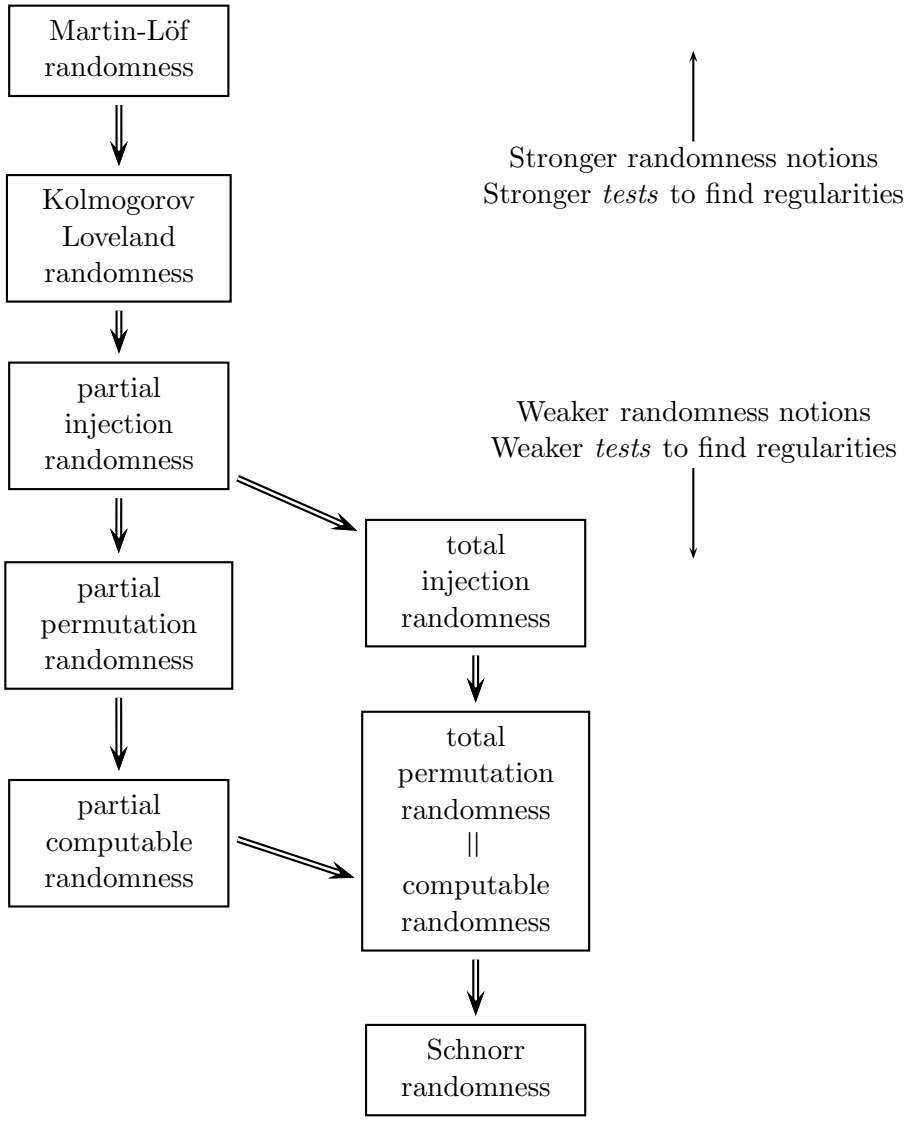

Figure 9: All known implications involving variations of computable randomness.

example by exposing different behaviour for the Kolmogorov complexity of initial segments of random sequences for the respective notions. Direct constructions are appealing because they construct concrete counterexample sequences. The hands-on constructions give a good insight into why different definitions behave differently. On the other hand, studying Kolmogorov complexity reveals properties of randomness notions that are interesting in their own right. This approach can also be preferable in situations where direct constructions would just get too messy and complicated.

I will focus on direct constructions. In Section 4.1, I will repeat the well-known construction of a sequence that is computably random but not partial computably random. This is a build-up to my own construction of a sequence that is partial computably random but not total injection random in Section 4.2. In Section 4.3, I briefly mention some other related constructions. Finally, in Section 4.4, I present separations obtained by studying Kolmogorov complexity.

4.1 A sequence that is total computably random, but not partial computably random

Theorem 4.1.1.
There exists a sequence that is total computably random, but not partial computably random.

We give a proof along the lines of [48, Theorem 7.5.7].

Proof. To construct a sequence Z that is total computably random, we need to diagonalize against all total computable betting strategies. Since we cannot effectively enumerate all total computable betting strategies, we will

enumerate all partial computable betting strategies, but ignore the partial ones when constructing the sequence Z. Also, in the sequence Z we will encode exactly which martingales in the enumeration are total. We can do this step-by-step in a non-circular way, such that a partial computable martingale can read the encoded information and use this to correctly predict certain bits of Z.

Without loss of generality we only need to consider betting strategies with initial capital 1. Let $B_0, B_1, B_2, \ldots$ be an effective enumeration of all *partial* computable martingales $2^{<\omega} \to \mathbb{Q}$ with initial capital 1. This enumeration needs to be effective in the sense that $B_k(\sigma)$ must be uniformly computable from $k \in \mathbb{N}$ and $\sigma \in 2^{<\omega}$. Such an enumeration can be obtained from an effective enumeration of all partial computable functions, letting these functions only produce outputs as long as these outputs don't contradict the function being a martingale of the required form.

We will construct Z as a concatenation of strings σ_i and digits α_i, to be defined later. The string σ_i will encode whether the martingale B_i is total. Using that information, a partial betting strategy will be able to predict the following digit α_i with certainty. The length of σ_i will be $i + 1$. Define

$$z_0 = \lambda,$$

$$z_{2i+1} = z_{2i}\sigma_i,$$

$$z_{2i+2} = z_{2i+1}\alpha_i,$$

for every $i \in \mathbb{N}$, and

$$Z = \lim_{i \to \infty} z_i.$$

Furthermore, define

$$n_i = |z_{2i+1}|.$$

We will only start diagonalizing against the martingale B_k from position n_k in the sequence onwards. Like this, at any position we only need to take into account finitely many martingales, which we call the *active martingales* at that position. The totality of the martingale B_i is be encoded before position n_i in Z, i.e. at a point where the we only need to worry about martingales $B_0, \ldots, B_{i-1}$.

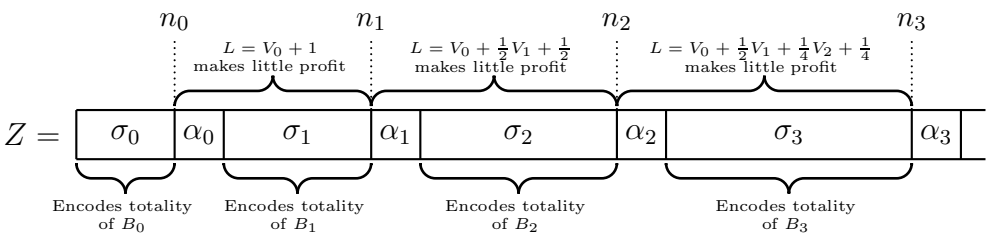

Figure 10: Illustration to the construction of a sequence Z which is total computably random but not partial computably random.

Let

$$V_k(\sigma) = \begin{cases} 1 & \text{if } |\sigma| < n_k \\ \frac{B_k(\sigma)}{B_k(\sigma\restriction_{n_k})} & \text{if } |\sigma| \geq n_k \text{ and } B_k \text{ is total} \\ 0 & \text{if } |\sigma| \geq n_k \text{ and } B_k \text{ is partial} \end{cases}$$

and define

$$L = \sum_{k \in \mathbb{N}} 2^{-k} V_k.$$

Then L is a supermartingale. If any total martingale B_k succeeds on a sequence Z, then also V_k and L will succeed on Z. So to make Z total computably random, it is sufficient to make sure that L doesn't succeed on Z.

Now we define the strings σ_i and the digits α_i inductively as follows:

- Given z_{2i}, there are at least 2 extensions of z_{2i} of length $i+1$ such that L multiplies its capital by at most $1+2^{-i}$ on these extensions. (Otherwise, the expected capital after betting on such an extension would be more than a factor

$$\frac{2^{i+1}-1}{2^{i+1}}(1+2^{-i}) = 1 + 2^{-i} - 2^{-i-1} - 2^{-2i-1} \geq 1$$

of the original capital, a contradiction with Lemma 3.4.5.) Let ρ_0 and ρ_1 be the first two such extensions (in lexicographical order). Define

$$\sigma_i = \begin{cases} \rho_0 & \text{if } B_k \text{ is total,} \\ \rho_1 & \text{otherwise.} \end{cases}$$

- Given z_{2i+1}, we choose the next digit such that L makes no profit on it:

$$\alpha_i = \begin{cases} 0 & \text{if } L(z_{2i+1}0) \leq L(z_{2i+1}), \\ 1 & \text{otherwise.} \end{cases}$$

In the limit, the capital of the supermartingale L when betting on Z is bounded by a factor

$$\prod_{i \in \mathbb{N}} 1 + 2^{-i}$$

of the original capital. Indeed, this product is finite. Since

$$1 + x < \sum_{i \in \mathbb{N}} \frac{x^i}{i!} = e^x$$

for any positive real number x, we have

$$\prod_{i=0}^{k} 1 + 2^{-i} < \prod_{i=0}^{k} e^{2^{-i}} = e^{\sum_{i=0}^{k} 2^{-i}} < e^2$$

for any $k \in \mathbb{N}$. (If one wants to avoid using analysis in this proof, one can change the definition of σ_i, thereby replacing the sequence $(1 + 2^{-i})$ with a computable sequence (a_i) that converges more quickly to 1 and for which the convergence of $\prod_{i \in \mathbb{N}} a_i$ can be proven with more elementary means. For example, one can define a_i inductively, ensuring that $\prod_{i=0}^{k} a_i$ is less than some fixed bound at every step.) Consequently, L does not succeed on Z. Therefore Z is computably random.

It remains to prove that Z is not partial computably random. Consider the following partial computable betting strategy. Read the first two digits of the sequence, i.e. the value of σ_0, without betting anything. We can compute if this is the first or the second string in lexicographical order such that L multiplies its capital by at most $1 + 1$ on this string, since L is computable on strings of length up to n_0. Accordingly, assume that B_0 is total or partial. Using this assumption, try to compute L on strings of length up to n_1. If the assumption is correct, we will succeed in this computation. Hence, we can bet our whole capital on the value of α_0 and double our money with certainty. Moreover, we can read the following three digits of the sequence, i.e. the value of σ_1, figure out if this is the first or the second extension such that L multiplies its capital by at most $1 + \frac{1}{2}$, and assume that B_1 is total or partial accordingly. And so on. If this strategy is betting on the sequence Z, then it will succeed, as all assumptions will be correct, and the capital will be doubled on every digit α_i. Therefore, Z is not partial computably random, as required. Note that strategy is certainly only partial computable, since

other sequences do not encode totality of martingales correctly, and hence the strategy when betting on such a sequence might be deluded into trying to compute undefined values of a partial computable martingale. □

4.2 A sequence that is partial computably random, but not total injection random

Having proven Theorem 4.1.1 in detail, I will allow myself to be less comprehensive in the proof of the following theorem, many aspects of which are analogous to the above proof.

Theorem 4.2.1.
There is a sequence that is partial computably random but not total injection random.

Proof. To construct a partial computably random sequence, this time we need to diagonalize against partial computable martingales as well as total computable martingales. As before, we will only start taking the martingale B_k into account from some position n_k onwards. However, in this proof, the sequence (n_k) will not be computable. It will be defined inductively later on. For now, let (n_k) be any ascending sequence of natural numbers.

To diagonalize against all total computable martingales, we define:

$$V_k^n(\sigma) = \begin{cases} 1 & \text{if } |\sigma| \leq n \\ \frac{B_k(\sigma)}{B_k(\sigma\restriction_n)} & \text{if } |\sigma| > n,\ B_k(\sigma)\downarrow \text{ and } B_k(\sigma\restriction_n) > 0 \\ 0 & \text{otherwise} \end{cases}$$

and

$$L^{(n_k)} = \sum_{k\in\mathbb{N}} 2^{-k} V_k^{n_k}.$$

$L^{(n_k)}$ is a supermartingale with the property that if any B_k succeeds on some sequence Y, then L succeeds on Y as well. Contrary to the previous proof, we will not encode any information on the totality of martingales into the sequence Z that we are constructing. (Indeed, this time we need a total betting strategy to succeed on Z, and in the previous proof it was exactly the encoded information that forced the successful betting strategy to be partial.) We simply take $Z^{(n_k)}$ to be the left-most non-ascending path on L considered as a tree, i.e. if $Z^{(n_k)}\restriction_n$ is defined, then we take

$$Z^{(n_k)}(n) = \begin{cases} 0 & \text{if } L(Z\restriction_n 0) \leq L(Z\restriction_n) \\ 1 & \text{otherwise.} \end{cases}$$

In the second case, as L is a supermartingale, we have $L(Z^{(n_k)}\restriction_n 1) \leq L(Z^{(n_k)}\restriction_n)$. Hence $\limsup_{n\to\infty} L^{(n_k)}(Z^{(n_k)}\restriction_n) \leq L^{(n_k)}(\lambda)$ and $L^{(n_k)}$ fails on $Z^{(n_k)}$. Consequently, any sequence $Z^{(n_k)}$ is partial computably random.

We claim that for a suitable choice of (n_k), the partial computably random sequence $Z^{(n_k)}$ is not total injection random. As the sequence (n_k) will not be computable, the betting strategy will have to guess the values of (n_k). Hence we introduce the following notation:

$$L_i^{\langle n_0,\ldots,n_{i-1}\rangle} = \sum_{k=0}^{i-1} 2^{-k} V_k^{n_k}$$

for any i-tuple of increasing natural numbers $n_0, \ldots, n_{i-1}$.

Also, all computations in our strategy need to halt, so we need to approximate as follows:

$$V_k^n[s](\sigma) = \begin{cases} 1 & \text{if } |\sigma| \leq n \\ \frac{B_k(\sigma)}{B_k(\sigma\restriction_n)} & \text{if } |\sigma| > n,\ B_k[s](\sigma)\downarrow, \text{ and } B_k(\sigma\restriction_n) > 0 \\ 0 & \text{otherwise,} \end{cases}$$

and

$$L_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s] = \sum_{k=0}^{i-1} 2^{-k} V_k^{n_k}[s].$$

So $L_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s]$ sums only the first i martingales, computed up to stage s, and it activates the martingales $B_0, \ldots, B_{i-1}$ at the given positions $n_0, \ldots, n_{i-1}$ respectively.

We let $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}$ be the left-most non-ascending path of $L_i^{\langle n_0,\ldots,n_{i-1}\rangle}$, and $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s]$ the left-most non-ascending path of $L_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s]$. Our betting strategy will use these $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s]$ (which are computable) as guesses for the actual Z.

More specifically, the total injection betting strategy will associate with every position a guess as to how many of the active martingales are defined along the sequence. We will approximate the martingales until this guess seems to be true, at least up to that position. The injection will then pick this position to be the subject of the next bet of the betting strategy. Moreover we can make sure that, for any fixed number of active martingales, we can bet correctly at sufficiently large positions with correct associated guesses. However, it is not computable exactly how large is sufficient, and we need to make sure that we don't activate a new martingale before we have made enough profit with the previous set of active martingales. This is why the sequence (n_i), which determines how quickly martingales are activated, cannot be computable in this construction. This in turn makes it necessary to include all possible values for n_i (for the martingales that we

presume to be active) into our guesses, in order to be able to approximate these martingales correctly.

Our total injection betting strategy works as follows.

The ordered list of positions that an injection betting strategy bets upon, may be given by a computable enumeration of an infinite subset of $\mathbb{N}$. We achieve this by uniformly assigning a computation to each $k \in \mathbb{N}$, and by betting on position k at the stage that the computation corresponding to k terminates, if ever. In particular, we will bet on $k = \langle i, \langle n_0, \ldots, n_{i-1} \rangle, l, m \rangle$ at the first stage s that

$$\left| \left\{ j \in \{0, \ldots, i-1\} \ : \ B_j[s] \left(Z_i^{\langle n_0, \ldots, n_{i-1} \rangle}[s] \restriction_{k+1} \right) \downarrow \right\} \right| = l.$$

At this point, and if i has a value that we are still interested in, we will guess that all computations involved in defining $Z_i^{\langle n_0, \ldots, n_{i-1} \rangle} \restriction_k$ that converge, have halted by stage s; hence we will bet on $Z^{(n_k)}(k) = Z_i^{\langle n_0, \ldots, n_{i-1} \rangle}[s](k)$. Under certain conditions, this guess is guaranteed to be correct. In particular the following lemma holds:

Lemma 4.2.2.

Suppose that

(a) $l = \left| \left\{ j \in \{0, \ldots, i-1\} \ : \ B_j \text{ is defined along } Z_i^{\langle n_0, \ldots, n_{i-1} \rangle} \right\} \right|$, *and*

(b) m *is sufficiently large.*

Let $k = \langle i, \langle n_0, \ldots, n_{i-1} \rangle, l, m \rangle$. *Then there is a stage* s *such that*

$$\left| \left\{ j \in \{0, \ldots, i-1\} \ : \ B_j[s] \left(Z_i^{\langle n_0, \ldots, n_{i-1} \rangle}[s] \restriction_{k+1} \right) \downarrow \right\} \right| = l. \tag{23}$$

Moreover, at this stage we have

$$Z_i^{\langle n_0, \ldots, n_{i-1} \rangle}(k) = Z_i^{\langle n_0, \ldots, n_{i-1} \rangle}[s](k).$$

Proof. We abbreviate $Z_i = Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}$.

There are only finitely many $n \in \mathbb{N}$ such that $B_j(Z_i{\restriction}_n)\downarrow$ for some $j \in \{0,\ldots,i-1\}$ such that B_j is *not* defined along Z_i. Let N be the maximal such n. Let s_0 be the first stage such that

$$B_j[s](Z_i{\restriction}_n)\downarrow \text{ if and only if } B_j(Z_i{\restriction}_n)\downarrow$$

for all $j \in \{0,\ldots,i-1\}$ and all $n \leq N+1$.

Given *(a)*, (23) will hold for s large enough. But note that the larger we take m, the larger k is, and the longer it will take for (23) to hold. So we can take m large enough to have $k > N$ and $s \geq s_0$.

By choice of N, s_0 and m, we have

$$Z_i{\restriction}_N = Z_i[s]{\restriction}_N,$$

and $B_j(Z_i{\restriction}_N)\uparrow$ for all $j \in \{0,\ldots,i-1\}$ such that B_j is *not* defined along Z_i. Hence, when (23) holds, we must have

$$\begin{aligned}\big\{j \in \{0,\ldots,i-1\} :\ & B_j[s]\left(Z_i[s]{\restriction}_{k+1}\right)\downarrow\big\}\\ &\subseteq \{j \in \{0,\ldots,i-1\} :\ B_j \text{ is defined along } Z_i\}\end{aligned}$$

and by *(a)* this is actually an equality. This means that all computations involved in defining $Z_i{\restriction}_{k+1}$ have halted by stage s, so the guess

$$Z_i(k) = Z_i[s](k)$$

is correct. □

We are now ready to define the sequence (n_k) and the total injection

strategy that will succeed on $Z^{(n_k)}$. We have already defined the computable injection above. Now we partition the initial capital; to every natural number $j = \langle i, \langle n_0, \ldots, n_{i-1}\rangle, l\rangle$ we assign a fraction 2^{-j-1} of our starting capital. When we are asked to bet on $k = \langle i, \langle n_0, \ldots, n_{i-1}\rangle, l, m\rangle$, we will only use the capital assigned to the number $\langle i, \langle n_0, \ldots, n_{i-1}\rangle, l\rangle$. In particular, if we are asked to bet on this position k at stage s, then we will put $\frac{3}{4}$ of this capital on the outcome $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}[s](k)$ and $\frac{1}{4}$ of this capital on the other outcome. Once the capital assigned to some $\langle i, \langle n_0, \ldots, n_{i-1}\rangle, l\rangle$ exceeds 1, we start betting evenly on positions with this value of i, and we say that the *substrategy for i has succeeded.*

Remark 4.2.3. *The substrategy for i is certain to succeed when betting on $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}$. Indeed, by Lemma 4.2.2, when l has the correct value and m is big enough, then at some point we will bet on position $k = \langle i, \langle n_0, \ldots, n_{i-1}\rangle, l, m\rangle$ and this bet is guaranteed to be successful, i.e. to increase the capital assigned to $\langle i, \langle n_0, \ldots, n_{i-1}\rangle, l\rangle$ with a factor $\frac{3}{2}$. So the capital assigned to $\langle i, \langle n_0, \ldots, n_{i-1}\rangle, l\rangle$ will exceed 1 if we go on for long enough.*

Remark 4.2.4. *If the substrategy for i succeeds when betting on $Z_i^{\langle n_0,\ldots,n_{i-1}\rangle}$, and the highest position that the strategy has bet on before succeeding is position k, then the substrategy will run exactly the same, and hence also succeed at the same point, on any sequence Y with $Y\restriction_{k+1} = Z\restriction_{k+1}$. In particular, if*

$$k < n_i < n_{i+1} < n_{i+2} < \ldots,$$

then the substrategy for i will run exactly the same on $Z_j^{\langle n_0,\ldots,n_{j-1}\rangle}$ for any $j \geq i$, and also on $Z^{(n_k)}$.

We now recursively define n_k by letting $n_0 = 0$ and taking

$$n_i = 1 + \begin{pmatrix}\text{highest position that the strategy has bet on}\\ \text{after the substrategies for } 0, \ldots, i-1 \text{ have suc-}\\ \text{ceeded when betting on } Z_i^{\langle n_0, \ldots, n_{i-1}\rangle}\end{pmatrix}.$$

By Remark 4.2.3, these substrategies indeed all succeed, so the sequence is well-defined. Moreover, by Remark 4.2.4, the substrategies all succeed on $Z^{(n_k)}$, as well. So the total injection betting strategy succeeds on $Z^{(n_k)}$, as there are infinitely many substrategies that, with disjoint parts of the initial capital, all generate one unit of money. Therefore $Z^{(n_k)}$ is a sequence that is partial computably random, but not total injection random, concluding the proof of Theorem 4.2.1. □

4.3 Other constructions

Nies, Stephan and Terwijn

Nies, Stephan and Terwijn [49] proved that sequences as constructed in Theorem 4.1.1 (computable random but not partial computably random) can be found exactly in the high Turing degrees. A similar fact holds for sequences that are Schnorr random but not computably random.

Theorem 4.3.1 (Nies, Stephan and Terwijn [49], see also [16, 8.11.6] and [48, 3.5.13, 7.5.9 and 7.5.10])**.**
The following are equivalent for a Turing degree $\mathcal{A}$:

- *$\mathcal{A}$ is a high Turing degree;*
- *$\mathcal{A}$ contains a sequence that is computably random but not partial computably random;*

- *$\mathcal{A}$ contains a sequence that is Schnorr random but not computably random.*

One direction of this result follows from the fact that in non-high Turing degrees, the whole hierarchy of randomness notions between Schnorr randomness and Martin-Löf randomness collapses. This means: every non-high Schnorr random is also Martin-Löf random. Indeed, if a Turing degree $\mathcal{A}$ is not high, then for every total function $f \in \mathcal{A}$, there is a computable function that is not dominated by f. In particular, if some sequence $Z \in \mathcal{A}$ fails a Martin-Löf test (U_i), then the function f mapping $i \in \mathbb{N}$ to the first stage s such that $Z \in U_i[s]$ is total and computable in $\mathcal{A}$. Hence there is a computable function g that is not dominated by f, i.e. $g(i) > f(i)$ for infinitely many values of i. Now $(U_i[g(i)])$ is a Schnorr test (even a Kurtz test) such that $Z \in U_i[g(i)]$ for infinitely many values of i. By Remark 3.2.5, Z is not Schnorr random, as required.

Kastermans and Lempp

The main open problem about separating randomness notions, is the question whether Martin-Löf randomness is equivalent to or strictly stronger than Kolmogorov-Loveland randomness. The result that comes closest to a solution to this question was obtained by Kastermans and Lempp [24]. They constructed a sequence that is partial injection random, but not Martin-Löf random.

Theorem 4.3.2 (Kastermans and Lempp [24], see also [16, 8.11.6])**.** *There is a sequence that is partial injection random, but not Martin-Löf random.*

Before this result was published, Miller and Nies [44] suggested that a sep-

aration of Martin-Löf randomness from permutation or injection randomness might provide a stepping stone towards a separation of Martin-Löf randomness and Kolmogorov-Loveland randomness. Kastermans and Lempp have now obtained the weaker separation, but so far it has not helped towards solving the bigger question.

4.4 Separations by initial segment complexity

A less direct but very fruitful way to compare the strength of different randomness notions, is to investigate how low the initial segments complexities of random sequences can get. As we have seen in Theorems 3.3.1 and 3.3.3, for Martin-Löf randomness there is a very clear distinction between the initial segment complexities of random and nonrandom sequences. For other notions of randomness however, this is usually not the case.

Random sequences with low complexity

Computable randomness

In the proof of Theorem 4.1.1 we constructed a computably random sequence. The only incomputable step in this construction was the question "is B_k total or partial?". In particular, if we are given n, then we can compute the initial segment of length n of the computably random sequence with just one bit of information for every martingale that is active at position n; that bit encoding the information "is B_k total?" for every active martingale B_k. In the proof, the number of active martingales grows at a fixed rate, determined by the sequence (n_i). However, we can replace this function

by any computable function that grows more quickly, thereby making the number of active martingales grow as slowly as any computable order. This gives the following:

Theorem 4.4.1.
For any computable order h, there exists a computably random sequence Z such that

$$C(Z\upharpoonright_n|n) < h(n) + O(1).$$

Partial computable randomness

In the proof of Theorem 4.2.1 we constructed a partial computably random sequence by a similar construction. Here we not only need to know whether an active martingale is total or partial, but in case it is partial, we also need to know at what point in the construction that we come across this partiality. In particular: to compute (given n) the initial segment of length n of the partial computably random sequence, we need to know for every active martingale at exactly which position in $\{0, 1, \ldots, n\}$ it becomes undefined along the sequence, if any. So we need $\log(n)$ bits of information for every active martingale. This gives the following:

Theorem 4.4.2.
For any computable order h, there exists a partial computably random sequence Z such that

$$C(Z\upharpoonright_n|n) < h(n)\log(n) + O(1).$$

Total injection randomness

Using a similar argument, Bienvenu, Hölzl, Kräling and Merkle [3] constructed a total injection random sequence with low initial segment complexity.

Theorem 4.4.3 (Bienvenu, Hölzl, Kräling and Merkle [3])**.**
For any computable order h, there exists a total injection random sequence Z such that

$$C(Z\upharpoonright_n|n) < h(n) + \log(n) + O(1).$$

Partial permutation randomness

The construction becomes considerably more complicated when the martingales involved are both partial and nonmonotonic. As we will see later, there are no partial permutation random sequences where *every* initial segment has a low complexity. However, Bienvenu, Hölzl, Kräling and Merkle [3] still managed to construct a partial permutation random sequence with *infinitely many* initial segments of low complexity.

Theorem 4.4.4 (Bienvenu, Hölzl, Kräling and Merkle [3])**.**
For any computable order h and any infinite computable set $S \subseteq \mathbb{N}$, there exists a partial permutation random sequence Z such that

$$C(Z\upharpoonright_n|n) < h(n)$$

for infinitely many $n \in S$.

Lower bounds for the complexity of random sequences

Computable randomness

In Theorem 4.4.1 we already discovered that computably random sequences can have very low initial segment complexities. Indeed, suppose we require slightly lower initial segment complexities by removing the computable order from the condition of Theorem 4.4.1. Then the condition suddenly only holds for just computable sequences.

Theorem 4.4.5 (Loveland [37], see also [16, 3.4.1])**.**
Z is computable if and only if

$$C(Z\upharpoonright_n|n) < O(1).$$

This leads to the following question:

Question 4.4.6.
Does there exist a non-trivial lower bound for the initial segment complexities of computably random sequences.

Non-trivial here means that some incomputable sequences should have complexities that are below the bound. Theorem 4.4.1 suggests that such a bound will involve incomputable functions that grow more slowly than any computable order.

Partial computable randomness

If we remove the computable order from the condition in Theorem 4.4.2, then no more partial computably random sequences satisfy the condition. At least in this case, the lower bound is non-trivial.

Theorem 4.4.7 (Merkle [42])**.**
If

$$C(Z\upharpoonright_n|n) < O(\log(n)),$$

then Z is not partial computably random.

Note that we might as well use unconditional complexity $C(Z\upharpoonright_n)$ in this Theorem, since there is only a $O(\log(n))$ difference between $C(Z\upharpoonright_n|n)$ and $C(Z\upharpoonright_n)$ anyway.

Total injection randomness

For total injection randomness we have a more complicated lower bound.

Theorem 4.4.8 (Bienvenu, Hölzl, Kräling and Merkle [3])**.**
If (n_k) is a computable sequence of natural numbers such that $n_{k+1} \geq 2n_k$ for all k, such that

$$C(Z\upharpoonright_{n_k}|k) < \log(n_k) + 3\log(\log(n_k)),$$

then Z is not total injection random.

Partial permutation randomness

Andrei Muchnik proved that all partial permutation random sequences have initial segment complexities that are relatively close to their lengths. Comparing this with Theorem 4.4.2, we see that partial permutation randomness is significantly stronger than (monotone) partial computable randomness. This contrasts with the total case, where monotone and permutation randomness are equivalent.

Theorem 4.4.9 (Andrei Muchnik [46, 9.1])**.**
If there is a computable order h such that

$$K(Z\upharpoonright_n) < n - h(n) - O(1),$$

then Z is not partial permutation random.

The article [46] in fact only states this theorem for partial injection randomness, a weaker result, but the proof actually provides a permutation betting strategy that proves the stronger statement.

Separations using complexity

We now know fairly well how low the initial segment complexity can be for different notions of computable randomness. Comparing them, we get the following conclusions, all taken from [3].

Theorem 4.4.10.
There exists a sequence that is partial computably random but not partial permutation random.

Proof. By Theorem 4.4.2 there exists a partial computably random sequence Z with

$$C(Z\upharpoonright_n|n) < \log(n)\log(n) + O(1).$$

Therefore

$$K(Z\upharpoonright_n) < C(Z\upharpoonright_n|n) + 2\log(n) + O(1) \leq \log(n)\log(n) + 2\log(n) + O(1)$$

so by Theorem 4.4.9 with h being any computable order that grows more slowly than $n - \log(n)\log(n) - 2\log(n)$, we know that Z is not partial permutation random. □

Theorem 4.4.11.

There exists a sequence that is total injection random but not partial computably random.

Proof. By taking $h(n) = \log(n)$ in Theorem 4.4.3, there exists a total injection random sequence Z with

$$C(Z\upharpoonright_n|n) < 2\log(n) + O(1).$$

Therefore by Theorem 4.4.7 we know that Z is not partial computably random. □

Theorem 4.4.12.

There exists a sequence that is partial permutation random but not total injection random.

Proof. Pick a computable sequence (n_k) according to the conditions of Theorem 4.4.8 with $K(k) \leq \log(n_k)$ for all k. Let $S = \{n_k : k \in \mathbb{N}\}$. By Theorem 4.4.4 there exists a partial permutation random sequence Z with

$$C(Z\upharpoonright_n|n) < \log(\log(n))$$

for infinitely many $n \in S$. Consequently, for infinitely many $k \in \mathbb{N}$ we have

$$\begin{aligned} C(Z\upharpoonright_{n_k}|k) &\leq C(Z\upharpoonright_{n_k}) + O(1) \\ &\leq C(Z\upharpoonright_{n_k}|n_k) + K(n_k) + O(1) \\ &\leq C(Z\upharpoonright_{n_k}|n_k) + K(k) + O(1) \\ &\leq \log(\log(n_k)) + \log(n_k) + O(1). \end{aligned}$$

By Theorem 4.4.8, Z is not total injection random. □

The last theorem implies my Theorem 4.2.1 and was also published [3] before my result. Still, my direct construction has its own appeal. Moreover, the techniques used in my construction are more likely to be of help with the open problem of Kolmogorov-Loveland randomness versus Martin-Löf randomness, as no useful bounds for the initial segment complexity of Kolmogorov-Loveland random sequences are known.

Chapter 5

Axioms about complexity

Chaitin was the first to realize that stating that certain strings are incompressible ("$C(\sigma) > |\sigma|$") provides statements that are true but not provable in axiomatic theories like Peano Arithmetic (PA). This provides an elegant proof of Gödel's first incompleteness theorem [21] using Kolmogorov complexity.

As we have seen, requiring that initial segments of a sequence have a high complexity also gives characterizations of certain randomness notions. In particular Martin-Löf randomness has a simple definition in this way (Theorem 3.3.1). This raises the question of the proof-theoretic power of the theory which expresses that a certain sequence is Martin-Löf random.

All axioms of the form "$C(\sigma) > n$", formalized in PA or other appropriate axiomatic theories, are Π^0_1 sentences. (One could also call them *universal sentences*, since they only have universal quantifiers, but we avoid this because we use the term *universal* already in a different sense in *universal machine*.) We will concentrate on this class of Π^0_1 sentences in our investigations. Every Π^0_1 sentence can be interpreted as saying that a certain computation does not halt. Conversely, the non-halting of any computation can be formalized

as a Π^0_1 sentence. Therefore, proving all true (in the standard model) Π^0_1 sentences is in a sense the proof-theoretic equivalent of solving the halting problem. We will show that the theory of all true statements of the form "$C(\sigma) > n$" proves all true Π^0_1 sentences, thereby obtaining a proof-theoretic version of the Turing completeness of C (Theorem 2.4.2). The results will get more subtle when we consider axiomatic theories that express that a certain sequence is Martin-Löf random.

This chapter originates as joint work with Laurent Bienvenu, Andrei Romashchenko, Alexander Shen and Antoine Taveneaux, which is due to be published in the Annals of Pure and Applied Logic [4]. This chapter focusses only on those aspects of the article where randomness is directly involved, which was also my main involvement in the research.

I suppose in this chapter that the reader is familiar with the basics of proof theory. In particular, a good intuition about formalizing mathematical statements into PA and about provability in PA is required. For background reading, see e.g. Mendelson's book [40].

5.1 Chaitin's result

We will now consider the proof-theoretic power of statements about Kolmogorov complexity in axiomatic theories like PA. Remember that Kolmogorov complexity is not computable. This result has a counterpart in proof theory: Gregory Chaitin observed in 1974 that statements of the form "$K(\sigma) > n$" can only be provable for a finite number of values of n.

Exactly what axioms we use, is not essential here. We could work in PA or in stronger theories. Essential are the following properties of PA:

- PA can prove all true statements that use only bounded quantifiers,
- "$\phi_{e,s}(x) = y$" is definable in PA as a relation in e, s, x and y, using only bounded quantifiers.

Consequently,

$$\text{“}\phi_e(x) = y\text{”} = \text{“}\exists s\colon \phi_{e,s}(x) = y\text{”}$$

and

$$\text{“}\phi_e(x)\downarrow\text{”} = \text{“}\exists y\colon \phi_e(x) = y\text{”}$$

are Σ^0_1 formulas which can be proven when they are true.

$$\text{“}\phi_e(x)\uparrow\text{”} = \text{“}\neg\phi_e(x)\downarrow\text{”}$$

is a Π^0_1 formula.

$$\text{“}K(\sigma) < n\text{”} = \text{“}\exists \tau\colon\ (|\tau| < n \text{ and } \mathbb{U}(\tau) = \sigma)\text{”}$$

is a Σ^0_1 formula, so all upper bounds for the complexity of any string are provable. Consequently,

$$\text{“}K(\sigma) > n\text{”}$$

is a Π^0_1 formula. The same holds for plain complexity C instead of prefix-free complexity.

We can now go back to Chaitin's theorem.

Theorem 5.1.1.

There is a bound $N \in \mathbb{N}$ such that any provable sentence of the form "$K(\sigma) > n$" has $n < N$.

Chaitin proved this theorem in 1974 [10, Theorem 4.1], but the idea goes back to 1971 [9].

Proof. Suppose for contradiction that for all $n \in \mathbb{N}$ there is a provable sentence "$K(\sigma) > n$". (Note that if "$K(\sigma) > n$" is provable and $n > m$, then also "$K(\sigma) > m$" is provable.) Let "$K(\sigma_n) > n$" be the first such sentence in a fixed enumeration of all theorems. Using this enumeration, we can compute σ_n from n, so $K(\sigma_n) < O(\log n)$. For large n, this is in contradiction with $K(\sigma_n) > n$. □

Note that this can be seen as a proof of Gödel's first incompleteness theorem [21] ("there are true statements that are not provable in PA") using Kolmogorov complexity.

5.2 Machines that are *provably* universal

We can give an alternative proof of Chaitin's result, using a lemma about the provability of equivalence of Turing machines.

Lemma 5.2.1.
There exists a machine M such that for any other machine N, it is not provable that $M \not\equiv N$.

Remark 5.2.2. *Such a machine M cannot halt on any input, but this is not provable, even though the* existence *of such a machine is provable.*

Proof of Lemma 5.2.1. Suppose for contradiction that for any e there exists and n such that "$\phi_e \neq \phi_n$" is provable. Let $f(e)$ be the first n for which "$\phi_e \neq \phi_n$" appears in a fixed enumeration of all theorems. Then f is a total

computable function such that $\phi_e \neq \phi_{f(e)}$ for all e, contradicting the Fixed Point Theorem (Lemma 2.3.1). □

Alternative proof of Theorem 5.1.1. Let M be a machine such as in the Lemma. Let c be the coding constant for M, i.e.

$$K(\sigma) < K_M(\sigma) + c \tag{24}$$

for all strings σ. Suppose for contradiction that "$K(\sigma) > n$" is provable for some σ and some $n > c$. Then also

$$\text{“}M(\tau) \neq \sigma\text{”}$$

is provable for all τ of length less than $n - c > 0$. So it is provable that M is not equivalent to e.g. the machine that maps the empty string to σ and diverges on all other inputs. This contradicts the choice of M. □

There is however one hidden problem with this proof: (24) needs to be provable. In other words, our axiomatic theory needs to be able to prove information about the coding constants of our universal machine. This is not guaranteed. There exist universal machines that are not *provably* universal.

Theorem 5.2.3.

There exists a machine M which is equivalent to the standard universal machine $\mathbb{U}$ (constructed in Section 2.4), but for which it is not provable that it halts on infinitely many inputs.

Proof. Let "$\forall n\colon \psi(n)$" be some Π^0_1 sentence which is true but not provable

(such as the consistency of our theory). Define

$$M(\sigma) = \begin{cases} \mathbb{U}(\sigma) & \text{if } \forall n < |\sigma| \colon \psi(n) \\ \uparrow & \text{otherwise} \end{cases}.$$

From "$\exists N \colon \neg\psi(N)$" we would be able to prove

$$\text{“}\exists N \forall \sigma \colon (|\sigma| > N \implies M(\sigma)\uparrow)\text{”},$$

and hence that M only halts on finitely many inputs. By taking the contrapositive: if it were provable that M halts on infinitely many inputs, then "$\forall n \colon \psi(n)$" would be provable, contradicting our choice of ψ. □

From now on, we will suppose that we don't have such a weird universal machine. We assume that PA (or whichever base theory is used) has some understanding of the workings of the universal machine, and in particular that PA can prove inequalities like (24). This is fine for the standard universal machine $\mathbb{U}$.

There is a similar issue with the computational process that we use. Any of the usual methods (such as Turing machines) will do fine. It is however possible to consider a machines that in parallel to executing their program, try to prove the inconsistency of PA, and will go into an infinite loop if they find such a proof. We know that these machines will behave exactly like ordinary Turing machines, but PA does not, since PA cannot prove its own consistency. PA cannot even prove that any of these machines compute a total function. We will assume that we are not dealing with such a strange computational model. We assume that PA can prove basic facts about our computations, for example that the machine which computes addition actually computes a total function.

5.3 Axioms about strings of high complexity

We now consider the strength of theories consisting of axioms stating that some strings have high complexity (in addition to the axioms of the base theory, e.g. PA, which we always implicitly assume). By Chaitin's result the true axioms of the form

$$\text{“}C(\sigma) > n\text{”} \tag{25}$$

where n is larger than the N in Theorem 5.1.1, give a theory that is strictly stronger than PA. Exactly how strong can this theory get? Since all axioms (25) are Π^0_1 sentences, the theory can at most get as strong as the theory consisting of all true Π^0_1 sentences.

A Π^0_1 sentence "$\forall n \phi(n)$" states that the computation that tries to find the least n such that $\neg\phi(n)$ never halts. Conversely, the statement that a particular computation doesn't halt, is always Π^0_1. Hence, *proving all true* Π^0_1 *sentences* seems to be the proof-theoretic equivalent of *solving the halting problem* in computability theory.

The true axioms of the form "$C(\sigma) > n$" allow PA to prove exact values of the complexity function C. Indeed, the negations "$C(\sigma) \leq n$" are Σ^0_1 formulas and hence automatically provable when true. Since Kolmogorov complexity is Turing complete (Theorem 2.4.2), we can expect that the true axioms of the form "$C(\sigma) > n$" will be strong enough to prove all true Π^0_1 sentences. This is indeed the case. In fact, a much weaker condition is sufficient. It is possible to prove all true Π^0_1 sentences with an axiom

$$\text{“}C(\sigma_n) > n - c\text{”}$$

for just one carefully chosen string σ_n of length n, for infinitely many n.

Theorem 5.3.1.

Fix some constant $c \geq 0$. For each n, let σ_n be the lexicographically first string of length n such that $C(\sigma_n) \geq n - c$. Any theory T consisting of infinitely many axioms of the form

$$\text{“}C(\sigma_n) > n - c\text{”}$$

can prove all true Π_1^0 sentences.

The proof is very similar to the proof of Theorem 2.4.2.

Proof. Let “$\forall n\colon \psi(n)$” be a Π_1^0 sentence. Consider the machine M that for successive values of n checks $\psi(n)$, and halts if it finds an n such that $\neg\psi(n)$. Under the assumptions of Section 5.2, PA can prove that

$$\text{“}\forall n\colon \psi(n)\text{”}$$

and

$$\text{“}M \text{ does not halt”}$$

are equivalent. So it is sufficient to prove that the theory T can prove the non-termination of every non-halting program.

Remember from Chapter 2 that C^s is the time-bounded Kolmogorov complexity. $C^s(\sigma)$ is the length of the shortest description that makes the universal machine output σ in less than s steps. The functions C^s are computable and approximate C from above. So we can define s_n as the least s such that $C^s(\tau) < n - c$ for every τ of length n that is lexicographically before σ_n. We prove that from a program P, we can compute a number n such that the computation P either terminates in less than s_n steps, or does not terminate at all.

Given a terminating program P and a number n, let $s(P)$ be the number of steps that P takes to terminate. Let σ be the lexicographically first string of length n such that $C^{s(P)}(\sigma) \geq n - c$. If P does not halt within s_n steps, then we know that $\sigma = \sigma_n$. On the other hand, for every P that terminates we get some string σ of length n with

$$\begin{aligned} C(\sigma) &< C(P, n) + O(1) \\ &< K(n) + C(P) + O(1) \\ &< C(P) + O(\log n). \end{aligned}$$

If n is large enough compared to $C(P)$, this gives $C(\sigma) < n - c$. For such an n, we know that σ is different from σ_n, so P must have halted within s_n steps. Consequently, if a program P terminates at all, then it must do so in less than s_n steps.

This argument can be formalized in PA as well. Having "$C(\sigma_n) \geq |\sigma_n| - c$" as an axiom, it is provable that σ_n is the first string σ of length n with $C(\sigma) \geq |\sigma| - c$, as we can find shorter descriptions for all preceding strings of length n. Then, given n, it is provable that the value of s_n satisfies its definition. Finally, given P and taking n suitably large, it is provable (doing the above proof inside PA) that P either terminates in s_n steps or does not terminate at all, as required. □

The same proof works for prefix-free complexity instead of plain complexity. Therefore we can restate the theorem with K instead of C.

Theorem 5.3.2.

Fix some constant $c \geq 0$. For each n, let σ_n be the lexicographically first string of length n such that $K(\sigma_n) \geq n - c$. Any theory T consisting of

infinitely many axioms of the form

$$\text{“}K(\sigma_n) > n - c\text{”}$$

can prove all true Π^0_1 *sentences.*

Proof. The proof of Theorem 5.3.1 works with plain complexity C replaced everywhere by prefix-free complexity K. □

Can we make all the strings σ_n for which we include the axioms, initial segments of the same sequence? Here the answer depends on which complexity we use.

For prefix-free complexity, we have the following result.

Theorem 5.3.3.

Fix some constant $c \geq 0$. *There exists a sequence* Z *and an infinite set* $A \subseteq \mathbb{N}$ *such that the theory consisting of the axioms*

$$\text{“}K(Z\upharpoonright_n) \geq n - c\text{”}$$

for all $n \in A$ *is consistent and proves all true* Π^0_1 *sentences.*

Proof. We order all strings by length and then lexicographically. That is, $\sigma < \tau$ if and only if $|\sigma| < |\tau|$, or $|\sigma| = |\tau|$ and σ is lexicographically before τ.

We construct Z as follows: let τ_0 be some string with $K(\tau_0) < |\tau_0| - c$. Inductively, let σ_n be the first (for the above order) string σ that extends τ_n with $K(\sigma) \geq |\sigma| - c$, and let τ_{n+1} be some string extending σ_n such that

$$K(\tau_{n+1}) < |\tau_{n+1}| - (n+1) - c.$$

Note that σ_n must exist, as by Corollary 3.3.3 every string can be extended

to a Martin-Löf sequence Y for which

$$\lim_{n\to\infty} (K(Y\restriction_n) - n) = \infty.$$

Let $Z = \lim_{n\to\infty} \sigma_n$. Consider the axioms

$$\text{“}K(\sigma_n) \geq |\sigma_n| - c\text{”}$$

for all $n \in \mathbb{N}$. (That is: $A = \{|\sigma_n| : n \in \mathbb{N}\}$ in the statement of the theorem.) We claim that this theory can prove all true Π^0_1 sentences. The proof is similar to the proof of Theorem 5.3.1.

As in Theorem 5.3.1, it is sufficient to prove that for every program P that does not terminate, our theory proves this non-termination.

Define s_n to be the first s such that $K^s(\sigma) < |\sigma| - c$ for all strings σ that extend τ_n and come before σ_n in our order. We prove that from a program P, we can compute a number n such that the computation P either terminates in less than s_n steps, or does not terminate at all.

Given a terminating program P and a number n, let $s(P)$ be the number of steps that P takes before halting. Let σ be the first string that extends τ_n with $K^{s(P)}(\sigma) \geq |\sigma| - c$. If P does not halt within s_n steps, then we know that $\sigma = \sigma_n$. On the other hand, for every P that terminates we get some string σ extending τ_n with $K(\sigma) < K(P) + K(\tau_n) + O(1)$. By definition, τ_n has a low complexity. Consequently

$$\begin{aligned} K(\sigma) &< K(P) + |\tau_n| - n - c + O(1) \\ &< K(P) + |\sigma| - n - c + O(1). \end{aligned}$$

Given the program P, we can find an n that is large enough such that

$$K(P) - n + O(1)$$

is negative. For such an n, we know that σ is different from σ_n, so P must have halted within s_n steps. Consequently, if a program P terminates at all, then it must do so in less than s_n steps.

As in the proof of Theorem 5.3.1, this reasoning can be formalized in PA. Having "$K(\sigma_n) \geq |\sigma_n| - c$" as an axiom, it is provable that σ_n is the first string extending τ_n such that $K(\sigma_n) \geq |\sigma_n| - c$. Then, given τ_n, it is provable that the value of s_n satisfies its definition. Finally, given P and taking τ_n for n suitably large, it is provable (doing the above proof inside PA) that P either terminates in s_n steps or does not terminate at all, as required. □

Remark that the sequence Z that we constructed in the proof, has arbitrarily large *complexity dips* in between the initial segments σ_n with complexity at least $|\sigma_n| - c$. Hence Z is not Martin-Löf random. This is essential by Theorem 5.4.1. Indeed, even if we choose a Martin-Löf random Z and a constant c small enough such that $K(Z\upharpoonright_n) > n - c$ is not true for all n, it still must be true for all but finitely many n. In this case the proof of Theorem 5.4.1 still works to show that the theory consisting of all true axioms "$K(Z\upharpoonright_n) > n - c$" does not prove all true Π^0_1 statements.

For plain complexity, the proof of Theorem 5.3.3 does not work. The reason is that not every string can be extended to a string with high plain complexity.

Question 5.3.4.

Does there exist a sequence Z and a theory T consisting of infinitely

many axioms of the form

$$\text{“}C(Z{\restriction_n}) > n - c\text{”}$$

such that T is consistent and proves all true Π^0_1 sentences?

Note that if there does exists such a sequence Z, then Z must be 2-random. This makes the question quite different from Theorem 5.3.3, as the sequence constructed in the proof of the theorem was necessarily non-random, whereas Question 5.3.4 relates to the properties of random sequences.

Moreover, remark that, although there are no Turing-complete 2-random sequences, some corresponding theory T might still be Turing complete.

5.4 Axioms expressing Martin-Löf randomness

What happens if we add information about the complexity of all initial segments of an entire sequence to Peano Arithmetic? In the first place, we are interested in expressing that some sequence Z is Martin-Löf random. This is equivalent with the fact that there exists some c such that

$$K(Z{\restriction_n}) > n - c$$

for all n. We cannot express this with just one axiom, but we could consider the theory $MLR_c(Z)$ that consists of infinitely many axioms, namely the axiom

$$\text{“}K(Z{\restriction_n}) > n - c\text{”}$$

for each n. This theory is consistent if and only if Z is indeed Martin-Löf random with the given constant c.

Our main result about the theories $MLR_c(Z)$ is the following.

Theorem 5.4.1.
If $MLR_c(Z)$ is consistent, then $MLR_c(Z)$ does not prove all true Π^0_1 sentences.

Note that this contrasts with the fact that there are Turing-complete Martin-Löf random sequences, like Ω. Even for Ω however, the consistent theories $MLR_c(\Omega)$ do not prove all true Π^0_1 statements.

The proof of Theorem 5.4.1 is interesting enough to merit a thorough introduction.

If $Z \not\geq_T \mathbf{0}'$, then the theorem follows immediately from the fact that the theory $MLR_c(Z)$ is not Turing complete.

If $Z \geq_T \mathbf{0}'$, then we use an idea due to Antoine Taveneaux. Note that the theory $MLR_c(Z)$ will be Turing complete, as we can compute Z from $MLR_c(Z)$. The much stronger set of axioms consisting of *all* true sentences of the form "$K(\sigma) > |\sigma| - c$" is Turing complete as well, yet not because we can compute Z from it, but because we are given so much information about the Kolmogorov complexity function. Our aim is to find a midpoint in between. We hope to add axioms about the complexities of more strings than just the initial segments of Z, such that the actual bits of Z become obscured. But we don't want too many axioms, to avoid that our theory becomes Turing complete for different reasons. This would give us a theory that is stronger than $MLR_c(Z)$ in the proof-theoretic sense, but not Turing complete. This theory doesn't prove all true Π^0_1 statements, just like in the case $Z \not\geq_T \mathbf{0}'$, so certainly the weaker theory $MLR_c(Z)$ doesn't either.

Now how do we pick these extra axioms? We will find a Turing degree $\mathcal{A}$ that doesn't derandomize Z (i.e. Z is still Martin-Löf random relative to $\mathcal{A}$) and computes a lower bound $\widetilde{K}$ for K, which is however (up to an additive constant) an upper bound for the relativized complexity $K^{\mathcal{A}}$. Then we add axioms stating that σ has high complexity for all strings σ that still have high values for the lower bound $\widetilde{K}$. In particular, this will include all initial segments of Z, since $\mathcal{A}$ doesn't derandomize Z. The extended theory is $\mathcal{A}$-computable. Hence $\mathcal{A}$ must be Turing complete if the theory it is to prove all true Π^0_1 statements. However, Z cannot be 2-random and Turing complete at the same time, a contradiction.

Finally, how do we find such a degree $\mathcal{A}$? PA degrees are perfect for the job. The complete extensions of Peano arithmetic form a Π^0_1 class, so we can use a low basis theorem to find a PA degree $\mathcal{A}$ that is close to being computable. In our case *close to being computable* means: not derandomizing Z. PA degrees can also compute a member of any Π^0_1 class. In particular, our PA degree $\mathcal{A}$ will be able to compute a suitable function $\widetilde{K}$, as the requirements for $\widetilde{K}$ are Π^0_1.

Now we put all of this together into an actual proof.

Proof of Theorem 5.4.1. All false Π^0_1 sentences can be computably enumerated. Since "$\phi_e(e)\uparrow$" is a Π^0_1 statement, the set of all true Π^0_1 sentences is Turing complete. Hence a theory that proves all true Π^0_1 statements must be Turing complete as well. In particular, if $Z \not\geq_T \mathbf{0}'$, then $MLR_c(Z)$ is not Turing complete, so does not prove all true Π^0_1 statements.

Now consider the other case, $Z \geq_T \mathbf{0}'$. Using the low basis theorem for randomness ([16, 8.7.2]) we can take a PA degree $\mathcal{A}$ such that Z is still Martin-Löf random relative to $\mathcal{A}$.

Consider all total functions $f\colon 2^{<\omega} \to \mathbb{N}$ such that

$$f(\sigma) \leq K(\sigma) \qquad \text{for all } \sigma \in 2^{<\omega} \tag{26}$$

and

$$\sum_{\sigma \in 2^{<\omega}} 2^{-f(\sigma)} \leq 1. \tag{27}$$

These conditions are Π^0_1. Moreover, (26) together with the fact that there is a computable upper bound for K, makes the class of such functions f a *bounded* Π^0_1 class in Baire space ([16, p73 footnote 11]). Hence the PA degree $\mathcal{A}$ computes a member of this class, say $\widetilde{K}$.

Note that (27) makes $\widetilde{K}$ an information content measure ([16, 3.7.7]) relative to $\mathcal{A}$. By [16, 3.7.8], there is a constant c such that

$$K^{\mathcal{A}}(\sigma) - c \leq \widetilde{K}(\sigma)$$

for all $\sigma \in 2^{<\omega}$. Since Z is Martin-Löf random relative to $\mathcal{A}$, by a relativized version of Corollary 3.3.3 we have

$$\lim_{n\to\infty} \left(K^{\mathcal{A}}(Z\upharpoonright_n) - n\right) = \infty.$$

Hence there exists an $N \in \mathbb{N}$ such that for all $n \geq N$

$$K^{\mathcal{A}}(Z\upharpoonright_n) > n + c$$

and thus

$$\widetilde{K}(Z\upharpoonright_n) > n.$$

Consider the set

$$C = \left\{ \sigma \in 2^{<\omega} \,:\, \widetilde{K}(\sigma) > |\sigma| \right\}$$

and let T be the theory consisting of the axioms

$$\text{“}K(\sigma) > |\sigma|\text{”}$$

for all $\sigma \in C$ and

$$\text{“}K(Z\restriction_n) > n - c\text{”}$$

for all $n < N$.

This theory is consistent, because $\widetilde{K}$ is a lower bound for K, so $\widetilde{K}(\sigma) > |\sigma|$ implies $K(\sigma) > |\sigma|$. The theory T is also stronger (in the proof-theoretic sense) than $MLR_c(Z)$, since $Z\restriction_n \in C$ for all $n \geq N$. Finally, since $\widetilde{K}$ is $\mathcal{A}$-computable, so are the set C and the theory T.

As in the first paragraph of this proof, $\mathcal{A}$ must be Turing-complete if T is to prove all true Π^0_1 sentences. But then Z, which is random relative to $\mathcal{A}$, is at least 2-random. This contradicts the assumption that $Z \geq_T \mathbf{0}'$. Hence T can't prove all true Π^0_1 sentences, and neither can the weaker theory $MLR_c(Z)$. □

More results about $MLR_c(Z)$

When we increase the constant c, the theory $MLR_c(Z)$ becomes weaker. The next result shows that in the limit for $c \to \infty$, we get back to PA.

We write $MLR_c(\sigma)$ for the theory consisting of axioms “$K(\sigma\restriction_n) > n - c$” for $n \leq |\sigma|$.

Theorem 5.4.2.

Let Z be Martin-Löf random. Let ϕ be a sentence that is provable in $MLR_c(Z)$ for every c. Then ϕ is also provable in PA.

Note that if c is too small, then $MLR_c(Z)$ is inconsistent and ϕ is trivially provable in $MLR_c(Z)$.

Proof. Suppose that $MLR_c(Z)$ proves ϕ for every natural number c. We will prove that either Z is not Martin-Löf random, or that ϕ is provable in PA.

Consider the sets

$$U_c = \{Z \ : \ MLR_c(Z) \text{ proves } \phi\}$$

for every c. These sets are Σ^0_1 classes uniformly in c, as every proof involves only finitely many axioms.

If $\mu(U_c) \leq 2^{-c}$ for all c, then (U_i) is a Martin-Löf test that succeeds on Z. Hence Z is not Martin-Löf random and we are done.

In the other case, we have $\mu(U_c) > 2^{-c}$ for some c. Then there must be some length n such that more than a fraction 2^{-c} of all strings of length n have some initial segment enumerated into U_c, and this fact is provable in PA. On the other hand, PA can prove that at most a fraction 2^{-c} of all strings of length n have an initial segment σ with $K(\sigma) \leq |\sigma| - c$. (Otherwise the weight condition (6) for prefix-free complexity would be violated, just like in the proof of Theorem 3.3.1.) Hence PA proves the existence of some string σ of length n such that $MLR_c(\sigma)$ is true and $MLR_c(\sigma)$ proves ϕ. Therefore, PA proves ϕ itself, as required. □

Remark 5.4.3. *The proof for the second case ($\mu(U_c) > 2^{-c}$ for some c) can be seen as a special case of the* conservation theorem for random proofs

as proven by Alexander Shen [4]. The proof strategy *in this case consists of generating a random string σ of length n and adding the axioms of $MLR_c(\sigma)$.*

Another way of putting Theorem 5.4.2 is: for any Martin Löf random Z, the intersection of the theories $MLR_c(Z)$ over all numbers c is just PA. The same happens if we fix c and intersect over all Z instead. We don't even need to intersect over all sequences Z, a class of large enough measure suffices.

Theorem 5.4.4.
Let c be a natural number. Let ϕ be a sentence that is provable in $MLR_c(Z)$ for every sequence $Z \in \mathcal{A}$, where $\mathcal{A}$ is a subset of Cantor space with $\mu(\mathcal{A}) > 2^{-c}$. Then ϕ is also provable in PA.

It doesn't matter that we also include non-random sequences Z in the condition of the theorem, as $MLR_c(Z)$ will be inconsistent and ϕ will be trivially provable in $MLR_c(Z)$.

Proof. Suppose that $MLR_c(Z)$ proves ϕ for every sequence $Z \in \mathcal{A}$. Since $\mu(\mathcal{A}) > 2^{-c}$, there must be some length n such that $MLR_c(\sigma)$ proves ϕ for more than a fraction 2^{-c} of all strings σ of length n, and this fact is provable in PA. On the other hand, PA can prove that at most a fraction 2^{-c} of all strings of length n have an initial segment σ with $K(\sigma) \leq |\sigma| - c$. Just like in the proof of Theorem 5.4.2 above, we conclude that PA proves ϕ, as required. □

Other theories related to $MLR_c(Z)$

The article [4] investigates two other axiomatic theories that formalize the fact that a sequence Z is Martin-Löf random. The first theory expands the

language of PA with a new function symbol $\mathcal{Z}$, then adds an axiom

$$\text{“}\mathcal{Z}(n) = \overline{Z(n)}\text{”}$$

for every $n \in \mathbb{N}$ (where $\overline{Z(n)}$ is the actual value of the n'th digit of Z) and the Martin-Löf randomness of Z can now, thanks to the new function symbol $\mathcal{Z}$, be expressed in just one axiom:

$$\text{“}\forall n\colon K(\mathcal{Z}\upharpoonright_n) \geq n - c\text{”}.$$

Let's call this theory $MLR'_c(Z)$. This theory can certainly prove everything that is deducible from $MLR_c(Z)$. Moreover, $MLR'_c(Z)$ proves sentences like

$$Ext_c(Z\upharpoonright_n) = \text{“}\forall m \geq n\, \exists \tau \in 2^m\, (Z\upharpoonright_n \prec \tau \text{ and } \forall i \leq m\colon K(\tau\upharpoonright_i) \geq i - c)\text{”},$$

which express the fact that the $Z\upharpoonright_n$ can be extended to a string of any length such that all initial segments of that string have high complexity. This suggest that we also consider the theory $MLR''_c(Z)$ which contains the axioms “$Ext_c(Z\upharpoonright_n)$” for all n, but without the extra function symbol $\mathcal{Z}$.

It turns out that $MLR'_c(Z)$ and $MLR''_c(Z)$ can prove exactly the same sentences when they don't involve the extra function symbol $\mathcal{Z}$ (see [4] for a proof). In model theory terminology: $MLR'_c(Z)$ is a conservative extension of $MLR''_c(Z)$. Both of these theories are also strictly stronger than $MLR_c(Z)$. In fact, the converse of Theorem 5.4.1 holds for these theories.

Theorem 5.4.5.

If there exists a sequence Y such that $MLR_c(Y)$, then there exists a sequence Z such that $MLR''_c(Z)$ is consistent and proves all true Π^0_1 sentences.

The proof is similar to Theorems 5.3.1 and 5.3.3.

Proof. Consider the Π^0_1 class

$$\mathcal{A} = \{Y \in 2^\omega \,:\, \forall n(K(Y\upharpoonright_n) \geq n - c)\},$$

which is non-empty by assumption. Let Z be the left-most (i.e. lexicographically least) element of $\mathcal{A}$. Then $MLR''_c(Z)$ is consistent.

As in Theorems 5.3.1 and 5.3.3, it is now sufficient to prove that for every program P that does not terminate, the theory $MLR''_c(Z)$ can prove this non-termination.

For $n \in \mathbb{N}$, $Z\upharpoonright_n$ is the lexicographically first string τ of length n such that $Ext_c(\tau)$. The formula "$Ext_c(\tau)$" is Π^0_1, as the only existential quantifier is bounded, so we can enumerate all strings $\tau' \in 2^n$ such that $\neg Ext_c(\tau')$. Let s_n be the first stage at which all strings that come before $Z\upharpoonright_n$ have appeared in this enumeration. We prove that from a program P, we can compute a number n such that the computation P either terminates in less than s_n steps, or does not terminate at all.

Given a terminating program P and a number n, let $s(P)$ be the number of steps that P takes before halting. Let σ be the first string of length n with $K^{s(P)}(\sigma) \geq n - c$. If $s(P) > s_n$, then we know that $\sigma = Z\upharpoonright_n$. On the other hand, for every P that terminates we get a string σ of length n with

$$\begin{aligned} K(\sigma) &< K(P, n) + O(1) \\ &< K(P) + O(\log n). \end{aligned}$$

Given P, we can find an n that is large enough such that $K(\sigma) < n - c$. For such an n, we know that σ is different from $Z\upharpoonright_n$, so P must have halted

within s_n steps. Consequently, if a program P terminates at all, then it must do so in less than s_n steps.

Once again, this reasoning can be formalized inside PA. Having "$Ext_c(Z\upharpoonright_n)$" as an axiom, it is provable that $Z\upharpoonright_n$ is indeed the lexicographically least string τ of length n with $Ext_c(\tau)$, as "$\neg Ext_c(\tau')$" is provable for all preceding strings τ'. Then, given n, it is provable that the value of s_n satisfies its definition. Finally, given P and taking n suitably large, it is provable (doing the above proof inside PA) that P either halts within s_n steps or does not terminate at all. □

Theorem 5.4.5 shows that $MLR''_c(Z)$ can be a strictly stronger theory than $MLR_c(Z)$. However, when we intersect over all possible values of c, we still get the same result as in Theorem 5.4.2.

Theorem 5.4.6.
Let Z be Martin-Löf random. Let ϕ be a sentence that is provable in $MLR''_c(Z)$ for every c. Then ϕ is also provable in PA.

Proof. Identical to the proof of Theorem 5.4.2, using the fact that PA can prove that at most a fraction 2^{-c} of all strings σ of a given length satisfy $\neg Ext_c(\sigma)$. □

As an interesting corollary, consider the theory $MLR'(Z)$ which is obtained from $MLR'_c(Z)$ by replacing the axiom

$$\text{“}\forall n\colon K(\mathcal{Z}\upharpoonright_n) \geq n - c\text{”}$$

by the weaker axiom

$$\text{“}\exists c\,\forall n\colon K(\mathcal{Z}\upharpoonright_n) \geq n - c\text{”}.$$

This theory is actually a conservative extension of PA.

Theorem 5.4.7.
If Z is Martin-Löf random and ϕ is a sentence in the language of PA that is deducible from $MLR'(Z)$, then ϕ is already provable in PA.

Proof. If ϕ is provable in $MLR'(Z)$ then it is also provable in $MLR'_c(Z)$ for every c. By the earlier remarks, the theory $MLR'_c(Z)$ is a conservative extension of the theory $MLR''_c(Z)$, so ϕ is also provable in $MLR''_c(Z)$ for every c. By Theorem 5.4.6, ϕ is provable in PA. □

Theorem 5.4.7 expresses in a sense the idea that, since almost all sequences are Martin-Löf random, the fact that the sequence given by some function symbol is Martin-Löf random should not give useful information.

5.5 Axioms expressing 2-randomness

A sequence is 2-random if and only if there exists a constant c such that

$$C(Z\upharpoonright_n) > n - c \tag{28}$$

for infinitely many n ([43, 49], see also [48, 3.6.10] or [16, 6.11.6]). (Note that because of the *complexity dips* for plain complexity, *no* sequence satisfies (28) for *all* n.) So we can consider a theory $2R_{A,c}(Z)$ that expresses that Z is 2-random using the axioms

$$\text{“}C(Z\upharpoonright_n) > n - c\text{”}$$

for all $n \in A$ where A is some infinite set of natural numbers. For fixed Z and c such that (28) for infinitely many n, the strongest consistent theory

among these is the one where A is maximal, i.e.

$$A = \{n \in \mathbb{N} : \ C(Z\restriction_n) > n - c\}\,.$$

The fact that 2-randomness implies Martin-Löf randomness is reflected in these theories.

Theorem 5.5.1.
Any theory $2R_{A,c}(Z)$ implies $MLR_{c'}(Z)$ for some c'.

Proof. Consider the machine M that on input σ tries to find a splitting $\sigma = \rho\tau$ such that $\mathbb{U}(\rho)\downarrow$. If successful, it outputs $\mathbb{U}(\rho)\tau$. Let d be the coding constant for M, i.e.

$$C(\sigma) \leq C_M(\sigma) + d$$

for all σ. Suppose for contradiction that $K(Z\restriction_n) \leq n - (d + c)$ for some n. Then for $m \geq n$, the string $Z\restriction_m$ has an M-description of length at most $m - (d + c)$, and hence

$$C(Z\restriction_m) \leq m - (d + c) - d = m - c$$

contradicting the axioms of $2R_{A,c}(Z)$. □

Are the consistent theories $2R_{A,c}(Z)$ strictly stronger than the theories $MLR_c(Z)$? Can they prove all true Π^0_1 sentences? Actually, we already asked exactly this question as Question 5.3.4, so this is an open problem.

5.6 Axioms that give exact complexities

How much information the exact complexities of strings (i.e. true axioms of the form "$K(\sigma) = n$") contain, might depend a lot on the universal machine used.

Theorem 5.6.1.

There is a universal machine M such that for any set X that contains a string of any length, the axioms "$K(\sigma) = \overline{K(\sigma)}$" for every string $\sigma \in X$ (where $\overline{K(\sigma)}$ is the numerical value of $K(\sigma)$) prove all true Π^0_1 sentences.

Proof. Let $\mathbb{U}$ be the standard universal machine, as constructed in Section 2.4. Let $H = \{e \in \mathbb{N} : \ \phi_e(e)\downarrow\}$ be the halting set.

Define M as follows. If τ has an even length, let

$$M(1\tau) = \begin{cases} \mathbb{U}(\tau) & \text{if } \mathbb{U}(\tau)\downarrow \text{ and } |\mathbb{U}(\tau)| \in H, \\ \uparrow & \text{otherwise;} \end{cases}$$

$$M(01\tau) = \mathbb{U}(\tau).$$

If τ has an odd length, let

$$M(01\tau) = \begin{cases} \mathbb{U}(\tau) & \text{if } \mathbb{U}(\tau)\downarrow \text{ and } |\mathbb{U}(\tau)| \in H, \\ \uparrow & \text{otherwise;} \end{cases}$$

$$M(001\tau) = \mathbb{U}(\tau).$$

Finally, M is undefined on all other inputs. It is easy to see that M is a universal machine. Now, if $|\sigma| \notin H$, then σ only has descriptions of even length. If $|\sigma| \in H$, then σ has a shortest description of odd length. So from the parity of the complexity of any string of length n, we can decide if n is in the halting set or not. This argument can be done inside PA as well. □

Note that the machine M constructed in the theorem is even *provably* universal in the sense of Section 5.2. The theorem also works for plain complexity C, by replacing the prefix-free universal machine $\mathbb{U}$ with the plain

universal machine $\mathbb{V}$ in the proof.

Question 5.6.2.

Is there a universal machine such that adding the exact complexities for infinitely many strings doesn't always prove all true Π^0_1 *statements?*

This is possibly even the case for the standard universal machine $\mathbb{U}$.

5.7 Summary

The main results from this chapter are summarized in Figure 11.

Does there exist $A \subseteq 2^{<\omega}$ such that all true Π^0_1 sentences are provable with consistent axioms...

	"$C(\sigma) > |\sigma| - c$" for $\sigma \in A$	"$K(\sigma) > |\sigma| - c$" for $\sigma \in A$
A contains at most one string of each length.	**Yes**	**Yes**
A contains infinitely many initial segments of a sequence.	**Maybe** Note: axioms imply that sequence is 2-random	**Yes**
A contains all initial segments of a sequence.	Axioms are never consistent	**No** Note: axioms imply that sequence is 1-random

Figure 11: Summary of results about the strength of theories whose axioms express that certain strings have high complexities.

Bibliography

[1] Laurent Bienvenu, Adam Day, Mathieu Hoyrup, Ilya Mezhirov, and Alexander Shen. A constructive version of Birkhoff's ergodic theorem for Martin-Löf random points.

[2] Laurent Bienvenu, Adam Day, Ilya Mezhirov, and Alexander Shen. Ergodic-type characterizations of algorithmic randomness. In *Programs, Proofs, Processes*, volume 6158 of *Lecture Notes in Computer Science*, pages 49–58. Springer, Berlin/Heidelberg, 2010.

[3] Laurent Bienvenu, Rupert Hölzl, Thorsten Kräling, and Wolfgang Merkle. Separations of non-monotonic randomness notions. *6th International Conference on Computability and Complexity in Analysis (CCA 2009)*, 2009.

[4] Laurent Bienvenu, Andrei Romashchenko, Alexander Shen, Antoine Taveneaux, and Stijn Vermeeren. The axiomatic power of Kolmogorov complexity. To be published in the Annals of Pure and Applied Logic.

[5] Laurent Bienvenu, Glenn Shafer, and Alexander Shen. On the history of martingales in the study of randomness. *Journal Electronique d'Histoire des Probabilités et de la Statistique*, 5(1), 2009. `http://www.jehps.net/juin2009.html`.

[6] Vasco Brattka, Joseph S. Miller, and André Nies. Randomness and differentiability.

[7] Harry Buhrman, Dieter Van Melkebeek, Kenneth W. Regan, D. Sivakumar, and Martin Strauss. A generalization of resource-bounded measure, with application to the BPP vs. EXP problem. *SIAM Journal on Computing*, 30:576–601, 2000.

[8] Cristian S. Calude and André Nies. Chaitin ω numbers and strong reducibilities. *Journal of Universal Computer Science*, 3(11):1162–1166, 1997.

[9] Gregory J. Chaitin. Computational complexity and Gödel's incompleteness theorem. *ACM SIGACT News*, (9):11–12, 1971.

[10] Gregory J. Chaitin. Information-theoretic limitations of formal systems. *Journal of the ACM*, 21:403–424, 1974.

[11] Gregory J. Chaitin. A theory of program size formally identical to information theory. *Journal of the ACM*, 22:329–340, 1975.

[12] Gregory J. Chaitin. Incompleteness theorems for random reals. *Advances in Applied Mathematics*, 8:119–146, 1987.

[13] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathematical Statistics*, 23(4):493–507, 1952.

[14] Alonzo Church. On the concept of a random sequence. *Bulletin of the American Mathematical Society*, 46:130–135, 1940.

[15] Barry Cooper. *Computability Theory*. Chapman & Hall, 2003.

[16] Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity*. Theory and Applications of Computability. Springer, 2011.

[17] William Feller. *An Introduction to Computability Theory and its Applications*. Wiley, New York, 1957.

[18] Johanna N. Y. Franklin, Noam Greenberg, Joseph S. Miller, and Keng Meng Ng. Martin–Löf random points satisfy Birkhoff's ergodic theorem for effectively closed sets.

[19] Johanna N. Y. Franklin and Keng Meng Ng. Difference randomness. *Proceedings of the American Mathematical Society*, 139:345–360, 2011.

[20] Péter Gács. Every sequence is reducible to a random one. *Information and Control*, 70:186–192, 1986.

[21] Kurt Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, i. *Monatshefte für Mathematik und Physik*, 38:173–198, 1931.

[22] Paul R. Halmos. *Measure Theory*. D. Van Nostrand Company, Inc., 1950.

[23] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[24] Bart Kastermans and Steffen Lempp. Comparing notions of randomness. *Theoretical Computer Science*, 411(3):602–616, 2010.

[25] Steven M. Kautz. *Degrees of random sets*. PhD thesis, Cornell University, 1991.

[26] Andrey N. Kolmogorov. Three approaches to the quantitative definition of information. *Problems in Information Transmission*, 1:1–7, 1965.

[27] Andrey N. Kolmogorov. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, 25(4):369–376, 1966.

[28] Dénes Kőnig. *Theorie der endlichen und unendlichen Graphen*. Akademische Verlagsgesellschaft, Leipzig, 1936.

[29] Antonín Kučera. Measure, Π^0_1-classes and complete extensions of PA. In *Recursion Theory Week (Oberwolfach, 1984)*, volume 1141 of *Lecture Notes in Mathematics*, pages 245–259. Springer, Berlin, 1985.

[30] Martin Kummer. On the complexity of random strings (extended abstract). In *13th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1046 of *Lecture Notes in Computer Science*, pages 25–36. Springer, 1996.

[31] Stuart A. Kurtz. *Randomness and genericity in the degrees of unsolvability*. PhD thesis, University of Illinois at Urbana-Champaign, 1981.

[32] Michiel Van Lambalgen. *Random sequences*. PhD thesis, Universiteit van Amsterdam, 1987.

[33] Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Verlag, 1993.

[34] Elliott H. Lieb, Daniel Osherson, and Scott Weinstein. Elementary proof of a theorem of Jean Ville. 2006, arXiv:cs/0607054.

[35] Donald W. Loveland. The Kleene hierarchy classification of recursively random sequences. *Transactions of the American Mathematical Society*, 125(3):497–510.

[36] Donald W. Loveland. A new interpretation of the von Mises' concept of random sequence. *Mathematical Logic Quarterly*, 12(1):279–294, 1966.

[37] Donald W. Loveland. A variant of the Kolmogorov concept of complexity. *Information and Control*, 15(6):510–526, 1969.

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

[39] Per Martin-Löf. On the notion of randomness. In *Intuitionism and proof theory, proceedings of the summer conference at Buffalo, N.Y. 1968*, pages 73–78, 1970.

[40] Elliott Mendelson. *Introduction to Mathematical Logic*. Chapman and Hall, fourth edition, 1997.

[41] Wolfgang Merkle. The Kolmogorov-Loveland stochastic sequences are not closed under selecting subsequences. *Journal of Symbolic Logic*, 68(4):1362–1376, 2003.

[42] Wolfgang Merkle. The complexity of stochastic sequences. *Journal of Computer and System Sciences*, 74(3):350–357, 2008.

[43] Joseph S. Miller. Every 2-random real is Kolmogorov random. *Journal of Symbolic Logic*, 69(3):907–913, 2004.

[44] Joseph S. Miller and André Nies. Randomness and computability: open questions. *Bulletin of Symbolic Logic*, 12(3):390–410, 2006.

[45] Joseph S. Miller and Liang Yu. On initial segment complexity and degrees of randomness. *Transactions of the American Mathematical Society*, 360(6):3193–3210, 2008.

[46] Andrei A. Muchnik, Alexei L. Semenov, and Vladimir A. Uspensky. Mathematical metaphysics of randomness. *Theoretical Computer Science*, 207:263–317, 1998.

[47] James R. Munkres. *Topology*. Prentice Hall, second edition, 2000.

[48] André Nies. *Computability and Randomness*. Oxford University Press, 2009.

[49] André Nies, Frank Stephan, and Sebastiaan A. Terwijn. Randomness, relativization and Turing degrees. *Journal of Symbolic Logic*, 70(2):515–535, 2005.

[50] Marian B. Pour-El and J. Ian Richards. *Computability in analysis and physics*. Perspectives in Mathematical Logic. Springer Verlag, 1989.

[51] Jan Reimann and Theodore A. Slaman. Measures and their random reals.

[52] Marc Renault. Four proof of the ballot theorem. *Mathematics Magazine*, 80:345–352, 2007.

[53] Claus-Peter Schnorr. *Zufälligkeit und Wahrscheinlichkeit. Eine algoritmische Begründung der Wahrscheinlichkeitstheorie*, volume 218 of *Lecture Notes in Mathematics*. Springer-Verlag, 1971. Available online at `http://www.leibniz-publik.de/de/fs1/object/display/bsb00057178_00001.html`.

[54] Claus-Peter Schnorr. Process complexity and effective random tests. *Journal of Computer and System Sciences*, 7:376–388, 1973.

[55] Alexander Kh. Shen. On relations between different algorithmic definitions of randomness. *Soviet Mathematics Doklady*, 38:316–319, 1989.

[56] Ray J. Solomonoff. A formal theory of inductive inference, part i. *Information and Control*, 7:1–22, 1964.

[57] Ray J. Solomonoff. A formal theory of inductive inference, part ii. *Information and Control*, 7:224–254, 1964.

[58] Frank Stephan. Martin-Löf random and PA-complete sets. In *Logic Colloquium '02*, volume 27 of *Lecture Notes in Logic*, pages 342–348. Association for Symbolic Logic, 2006.

[59] Teiji Takagi. A simple example of the continuous function without derivative. *Proceedings of the Physico-Mathematical Society of Japan*, 1:176–177, 1903.

[60] Jean Ville. *Étude critique de la notion de collectif*. Monographies des probabilités. Gauthier-Villars, Paris, 1939. As PhD thesis available online at `http://www.numdam.org/item?id=THESE_1939__218__1_0`.

[61] Richard von Mises. Grundlagen der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, 5:52–99, 1919.

[62] A. Wald. Die Widerspruchsfreiheit des Kollektivbegriffes der Wahrscheinlichkeitsrechnung. In *Ergebnisse eines mathematischen Kolloquiums*, volume 8, pages 38–72, 1936.

[63] Yongge Wang. *Randomness and complexity*. PhD thesis, Fakultät für Mathematik, Ruprecht Karls Universität Heidelberg, 1993.