The Classes of Algorithmically Random Reals

by

Stephanie Reid

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the
requirements for the degree of
Master of Science
in Computer Science.

Victoria University of Wellington
2003

Abstract

This work outlines various definitions of Martin-Löf randomness (the standard definition of algorithmic randomness), computable randomness, Schnorr randomness and Kurtz randomness, and describes how these ideas can be extended to an infinite hierarchy of randomness classes that is based on the arithmetic hierarchy. We provide a characterization of Kurtz randomness based on computable machines. Furthermore, we investigate s-randomness, and provide a characterization of the s-randomness classes in terms of prefix-free machines. Whilst standard randomness only requires random numbers to avoid measure zero sets, s-randomness takes the Hausdorff dimension of a measure zero set into account as well, to create a dense collection of randomness classes.

Acknowledgments

I have been privileged to have had Rod Downey as my supervisor for Masters. I am very fortunate to have had the opportunity to benefit from his boundless knowledge, enthusiasm and tolerance.

I would like to thank Evan Griffiths, who generously spent many hours of his time helping me with my studies. Without him, much of this thesis could not have been written. Also thanks to Jan Reimann, for helpful suggestions on Chapter 16.

During 2002 funding provided through the Helen Stewart Royle Scholarhip, the William Georgetti Scholarship, and the Victoria University of Wellington Postgraduate Scholarship for Masters, and I am very grateful to the benefactors of these scholarships for this support.

I am also indebted to my family and friends for all the support they have given me in many different ways.

Contents

Chapter 1

Introduction

This work covers the techniques and results that form the foundation of the field of algorithmic randomness. Randomness as considered here stems from the effectivization of probabilistic sets, and as such, is inextricably linked with computability theory. Randomness is not a black or white concept, and so there are various definitions of randomness, each of which separate the real numbers into two classes - random and non-random - in slightly different ways. In each case however, the random reals encompass the vast majority of all reals, whilst the non-random reals represent only a measure zero subset of the reals.

Much of this work involves manipulation of collections of binary strings. With this in mind, we will frequently use the following notation. If σ and τ are binary strings, we write $\sigma \sqsubseteq \tau$ if σ is a substring of τ. $\sigma \sqsubset \tau$ means that σ is strictly a substring of τ. λ denotes the empty string (that is, a string of length 0). A set A of strings if prefix-free if and only if for all strings $\sigma \in A$, if τ is a string such that $\sigma \sqsubset \tau$, $\tau \notin A$.

Due to the nature of this field certain themes and techniques reappear frequently throughout this work. Although reworking similar problems is omitted as much as possible to avoid becoming repetitious, we attempt to note these themes in order to clarify the internal logic of this theory, and because both the similarities and differences between the various classifications are of interest.

1.1 Overview

Part I provides the background to the study of randomness. There are two main streams of mathematical theory that merge to give us the tools to tackle this field. The first is the study of probability through measurable sets. The tools required for this are covered in Chapter 5. The other is computability theory, which we use to effectivize our measure theoretic work. We provide an overview of this in Chapter 2. Chapters 3 and 4 provide some slightly more advanced ideas and results from computability theory. However, these chapters are less important for our work and the results are only really used in Chapters 12 and 15.

Part II describes the development of the concept and definition of algorithmic randomness. Chapter 6 surveys the work in the field up to 1960. This work was based on the current understanding of probability but lacked (in the early years) the knowledge of effective functions that allowed a meaningful characterization to be formulated. The problem faced by mathematicians at this time was the lack of an “absolute” standard of randomness, and all their attempts were proved to be lacking in some way. The use of computability theory to provide an “effective” (if not an absolute) type of randomness was necessary in order to bypass this problem. By the time the use of computable functions was suggested (by Church in 1940), the interest in finding a characterization of randomness had been lost because new axioms of probability that did not require a formalization of this concept had been invented.

The later chapters in this section look at more successful ways of dealing with randomness. These include Martin-Löf’s formulation of randomness as avoidance of measure zero sets (Chapter 8), the formal association of randomness with incompressibility (Chapters 7 and 11) and with predictability (Chapter 9). We also look at Schnorr and computable randomness in Chapter 10 and Kurtz randomness in Chapter 13. Chapter 12 considers the halting probability as an important example of a random real.

In Part III we show how to extend the randomness definitions to an infinite hierarchy, based on the the arithmetic hierarchy, and use Hausdorff dimension to create a dense hierarchy of classes. We also prove (in Chapter 14) that the definitions covered in Part II are different.

1.2 References

We provide here a chapter-by-chapter list of the resources used most frequently as references during the writing of this thesis:

Chapters 2, 3, 4:

- Hartley Rogers, *Theory of recursive functions and effective computability*, [38].
- Robert Soare, *Recursively enumerable sets and degrees*, [42].

Chapter 5:

- Rod Downey and Denis Hirschfeldt, *Algorithmic randomness and complexity*, [15]
- Gerald Edgar, *Integral, probability and fractal measures*, [18].
- Murray Spiegel, *Advanced Calculus*, [46]

Chapter 6:

- Klaus Ambos-Spies and Antonin Kučera, *Randomness in Computability Theory*, [1].

- Ming Li and Paul Vitanyi, *An Introduction to Kolmogorov Complexity and its Applications*, [31].
- Michiel van Lambalgen, *Random Sequences*, [50].
- Yongge Wang, *Randomness and Complexity*, [54].

Chapters 7, 8:

- Ming Li and Paul Vitanyi, *An Introduction to Kolmogorov Complexity and its Applications*, [31].

Chapter 9:

- Elvira Mayordomo, *Contributions to the study of resource-bounded measure*, [36].
- Claus-Peter Schnorr, *A unified approach to the definition of random sequences*, [39].

Chapter 12, Section 12.1:

- Kohtaro Tadaki, *A Generalization of Chaitin's Halting Probability Ω and Halting Self-Similar Sets*, [47].
- Lance Fortnow, *My Computation Complexity Web Log*, [20].

Chapter 12, other sections:

- Cristian Calude, Peter Hertling, Bakhadyr Khoussainov and Yongge Wang, *Recursively enumerable reals and Chaitin Ω numbers*, [4]
- Cristian Calude and André Nies, *Chaitin Ω numbers and strong reducibilities*, [5]
- Rod Downey, Denis Hirschfeldt and Geoffrey Laforte, *Randomness and reducibility*, [16]

- Rod Downey and Denis Hirschfeldt, *Algorithmic randomness and complexity*, [15]
- Robert Solovay, *Draft of a paper (or series of papers) on Chaitin's work*, [45].

Chapter 13:

- Stuart Kurtz, *Randomness and genericity in the degrees of unsolvability*, [28].
- Yongge Wang, *Randomness and Complexity*, [54].

Chapter 14:

- Steven Kautz, *Degrees of random sets*, [23].
- Stuart Kurtz, *Randomness and genericity in the degrees of unsolvability*, [28].
- Claus-Peter Schnorr, *A unified approach to the definition of random sequences*, [39].
- Yongge Wang, *Randomness and Complexity*, [54].

Chapter 15:

- Steven Kautz, *Degrees of random sets*, [23].
- Stuart Kurtz, *Randomness and genericity in the degrees of unsolvability*, [28].

Chapter 16, Section 16.1:

- Gerald Edgar, *Measure, Topology and fractal geometry*, [17].
- Kenneth Falconer, *Fractal geometry: mathematical foundations and applications*, [19].

- Claude Rogers, *Hausdorff measures*, [37].

Chapter 16, other sections:

- Rod Downey and Denis Hirschfeldt, *Algorithmic randomness and complexity*, [15].
- Jack Lutz, *The dimensions of individual strings and sequences*, [32].
- Jack Lutz, *Dimension in Complexity classes*, [34].
- Kohtaro Tadaki, *A Generalization of Chaitin's Halting Probability Ω and Halting Self-Similar Sets*, [47].
- Sebastiaan Terwijn, *Complexity and randomness*, [48].

1.3 Summary of results

New results in this work include:

- Theorems 71, 72 and 73 in Chapter 13. These provide characterizations for Kurtz randomness in terms of machines, and Solovay tests. Also in Chapter 13, Theorem 74 shows that Kurtz test sets can be adapted to characterize Schnorr randomness. These results form part of [14].
- Section 15.3 of Chapter 15 shows that the hierarchy of randomness classes imposed by the difference hierarchy collapses to be equivalent to 2-randomness.
- Theorem 107 of Chapter 16 provides a machine definition of s-randomness.

Part I

Background material

Chapter 2

Computability theory

2.1 Algorithms

An algorithm is a sequence of detailed, mechanical instructions for performing a task, so that a person or machine following the algorithm exactly can do the task without any actions except those detailed by the algorithm. A computer is any machine (or person) that is capable of performing an algorithm, and computability theory is the study of what functions can be performed by an algorithm.

The functions we are interested in are defined on the natural numbers:

$$\omega = \{0, 1, 2, 3, \ldots\}.$$

A function f takes an input of one or more natural numbers $x_1, x_2, \ldots, x_n \in \omega$ and outputs a number $y \in \omega$. Those functions which can be described by an algorithm are called the *computable* functions. Those that have no corresponding algorithm are called *non-computable*.

It is obvious that all the traditional functions (for example, addition, subtraction, multiplication and division) can be written formally as algorithms. In fact school-children are taught algorithms for calculating these functions at an early age. It is harder to think of a non-computable function because all the functions we are used to are intuitively computable.

However, later in this chapter we shall show that in fact not all functions are computable.

The earliest work in the field of computability revolved around finding formal models of computation; that is, finding a precise description of those functions that are computable. Although the intuitive meaning of a computable function is clear, it is trickier to create a mathematical model that captures it. The key figures of this time (including Church, Gödel, Kleene, and Turing) suggested a variety of possible models; here, we consider the partial recursive functions. The partial recursive functions are based on a set of functions known as the primitive recursive functions:

Definition 1 (The Primitive Recursive Functions). *The class C of primitive recursive functions contains any function that can be built from applications of the following:*

1. *$Z(x)$, the zero function. For all $x \in \omega$, $Z(x) = 0$.*

2. *$S(x)$, the successor function. For all $x \in \omega$, $S(x) = x + 1$.*

3. *$P_{n,m}(x_1, \ldots, x_n)$, the projection functions. For all $n, m, x_1, \ldots, x_n \in \omega$ such that $m \leq n$, $P_{n,m}(x_1, \ldots, x_n) = x_m$.*

4. *Composition. If $g_1, g_2, \ldots, g_m, h \in C$ where h is a function of m variables and for each $i \leq m$, g_i is a function of n variables then*

$$f(x_1, \ldots, x_n) = h(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$$

   is also in C.

5. *Primitive Recursion. If $g, h \in C$ where g is a function of $n - 1$ variables and h is a function of $n + 1$ variables then*

$$\begin{aligned} f(0, x_2, \ldots, x_n) &= g(x_2, \ldots, x_n) \\ f(x_1 + 1, x_2 \ldots, x_n) &= h(x_1, x_2, \ldots, x_n, f(x_1, \ldots, x_n)) \end{aligned}$$

   is also in C.

The primitive recursive functions are not in themselves a suitable class to use as the computable functions, as they exclude functions that are intuitively computable. We build such a function now. It is clear that the primitive recursive functions can be *enumerated* - they can be given an ordering and be written out in a long list. Let us do this, and define f_n to be the n-th member of this list. Then define the function

$$g(n) = f_n(n) + 1.$$

The function g cannot be primitive recursive. If it were so, then for some i, g is f_i in the enumeration of the primitive recursive functions. But by definition,

$$\begin{aligned} g(i) &= f_i(i) + 1 \\ &= g(i) + 1, \end{aligned}$$

which is absurd. However, g is clearly computable in an intuitive sense; $g(n)$ can be computed simply by adding 1 to the result of $f_n(n)$. This means that the primitive recursive functions cannot be an adequate characterization of computable functions. Actually, the same "diagonalization" argument can be applied to any class of *total* functions - functions that give an answer for every input. Obviously, a suitable class of computable functions must include *partial functions* - functions that don't give an answer for an input but just fail to terminate. Then the diagonalization argument cannot work because perhaps $f_i(i)$ doesn't terminate (we write this $f_i(i) \uparrow$ - conversely if a function does halt we write $f_i(i) \downarrow$). In this case, $g(i) = g(i) + 1$ isn't a contradiction because $g(i)$ might not exist anyway. One way of extending the primitive recursive functions is to a class of partial recursive functions as follows:

Definition 2 (The Partial Recursive Functions, Kleene). *If $\psi(y, x_1, \ldots, x_n)$ is a partial recursive function of $n + 1$ variables, then the minimisation function*

$(\mu y)\psi(y, x_1, \ldots, x_n)$ *is a partial recursive function of* n *variables, defined as*

$$(\mu y)\psi(y, x_1, \ldots, x_n) = \begin{cases} m & \textit{if } \psi(m, x_1, \ldots, x_n) \downarrow= 0 \\ & \quad \textit{and } (\forall k \leq m)\psi(k, x_1, \ldots, x_n) \downarrow\neq 0 \\ \uparrow & \textit{otherwise} \end{cases}$$

The class of partial recursive functions is the smallest class of functions that contains the primitive recursive functions and is closed under minimisation.

2.2 Turing machines

Another way of considering the computable functions is via Turing machines. From a modern point of view this is a way of thinking about computable functions that is very natural, as Turing machines are the basis for our computers and the imperative programming languages that are most commonly used today. Introduced by Alan Turing [49] in 1936, his machine (which he called a Logic Computing Machine but which is now known as a Turing machine) was very simple. It consisted of a very long (essentially infinite) tape, that has been marked off into squares. The Turing machine has a head that can move up and down the tape. It can read, write and erase symbols on the square directly beneath it. (see Figure 2.1).

The head of the Turing machine has a finite number of states $q_0, q_1, \ldots q_n$ that it can be in. For each possible state, it has a set of instructions that tells it how to behave. This behaviour will vary depending on what it reads on the tape square it is currently above. For simplicity, imagine we restrict ourselves to using an alphabet containing just one symbol (the number 1) will be used. So the Turing machine will need two instructions for each of its states. It will need to know how to behave when it is over a blank square, and how to behave when it is over a square containing the symbol 1.

These instructions are typically written as quadruples. Here is an example of a quadruple:

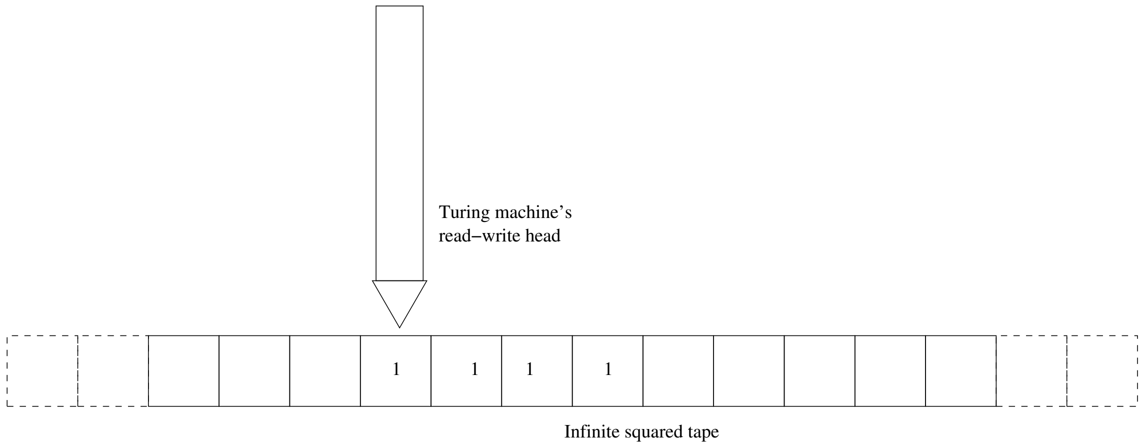

Figure 2.1: A schematic depiction of a Turing machine

$$< \ \underset{\substack{\text{Start}\\\text{state}}}{q_0}, \ \underset{\substack{\text{Symbol}\\\text{read}}}{1}, \ \underset{\substack{\text{Action}\\\text{taken}}}{\text{move right}}, \ \underset{\substack{\text{End}\\\text{state}}}{q_1} \ >$$

This particular quadruple instruction would be followed if the head of the Turing machine was in state q_0, and the symbol it was reading was a 1. If it is reading a blank square, it would require another quadruple, perhaps

$$< \ q_0, \ \texttt{BLANK}, \ \text{print } 1, \ q_2 \ > .$$

The actions the Turing machine can take are:

1. move left;
2. move right;
3. write on current square;
4. erase current square.

Once it has performed the required action, it changes into the end state, say state q_1. There will be other quadruples telling the Turing machine how to behave when it is in state q_1. This continues until the Turing machine reaches a special state, the HALT state. At this point it stops and the computation is now complete. Note that a machine T, given some input x, may never enter the Halt state, and may simply run forever. An example of this is a modern computer program entering an "infinite loop". In that case, we write $T(x)\uparrow$. If T does halt on input x we write $T(x)\downarrow$.

It is possible to write algorithms that this simple computer can perform. One of the simplest (non-trivial) programs for a Turing machine is the program that when given input n, adds 1 to n. Let us call this program P. P interprets n adjacent 1's as the natural number n.

A Turing machine assumes when it starts running that the read/write head is at the far left of the input. Furthermore, it is customary that when a Turing machine halts it is again at the far left of the output (as then it would be ready to do another computation, if required).

If a Turing machine were to perform P it would first have to move along to the end of the input. It would know that it was at the end of the input when it reads a blank square for the first time. At this point it would print another 1. Since there were n 1's originally on the tape, there are now $n+1$ 1's. The Turing machine must then return to the beginning of the input. When it arrives there it is done. It has taken an input n and outputted $n+1$.

We could write P in pseudo-code as follows:

while(read 1) move **RIGHT**
then print 1

while(read 1) move **LEFT**
then move **RIGHT**

HALT

This program can then be converted into quadruples that a Turing machine could understand. Note how the **while** loops are simulated using the states of the Turing machine:

$$\begin{array}{llll}
< q_0, & 1, & \text{move right}\ , & q_0 > \\
< q_0, & \text{blank}, & \text{print } 1, & q_1 > \\
< q_1, & 1, & \text{move left}\ , & q_1 > \\
< q_1, & \text{blank}, & \text{move right}\ , & q_{\text{HALT}} > .
\end{array}$$

It is obviously terribly cumbersome to write even as simple a program as this for a Turing Machine. However, it certainly can be done. Furthermore, a similar program could be written for a Turing machine to perform subtraction, or we could develop P further into a more complicated program. For example, we could write a program $\widehat{P}$ for a Turing machine that would take two numbers as input (say n and m) and add them together, outputting $n + m$.

2.3 The Church-Turing Thesis

Let f be a Turing computable function. This means that there is a Turing machine M which, given any $x \in \omega$ as input, outputs $f(x)$. It has been shown that the class of Turing computable functions is identical to both the class of partial recursive functions as defined earlier. Furthermore, other formal models have been suggested, such as Minsky machines and the lambda calculus. In each case, it turns out that they compute exactly the same class of algorithms as Turing machines do.

This suggests that this particular class of functions is in a sense special. No matter which way computability is characterized, these same functions are attained. The implication is that these functions must be precisely those that humans intuitively see as the computable functions.

The Church-Turing Thesis is a formalization of this conclusion - that the computable functions (i.e. any algorithmic function) are precisely the

Turing computable functions. Because of this Turing computable functions have become synonymous with effectively computable functions. In this thesis we will assume that this is the case, and freely interchange the terms "computable function", "(partial) recursive function" and "Turing machine", or even just "machine".

2.4 Numbers and sequences

The natural numbers ω are in one-to-one correspondence with the set of finite binary sequences $2^{<\omega}$ by the length-lexicographic ordering (see Table 2.1).

ω	$2^{<\omega}$
0	λ
1	0
2	1
3	00
4	01
5	10
6	11
$\vdots$	$\vdots$

Table 2.1: Correspondence between ω and $2^{<\omega}$.

Because of this correspondence it is possible to assume that computable functions have finite binary sequences as input. In fact, we will from now on allow our Turing machines to read binary strings - so as to be able to have 1s or 0s as input and to differentiate between them. This makes the definition of the quadruples slightly more technical, but gives us greater variety in the input we are able to use. It makes no difference to the class of functions that are described via Turing machines. This can be seen by appealing to the Church-Turing thesis, or more directly by considering how a Turing machine that accepts binary could be redefined to compute the same function using only 1's, and vice versa.

Similarly, any *subset* of the natural numbers has a one-to-one correspondence with the set of *infinite* dyadic sequences. Given $X \subseteq \omega$, an equivalent infinite binary sequence x (with bits $x[n]$ for each n) can be defined for X via

$$x[n+1] = \begin{cases} 1 & \text{if } n \in X \\ 0 & \text{otherwise.} \end{cases}$$

This associated sequence x is unique to X, and is known as the *characteristic function* of X. The importance of this relationship between infinite binary sequences and sets will be discussed in Chapter 5, where we look at the real numbers.

2.5 The Universal Turing machine

The Turing machine discussed above was defined so that when given input n it would compute $n+1$. However it is possible to build a so-called *Universal Turing machine* that like a modern computer is capable of emulating any Turing computable function. To build such a machine, we want to effectively enumerate the Turing machines; that is, to create an ordered list $T_0, T_1, \ldots$ of all the Turing machines. There are various ways to do this, but a common way is to order the Turing machines according to increasing Gödel numbers. Gödel numbering assigns a unique value to each Turing machine, such that given a machine T we can compute its Gödel number and vice versa.

Let T be a Turing machine. Each Turing machine is uniquely defined by the (finite) set of quadruples in its program, $Q_0, Q_1, \ldots, Q_n$. Let $p_0, p_1, \ldots$ be the prime numbers in increasing sequence. Assign to each quadruple $Q_r =< q_i, s_j, a_k, q_m >$ the number

$$G(Q_r) = p_{4r}^{1+i} p_{4r+1}^{1+j} p_{4r+2}^{1+k} p_{4r+3}^{1+m},$$

where we assume the following conventions:

1. A space containing a 0 is denoted s_0;
2. A space containing a 1 is denoted s_1;
3. A blank space is denoted s_2;
4. a_0 represents the action "print 0";
5. a_1 represents the action "print 1";
6. a_2 represents the action "move right";
7. a_3 represents the action "move left";
8. a_4 represents the action "erase square".

$G(Q_r)$ is the Gödel number of the quadruple Q_r. The Gödel number $G(T)$ of the Turing machine T is the product of all the Gödel numbers of its quadruples. Given $G(T)$ it is possible to compute what the quadruples of T are. All that is required is to calculate the prime decomposition of $G(T)$, and decode each quadruple from the indices of the relevant prime factors. Since the function $f_i(p_0^{x_0} p_1^{x_1} p_2^{x_2} \ldots) = x_i$ is a primitive recursive function, the enumeration of all the Turing machines according to ascending Gödel number is effective.

Now we can define our Universal Turing machine U as follows. Firstly, we use the standard pairing function $\bar{p}n = 1^{|p|}0pn$. This allows our Turing machine to correctly extract the numbers p and n even when it only accepts a single number as input. p gives U the number of the Turing machine T_p that U should emulate, and n is T_p's input. If p is not the Gödel number of any Turing machine, set $U(\overline{p}n) \uparrow$.

It is also possible to enumerate all the Turing machines with two, three, or any number of input values.

There are infinitely many Gödel numbers for the same computable function. Furthermore, for any computable function ϕ_x, we can find an unlimited number of indices y such that $\phi_x = \phi_y$, simply by adding some meaningless quadruples to the end of the program, and thus getting a new Gödel number. This trick is known as "padding".

Another simple, yet extremely useful trick is embodied in the *s-m-n* theorem:

Lemma 1 (The *s-m-n* theorem). *There exists a total computable function s such that for all x, y, z, $\phi_x(y, z) = \phi_{s(x,y)}(z)$, where ϕ_e is the e-th Turing machine in the standard enumeration.*

Proof. Define s as the function that on input x and y, obtains the x-th partial recursive function in the standard enumeration and runs it with input $(y, \_)$. The second number in the input is obtained from the input of $\phi_{s(x,y)}$. □

The following Lemma is another fundamental result of computability theory. For the proof of the Lemma, see the Appendix.

Lemma 2 (The Recursion Theorem, Kleene). *For every total computable function f there is a corresponding "fixed point" n such that $\phi_n = \phi_{f(n)}$, where ϕ_n is the n-th Turing machine in the standard enumeration.*

2.6 The Halting Problem

It would be nice if *every* function $f : \omega \to \omega$ could be computed on a Turing machine. However, this is not possible. Turing in his original paper [49] gave the Halting Problem as an example of something that a Turing machine cannot compute; that is, a non-computable function. The Halting Problem asks if we can work out whether a given machine halts on a given problem.

Theorem 3. *Suppose that $\phi_0, \phi_1, \ldots$ is a standard enumeration of the Turing Machines. There is no Turing Machine ϕ, such that for any natural number x,*

$$\phi(x) = \begin{cases} 1 & \textit{if } \phi_x(x) \downarrow \\ 0 & \textit{otherwise.} \end{cases}$$

Proof. Suppose for contradictive purposes that such a ϕ does exist. Then we could use ϕ as an oracle to check for each x whether $\phi_x(x)$ halts, and accordingly define the following computable function:

$$\psi(x) = \begin{cases} \phi_x(x) + 1 & \text{if } \phi_x(x) \downarrow \\ 1 & \text{if } \phi_x(x) \uparrow . \end{cases}$$

Since ψ is computable, there must be some y such that $\psi = \phi_y$ in the standard enumeration of Turing machines. We observe that by definition ψ and hence ϕ_y is a total function (i.e. halts on every input), and that, in particular, $\phi_y(y)$ halts. Thus

$$\begin{aligned} \phi_y(y) &= \psi(y) \\ &= \phi_y(y) + 1. \end{aligned}$$

Since $\phi_y(y)$ is defined, this is impossible. Hence ϕ cannot exist. $\square$

The Halting Problem can be extended to a more general form that considers the existence of a partial computable function $\widehat{\phi}$ such that for any $x, y \in \omega$:

$$\widehat{\phi}(\bar{x}y) = \begin{cases} 1 & \text{if } \phi_x(y) \downarrow \\ 0 & \text{otherwise.} \end{cases}$$

As above, it can be shown that no such $\widehat{\phi}$ can exist, as one can define $\widehat{\psi}$ via:

$$\widehat{\psi}(\bar{x}y) = \begin{cases} \phi_x(\bar{x}y) + 1 & \text{if } \phi_x(\bar{x}y) \downarrow \\ 1 & \text{if } \phi_x(\bar{x}y) \uparrow . \end{cases}$$

2.7 Computable and computably enumerable sets

Let ϕ be a Turing machine. There will be a certain set of numbers S_ϕ that ϕ successfully halts on. S_ϕ is called the *domain* of ϕ. Via their domains, Turing machines correspond to subsets of the natural numbers. A subset of ω that is the domain of a Turing machine has a special place in computability theory. We call these subsets of ω the *computably enumerable* sets:

Definition 3. *A set $A \subseteq \omega$ is computably enumerable (or c.e.) if there is a partial computable function ϕ such that $A = \mathrm{dom}\phi$.*

Let $\phi_0, \phi_1, \ldots, \phi_e, \ldots$ be an enumeration of the partial computable functions. This defines a corresponding enumeration on the c.e. sets

$$W_0, W_1, \ldots W_e, \ldots$$

where $W_e = \mathrm{dom}\phi_e$ is the e-th c.e. set. Each partial computable function (or Turing machine) can be coded by the set of numbers that it accepts. (The coding may not be unique, as many machines may have the same domains, but given a deterministic machine, the corresponding c.e. set will certainly be well-defined).

Proposition 4. *The Halting Problem can be coded by the c.e. set*

$$K = \{x : \phi_x(x) \downarrow\}.$$

Proof. Define a machine ϕ_K on all n via:

$$\phi_K(n) = \begin{cases} \phi_n(n) & \text{if } \phi_n(n) \downarrow \\ \uparrow & \text{if } \phi_n(n) \uparrow . \end{cases}$$

This is a valid machine because, given input n, ϕ_K simply mimics the behaviour of $\phi_n(n)$. Furthermore, the domain of ϕ_K is K, so K must be computably enumerable. □

Similarly, the generalized Halting Problem can be coded by the c.e. set

$$\widehat{K} = \{\bar{x}y : \phi_x(y) \downarrow\}.$$

Another way of defining c.e. sets is the following:

Lemma 5. *An (non-empty) set $A \subseteq \omega$ is computably enumerable if and only if there is a total computable function ψ such that the range of ψ is A; that is, for all $n \in A$ there is a $m \in \omega$ such that $\psi(m) = n$.*

In some ways this is conceptually easier to understand. Because this model uses a total function ψ, the c.e. set can easily be generated by calculating $\psi(0)$,$\psi(1)$,... in order. One after another the elements of A will fall out of it. One can imagine a c.e. set as a big bucket into which a number falls occasionally; we have no power over what this number will be. The function generating the set becomes unimportant.

If a number is in A, it will eventually be produced by ψ. However, there is no certainty about what number is going to appear next. Also, because the domain of ψ is infinite, one will never be able to say for sure that a number will *never* enter A.

The proof of this Lemma uses a trick known as *dovetailing*. This is useful if you want to list out all the elements that a particular partial function f halts on. Dovetailing works as follows:

> *At each stage s, run $f(0)$ for s steps, $f(1)$ for s steps, and so on, until $f(s)$ has been run for s steps.*

We denote a function f being run on input n for s steps as $f(n)[s]$. Dovetailing will enable us to find all elements of the domain of f, because if $f(n) \downarrow$, we will eventually observe this to be the case after sufficient time. In this manner, all the elements of the domain of f can be listed out. The fact that on some inputs f never halts will not be an issue.

Proof. Suppose that we have a total computable function ψ such that the range of ψ is A. Define a partial computable function ϕ via:

$$\phi(n) = (\mu s)\{\psi(s) = n\},$$

where μ is the minimization function defined in Definition 2. Then $\phi(n)$ will be defined if and only if n is in A.

Conversely, suppose we have a partial computable function ϕ such that the domain of ϕ is A. To define ψ, dovetail ϕ on all inputs:

$$\begin{array}{lllll}
\textbf{Stage 0}: & \phi(0)[0] & & & \\
\textbf{Stage 1}: & \phi(0)[1] & \phi(1)[1] & & \\
\vdots & \vdots & \vdots & & \\
\textbf{Stage } \mathrm{s}: & \phi(0)[s] & \phi(1)[s] & \dots & \phi(s)[s] \\
\vdots & \vdots & \vdots & & \vdots
\end{array}$$

If m_0 and s_0 are the first values such that $\phi(m_0)[s_0] \downarrow$, define $\psi(0) = m_0$. Similarly, for every i, if m_i and s_i are the i-th values such that $\phi(m_i)[s_i] \downarrow$, define $\psi(i) = m_i$. Then ψ is total and computable, and ψ's range is

$$\mathrm{dom}\phi = A,$$

as required. □

Suppose we wanted to find the answer to the question "is n an element of A?". If n *is* in A then we can find this out in finite time, but we can never know for sure if n is *not* in A. For this reason a computably enumerable set A is called *semi-decidable*. A nicer situation to be in would be to be able to know for sure whether or not n is in A. This property is satisfied by the class of sets that are known as the computable sets:

Definition 4. *A set $A \subseteq \omega$ is computable if both A and $\overline{A} = \omega - A$ are computably enumerable.*

Inclusion in a computable set is *decidable* - the question "is n in A?" can be answered "yes" or "no" in finite time. Because it is possible to enumerate both A and the complement of A, we can do this until n appears. Obviously, n must eventually appear in one of these. At this stage the enumeration can be stopped and the question is answered.

Chapter 3

Reducibility

Suppose that we are able to generate a set B, given another set A. This implies that the information required to know the elements of B are somehow stored in A. If a process to extract B from A exists, we say that B is *reducible* to A, written $B \leq A$. Informally, one could say that A is more complicated a set than B (or at least as complicated a set), in the sense that all information that is encoded in B is also encoded in A. There are different types of reducibilities that depend on the process used to generate B from A. In this chapter we consider two of the most common reducibilities, many-one reducibility and Turing reducibility.

3.1 m-reducibility

Definition 5. *A set B is m-reducible to A (also called* many-one *reducible), written $B \leq_m A$, if and only if there is a computable function $f : \omega \to \omega$ such that for all x,*

$$x \in B \text{ if and only if } f(x) \in A.$$

A trivial example of m-reducibility is to show that the halting problem

$$K = \{x : \phi_x(x) \downarrow\}$$

is m-reducible to the generalized halting problem

$$\widehat{K} = \{\bar{x}y : \phi_x(y)\downarrow\}.$$

This is to be expected, because obviously $\widehat{K}$ contains all the information in K (that is, whether or not $\phi_x(x)$ halts) as well as much more.

Proposition 6. *$K \leq_m \widehat{K}$.*

Proof. Note that for all x, $x \in K$ if and only if $\bar{x}x \in \widehat{K}$. So define $f(x) = \bar{x}x$, and we are done. $\square$

A c.e. set A is m-complete if for every c.e. set B, $B \leq_m A$. This implies that of all the c.e. sets, A contains the most information; in fact, A contains information about the elements of *all* the c.e. sets. Such sets do exist. For example, the generalised halting problem $\widehat{K}$ can be shown to be m-complete:

Proposition 7. *$\widehat{K} = \{\bar{x}y : \phi_x(y)\downarrow\}$ is m-complete.*

Proof. Let W_e be any computably enumerable set. Note that for any x,

$$\begin{aligned} x \in W_e \quad &\text{if and only if} \quad \phi_e(x)\downarrow \\ &\text{if and only if} \quad \bar{e}x \in \widehat{K}. \end{aligned}$$

For any e, define $f_e(x) = \bar{e}x$, and we are done. $\square$

We can also prove that the Halting Problem K is m-complete.

Proposition 8. *K is m-complete.*

Proof. Take any c.e. set W. To show that W is m-reducible to K, define the following function:

$$\psi(x,y) = \begin{cases} 1 & \text{if } x \in W \\ \uparrow & \text{otherwise.} \end{cases}$$

This function is computable because to compute $\psi(x, y)$ it is simply necessary to compute the elements of W. If we see x enter W then we halt the computation of ψ and output 1. If we never see x then ψ runs forever. The *s-m-n* theorem (Lemma 1) tells us that there is a function f such that $\phi_{f(x)}(y) = \psi(x, y)$. Now $\mathrm{dom}(\phi_{f(x)})$ is ω if $x \in W$, and $\emptyset$ if $x \notin W$. So $\phi_{f(x)}(f(x)) \downarrow$ if $x \in W$ and $\phi_{f(x)}(f(x)) \uparrow$ if $x \notin W$. Thus the following statement holds:

$$x \in W \iff f(x) \in K.$$

$\square$

3.2 Problems with m-reducibility

Although m-reducibility seems a natural way of comparing the complexity of two sets, it has some disadvantages as a type of reducibility. Recall that we would consider $B \leq A$ if A contains all the information contained in B, and hence B could be generated, if one knew A. An obvious example is that a set A contains all the information contained in its complement $\overline{A}$, as the elements of $\overline{A}$ are precisely the elements of ω that are not contained in A. If one knows the elements of A, then the elements of $\overline{A}$ are also known. However, m-reducibility does not adequately cover this scenario.

Proposition 9. *There is no non-computable c.e. set A such that*

$$\overline{A} \leq_m A.$$

Proof. Suppose that we have a c.e. set A such that $\overline{A} \leq_m A$; that is, there is a function f such that

$$n \in \overline{A} \text{ if and only if } f(n) \in A.$$

Because A is c.e. there is a Turing machine M such that

$$(\forall n \in \omega) M(n) \downarrow \text{ if and only if } n \in A.$$

Consider the function $M(f)$ formed by composing M and f, so

$$(M(f))(n) = M(f(n)).$$

From above, we know that

$$\begin{aligned}(\forall n \in \omega) M(f(n)) \downarrow \quad &\text{if and only if} \quad f(n) \in A \\ &\text{if and only if} \quad n \in \overline{A}.\end{aligned}$$

Hence $\overline{A}$ is the domain of a computable function (the composition of two computable functions) and must be a c.e. set. From this it follows that A must be a computable set, and so non-computable c.e. set can satisfy this property. We know that non-computable c.e. sets exist. For example, the proof of the Halting Problem can be used to show that K is not computable, even though it is computably enumerable. $\square$

This compels us to consider other types of reducibility. One possible alternative is Turing reducibility. Turing reducibility follows from the concept of relative computation, which we turn to now.

3.3 Relative computation

The definition of a Turing machine ϕ can be made stronger by allowing it to appeal to an *oracle*. An oracle is a set A that is somehow hardwired into ϕ. Whilst performing a computation, ϕ may ask finitely many questions of the form "is z a member of A?" We write ϕ^A to denote a Turing machine that has an oracle A installed.

Definition 6. *We call ϕ^A an A-computable function.*

Formally, an oracle set A can be hard-wired into the Turing machine via a second tape, which is read by a second, read-only head. On this tape is printed χ_A, the characteristic function of A. When the Turing machine wishes to make a request of A, it can run a sub-program for the read-only head. The head goes to the z-th bit of χ_A on the oracle tape and will change into either the "yes" state q_y if z is in A, or the "no" state q_n if it is not. Given this information, the Turing machine can continue its computation as usual.

An equivalent way of formalizing an oracle computation is to define a function to be A-computable if it can be formed from the partial computable functions with χ_A added to the basic partial computable functions.

3.4 Turing reducibility

The A-computable Turing machines can be coded in a similar way to the standard Turing machines, and hence the set of A-computable Turing machines can be effectively enumerated. We use ϕ_e^A to denote the e-th such A-computable Turing machine.

Definition 7. *A set B is Turing reducible to (or computable in) a set A, written $B \leq_T A$, if there is some total A-computable Turing machine ϕ^A such that ϕ^A decides B. That is,*

$$n \in B \Longleftrightarrow \phi^A(n) = 1.$$

Two sets A and B are Turing equivalent *if $A \leq_T B$ and $B \leq_T A$. This is written $A \equiv_T B$. The* degree *of A is written* **a** *and is defined as*

$$\mathbf{a} = \{B : B \equiv_T A\}.$$

$A \equiv_T B$ is reflexive, symmetric and transitive. (To see this is true, consider the definition of Turing reducibility). Thus Turing equivalence is an equivalence relation, and the Turing degrees form a partially ordered set where $\mathbf{a} \leq \mathbf{b}$ if and only if $A \leq_T B$.

Define a set B to be A-c.e. if there is an e such that $B = \mathrm{dom}(\phi_e^A)$. As a parallel to Lemma 5, the following lemma holds:

Lemma 10. *Let B be a (non-empty) subset of ω. B is A-computably enumerable if and only if there is a A-computable total function ψ^A such that the range of ψ^A is A; that is, for all $n \in B$ there is a $m \in \omega$ such that $\psi^A(m) = n$.*

In Chapter 2 a set was said to be computable if both it and its complement were computably enumerable. The following Lemma shows that the A-computable (or A-reducible) sets are a relativized version of this.

Lemma 11. *$B \leq_T A$ if and only if B and $\overline{B}$ are c.e. in A.*

Proof. If $B = \mathrm{dom}\phi_i^A$ and $\overline{B} = \mathrm{dom}\phi_j^A$, to decide if $n \in B$ run $\phi_i^A(n)$ and $\phi_j^A(n)$ simultaneously - by definition one of these must halt and answer the question. $\square$

3.5 Turing reducibility and m-reducibility

Turing reducibility is a more general reducibility than many-one reducibility. However, it includes m-reducibility, as everything that is m-reducible is also Turing reducible.

Proposition 12. *For all sets A, B,*

$$B \leq_m A \Rightarrow B \leq_T A.$$

Proof. Suppose that $B \leq_m A$; that is, there is a computable function f such that

$$x \in B \text{ if and only if } f(x) \in A.$$

Define a Turing machine M with oracle A that, on input x, computes $f(x)$. It then asks "is $f(x)$ in A?" If the answer is yes, $M^A(x) \downarrow= 1$. Otherwise, $M^A(x) \downarrow= 0$. M^A is an A-computable machine, and M^A decides B. Thus $B \leq_T A$, as required. $\square$

However, the converse is not true. Turing reducibility does not imply m-reducibility. We can see this to be true by finding two sets B and A such that $B \leq_T A$ but $B \not\leq_m A$. Proposition 9 shows that for any noncomputable c.e. set A, $\overline{A} \not\leq_m A$. However, $\overline{A} \leq_T A$:

Proposition 13. *For all sets A, $\overline{A} \leq_T A$.*

Proof. We will build an A-computable function ϕ^A such that ϕ^A decides $\overline{A}$. To compute $\phi^A(n)$, ask the oracle "is n in A?". Then define $\phi^A(n)$ via

$$\phi^A(n) = \begin{cases} 1 & \text{if } n \notin A \\ 0 & \text{if } n \in A. \end{cases}$$

ϕ^A is an A-computable function, and outputs the characteristic function of $\overline{A}$ as required. ☐

3.6 The Relativized Halting Problem and the Jump degrees

The Halting Problem can be relativized to show that there is no A-computable machine ϕ^A such that for any x, $\phi^A(x)$ will tell you whether $\phi_x^A(x)$ halts or not. This problem can be coded by the set

$$K^A = \{x : \phi_x^A(x) \downarrow\}.$$

For any A, K^A is known as the "jump" of A and written A'. More generally, $A^{(n)}$ is known as the n-th jump of A, and is recursively defined by:

$$\begin{aligned} A^{(0)} &= A \\ A^{(n+1)} &= (A^{(n)})'. \end{aligned}$$

The degree of A' is written $\mathbf{a}'$. The unrelativized halting problem K is the jump of the computable sets and is written $K^{\emptyset}$ or $\emptyset'$. As a special case,

$$\begin{aligned} deg(\emptyset) &= \mathbf{0} \\ deg(K) &= \mathbf{0}' \\ &\vdots \\ deg(\emptyset^{(n)}) &= \mathbf{0}^{(n)}. \end{aligned}$$

Proposition 14.

i. *K^A is not computable in A.*

ii. *K^A is A-c.e.*

iii. *$A \leq_T A'$.*

iv. *If A is c.e. in B and $B \leq_T C$, then A is c.e. in C.*

v. *A set B is c.e. in A if and only if B is c.e. in $\overline{A}$.*

vi. *B is A-c.e. if and only if $B \leq_m A'$.*

vii. *The jump is well defined on degrees. That is, if*

$$deg(A) = deg(B),$$

then

$$deg(A') = deg(B').$$

Proof.

i. This is a relativization of Theorem 3.

ii. This is a relativization of Proposition 4.

iii. A relativization of Proposition 8 shows that $A \leq_m A'$. (It is a simple matter to show that the *s-m-n* theorem can be relativized). Thus, from Proposition 12 we may conclude that $A \leq_T A'$.

iv. Suppose that A is the range of a B-computable function ϕ_i^B. ϕ_i^B can be converted into a C-computable function by attaching C as an oracle, and then running ϕ_i^B. Each time ϕ_i^B queries if $b \in B$, we can use C to decide whether or not this is true. Hence A is the range of a C-computable function, and so A is c.e. in C.

v. Suppose B is c.e. in A. Since $A \leq_T \overline{A}$ by Proposition 13, it follows from (iv) that B is c.e. in $\overline{A}$. The converse is identical.

vi. Suppose that B is A-c.e. Relativizing Proposition 8 shows that

$$B \leq_m A'.$$

Conversely, if $B \leq_m A'$, this implies that there is a computable function f such that

$$\begin{array}{rll} x \in B & \text{if and only if} & f(x) \in A' \\ & \text{if and only if} & \phi^A_{f(x)}(f(x)) \downarrow . \end{array}$$

Define an A-computable function ψ^A by for all x,

$$\psi^A(x) = \phi^A_{f(x)}(f(x)).$$

Then $\mathrm{dom}(\phi^A) = B$, so B is A-c.e.

vii. Suppose that $deg(A) = deg(B)$. Then

$$\begin{array}{lll} & B' \text{ is } B\text{-c.e.} & \text{by (ii)} \\ \text{and} & B \leq_T A & \text{as } deg(A) = deg(B) \\ \Rightarrow & B' \text{ is } A\text{-c.e.} & \text{by (iv)} \\ \Rightarrow & B' \leq_m A' & \text{by (vi)} \\ \Rightarrow & B' \leq_T A' & \text{by Proposition 12.} \end{array}$$

The proof of $A' \leq_T B'$ is identical.

□

This means that for any A, the jump degrees $\mathbf{a}^{(n)}$ form an infinite hierarchy

$$\mathbf{a} < \mathbf{a}' < \mathbf{a}'' \ldots$$

Of particular interest is the hierarchy of the $\emptyset$-computable degrees

$$\mathbf{0} < \mathbf{0}' < \mathbf{0}'' \ldots,$$

which we will look at more in Chapter 4.

3.7 Other reducibilities

It is a truism that, since any Turing reducibility from a set A may only make finitely many requests of A, for any A-computable machine ϕ^A and for any n, there must be an $m \in \omega$ such that ϕ only asks about the first m elements of A, and so $\phi^A(n) = \phi^{A \upharpoonright m}(n)$. m is known as the *use* of ϕ^A on n. Placing restrictions on the size of the use creates stronger versions of Turing reducibility. We state here two adaptations of Turing reducibility that will be of interest later.

Definition 8 (Friedberg and Rogers, [21], Downey, Hirschfeldt and Laforte, [16]).

- *A set B is* weak truth-table reducible *to a set A, written $B \leq_{wtt} A$ if $B \leq_T A$ via some A-computable machine ϕ^A, and there is a computable function $f : \omega \to \omega$ such that the use of $\phi^A(n)$ is bounded by $f(n)$.*

- *A set B is* strongly weak truth-table reducible *to a set A, written $B \leq_{sw} A$, if $B \leq_{wtt} A$ (as above) and there is a constant c such that for all n the use of $\phi^A(n)$ is bounded by $n + c$.*

Another type of reducibility is the following:

Definition 9. *A set B is* truth-table reducible *to a set A, written $B \leq_{tt} A$, if there is a computable sequence of functions $\{F_i : i \in \omega\}$ such that for all i, $F_i : 2^{r_i} \to \{0, 1\}$ and*

$$B[i] = F_i(A \upharpoonright r_i).$$

Chapter 4

The Arithmetic Hierarchy

Suppose that S is a c.e. subset of ω. Recall that this means that there is a Turing machine ϕ such that for every $x \in S$, there is some stage s, such that $\phi(x)[s] \downarrow$. S can be described by:

$$S = \{x : (\exists s)\phi(x)[s] \downarrow\}.$$

The answer to the question "does $\phi(x)$ halt at stage s?" can be answered yes or no immediately - one can simply run $\phi(x)$ for s steps and see whether it in fact halts. It is computable, in contrast to the question "does $\phi(x)$ halt?", which can be answered yes, but cannot with certainty ever be answered no.

From the above discussion we can conclude that if S is a c.e. set, then there is a computable relation, namely

$$R_S(x, s) = \begin{cases} 1 & \text{if } \phi(x)[s] \downarrow \\ 0 & \text{otherwise} \end{cases}$$

such that $S = \{x : (\exists s)R(x, s)\}$. It turns out that the converse holds as well:

Theorem 15. *A set S is computably enumerable if and only if there is a computable relation $R(x, s)$ such that $S = \{x : (\exists s)R(x, s)\}$.*

Proof. As discussed above, if S is c.e. then there is an associated computable relation. Conversely, suppose that $S = \{x : (\exists s)R(x, s)\}$ where R is computable. Define a partial computable function $\psi = (\mu s)R(x, s)$. Then $S = \mathrm{dom}\psi$ as required. $\square$

In the arithmetic hierarchy, the c.e. sets are known as Σ^0_1 sets, and are a special case of the Σ^0_n sets. The Σ^0_n sets, along with a related concept (the Π^0_n sets), are defined as follows:

Definition 10. *A set S is Σ^0_n (or $S \in \Sigma^0_n$) if*

$$S = \{x : \underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations} R(x, x_1, \ldots, x_n)\},$$

where R is an $n+1$-ary computable relation and

$$\underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations}$$

implies that there are n quantifiers, alternating between $\forall$ and $\exists$, and where the jth quantifier is binding the instances of x_j in R. Similarly, a set S is Π^0_n if

$$S = \{x : \underbrace{(\forall x_1)(\exists x_2)(\forall x_3)\ldots}_{n\ alternations} R(x, x_1, \ldots, x_n)\}.$$

If a set is both Σ^0_n and Π^0_n we say it is Δ^0_n.

Notice that the quantifiers are alternating between universal and existential quantifiers. Unbroken sequences of quantifiers of the same flavour can be condensed down into a single quantifer. For example, the relation $(\exists y_1)(\exists y_2)R(y_1, y_2)$ can be replaced by the equivalent relation $(\exists y)R'(y)$ where if $z = p_1^{z_1}p_2^{z_2}$ is the standard Gödel number coding z_1 and z_2 then $R'(z) \equiv_{def} R(z_1, z_2)$. This of course works for sequences of any (finite) number of similar quantifiers and so two or more similar quantifiers are simply counted as one.

The purpose of this chapter is to cover some simple but important facts about the Arithmetic Hierarchy which will be useful in later chapters.

Proposition 16. *i. A set A is in Σ^0_n if and only if $\overline{A}$ is in Π^0_n, where $\overline{A}$ is the complement of A.*

ii. If A and B are in Σ^0_n, then $A \cap B$ and $A \cup B$ are in Σ^0_n. Similarly, if A and B are in Π^0_n, then $A \cap B$ and $A \cup B$ are in Π^0_n.

iii. If $p \geq 2$, and $A_1, A_2, \ldots, A_p$ are Σ^0_n sets, then $\bigcap_{i \leq p} A_i$ and $\bigcup_{i \leq p} A_i$ are in Σ^0_n. Similarly, finite unions and intersections of Π_n sets are also Π^0_n.

iv. Let $R(x, y, z)$ be a Σ^0_n relation. If

$$x \in A \iff (\forall y < z) R(x, y, z)$$

and

$$x \in B \iff (\exists y < z) R(x, y, z)$$

then A and B are Σ^0_n sets.

Similarly if $R(x, y, z)$ is a Π^0_n relation, A and B are Π^0_n sets.

v. A Π^0_{n+1} set can be represented as the intersection of countably infinite Σ^0_n sets. Similarly, a Σ^0_{n+1} set can be represented as the union of countably infinite Π^0_n sets.

vi. If $A \in \Sigma^0_n$, then for all $m > n$, $A \in \Sigma^0_m$ and $A \in \Pi^0_m$. This is true if $A \in \Pi^0_n$ as well.

Proof. i. Let $A \in \Sigma^0_n$. By the definition of the complement of A,

$$x \in A \iff x \notin \overline{A}.$$

Thus

$$\underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations} R(x, x_1, \ldots, x_n) \iff x \notin \overline{A},$$

so obviously

$$x \in \overline{A} \iff \neg \underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations} R(x, x_1, \ldots, x_n).$$

By the rules of sentential logic this becomes

$$x \in \overline{A} \iff \underbrace{(\forall x_1)(\exists x_2)(\forall x_3)\ldots}_{n\ alternations} \neg R(x, x_1, \ldots, x_n)\}.$$

Since $\neg R(x, x_1, \ldots, x_n)\}$ must be a computable relation (as $R(x, x_1, \ldots, x_n)$ is computable) it follows that $\overline{A}$ must be Π^0_n.

ii. Let

$$A = \{x : (\exists y_1)(\forall y_2)\ldots R(x, y_1, y_2, \ldots, y_n)\}$$

and

$$B = \{x : (\exists z_1)(\forall z_2)\ldots S(x, z_1, z_2, \ldots, z_n)\}.$$

Then

$$\begin{aligned} x \in A \cup B &\iff (\exists y_1)(\forall y_2)\ldots R(x, y_1, \ldots, y_n) \\ &\qquad \vee (\exists z_1)(\forall z_2)\ldots S(x, z_1, \ldots, z_n) \\ &\iff (\exists y_1)(\exists z_1)(\forall y_2)(\forall z_2)\ldots \\ &\qquad [R(x, y_1, \ldots, y_n) \vee S(x, z_1, \ldots, z_n)]. \end{aligned}$$

Since R doesn't contain any of the variables $z_1, \ldots, z_n$ and S doesn't contain any of the variables $y_1, \ldots, y_n$ it is quite safe to allow the unrelated quantifiers to bind them. By the rules of quantifier contraction as discussed above this can be reduced to

$$x \in A \cup B \iff (\exists x_1)(\forall x_2)\ldots[R'(x, x_1, \ldots, x_n) \vee S'(x, x_1, \ldots, x_n)].$$

Since the disjunction of two computable relations is a computable relation, this means that $A \cup B$ is a Σ^0_n set as well. The other claims follow identically.

iii. From (ii).

iv. From (ii).

v. If P is a Π^0_{n+1} set then there is an $n+2$-ary computable relation R such that

$$x \in P \iff \underbrace{(\forall x_1)(\exists x_2)(\forall x_3)\ldots}_{n+1\ alternations} R(x, x_1, \ldots, x_n, x_{n+1}).$$

Define for all $k \in \omega$, a new $n+1$-ary relation R_k, defined by

$$R_k(x, x_1, \ldots, x_n) = R(x, k, x_2, \ldots, x_n, x_{n+1}).$$

Define for each k a Σ^0_n set U_k such that

$$x \in U_k \iff \underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations} R_k(x, x_1, \ldots, x_n).$$

Then

$$\begin{aligned} x \in \cap_k U_k \quad &\iff \quad (\forall k)\, \underbrace{(\exists x_1)(\forall x_2)(\exists x_3)\ldots}_{n\ alternations} R_k(x, x_1, \ldots, x_n) \\ &\iff \quad (\forall k)\, \underbrace{(\exists x_2)(\forall x_3)(\exists x_4)\ldots}_{n\ alternations} R(x, k, x_2 \ldots, x_n, x_{n+1}) \\ &\iff \quad x \in P. \end{aligned}$$

A similar proof works for the Σ^0_{n+1} sets.

vi. As an example, suppose that $A \in \Sigma^0_2$. There is a computable relation R such that $A = \{x : (\exists y_1)(\forall y_2)R(x, y_1, y_2)\}$. A will be in any class requiring more quantifiers than this, because any number of dummy quantifiers can be added in. For example,

$$A = \{x : (\exists y_1)(\forall y_2)(\exists y_3)[R(x, y_1, y_2) \wedge y_3 = y_3]\},$$

so $A \in \Sigma^0_3$. This argument can be extended to any number (or type) of dummy quantifiers.

□

It is possible to consider the arithmetic hierarchy relative to a set A. For example, a set B is Σ^A_1 if there is an A-computable relation $R^A(x, y)$ such that

$$x \in B \iff (\exists y)R^A(x, y).$$

Lemma 17. *A set B is A-c.e. if and only if B is Σ_1^A.*

Proof. As for Theorem 15 with "computable" replaced by "A-computable" and "computably enumerable" replaced by "A-computably enumerable". ☐

4.1 Post's Theorem

At the beginning of this chapter it was shown that the c.e. sets were equivalent to the Σ_1^0 sets. Work originally done by E. Post and S. Kleene shows that this equivalence continues up the rest of the hierarchy. In other words, for any n the $\emptyset^{(n)}$-c.e. sets are equivalent to the Σ_{n+1}^0 sets. Furthermore, the $\emptyset^{(n)}$-computable sets are equivalent to the Δ_{n+1}^0 sets.

Recall that a c.e. set (i.e. a Σ_1^0 set) A is called m-complete if any c.e. set B is m-reducible to A. This definition can be generalized so that for any n, a Σ_n^0 set A is called Σ_n^0-complete if any Σ_n^0 set B is m-reducible to A.

Theorem 18. *i. $B \in \Sigma_{n+1}^0$ if and only if B is A-c.e. for some $A \in \Sigma_n^0$.*

ii. For any $n > 0$, $\emptyset^{(n)}$ is Σ_n^0-complete.

iii. $B \in \Sigma_{n+1}^0$ if and only if B is c.e. in $\emptyset^{(n)}$.

iv. $B \in \Delta_{n+1}^0$ if and only if B is computable in $\emptyset^{(n)}$ (that is, $B \leq_T \emptyset^{(n)}$).

v. $B \in \Delta_{n+1}^0$ if and only if B is A-computable for some $A \in \Pi_n^0$.

vi. If $A \in \Pi_n^0$, then $A \leq_T \emptyset^{(n)}$. Similarly if $A \in \Sigma_n^0$.

vii. For any $n \geq 1$, it is possible to enumerate the Σ_n^0 sets.

Proof. i. Suppose that B is a Σ_{n+1}^0 set, so

$$x \in B \iff (\exists y)R(x, y)$$

where $R(x, y)$ is a Π_n^0 relation. Thus B is Σ_1^A for some Π_n^0 set A, and so B is c.e. in A by Lemma 17. By Proposition 14(v), B is c.e. in $\overline{A}$, a

Σ^0_{n+1} set, as required. Conversely, suppose that B is c.e. in a Σ^0_n set. Then B is also c.e. in a Π^0_n set A (by Proposition 14(v)). That is, there is an i such that

$$x \in B \iff \phi_i^A(x) \downarrow,$$

or more precisely,

$$x \in B \iff (\exists s)\phi_i^A(x)[s] \downarrow .$$

$\phi_i^A(x)[s] \downarrow$ is computable, apart from a finite number of questions about individual elements of A, which is a Π^0_n set. Hence $\phi_i^A(x)[s] \downarrow$ represents a Π^0_n relation, and so B is a Σ^0_{n+1} set.

ii. This is proved via induction on n.

Base case: Let $n = 1$. Now $\emptyset' = K$, which we have proved earlier to be c.e. and m-complete.

Induction step: Assume that $\emptyset^{(k)}$ is Σ^0_k-complete. Consider $\emptyset^{(k+1)}$. Theorem 14(ii) ensures that $\emptyset^{(k+1)}$ is c.e. in $\emptyset^{(k)}$. Hence

$$\emptyset^{(k+1)} \in \Sigma^0_{k+1}$$

by (i). Let $B \in \Sigma^0_{k+1}$.

$$\begin{array}{rcll} B \in \Sigma^0_{k+1} & \iff & B \text{ is c.e. in some } \Sigma^0_{n+1} \text{ set} & \text{by (i)} \\ & \implies & B \text{ is c.e. in } \emptyset^{(k)} & \text{by Proposition 14(iv) as} \\ & & & \emptyset^{(k)} \text{ is } \Sigma^0_k\text{-complete} \\ & \implies & B \leq_m \emptyset^{(k+1)} & \text{by Proposition 14(vi).} \end{array}$$

Thus $\emptyset^{(k+1)}$ is Σ^0_{k+1}-complete.

iii. Suppose that B is c.e. in $\emptyset^{(n)}$. Since $\emptyset^{(n)}$ is a Σ^0_n set (from (ii)) it follows from (i) that $B \in \Sigma^0_{n+1}$.

Conversely, suppose that $B \in \Sigma^0_{n+1}$. Then (i) implies that B' is c.e. in some Σ^0_n set A. Because $\emptyset^{(n)}$ is Σ^0_n-complete, $A \leq_m \emptyset^{(n)}$, and thus $A \leq_T \emptyset^{(n)}$. Applying Proposition 14(iv) to this ensures that B is c.e. in $\emptyset^{(n)}$.

iv. This follows from (iii) along with Lemma 11:

$$\begin{aligned} B \in \Delta^0_{n+1} &\iff B \in \Sigma^0_{n+1} \text{ and } B \in \Pi^0_{n+1} \\ &\iff B \text{ and } \overline{B} \in \Sigma^0_{n+1} \\ &\iff B \text{ and } \overline{B} \text{ are } \emptyset^{(n)}\text{-c.e. (from (iii))} \\ &\iff B \leq_T \emptyset^{(n)} \text{ (from Lemma 11).} \end{aligned}$$

v. Let A be Π^0_n, and B be A-computable.

$$\begin{aligned} B \text{ is } A\text{-computable} &\iff B \text{ and } \overline{B} \text{ are } A\text{-c.e. (from Theorem 11)} \\ &\iff B \in \Sigma^0_{n+1} \text{ and } \overline{B} \in \Sigma^0_{n+1} \text{ (from (i))} \\ &\iff B \in \Sigma^0_{n+1} \text{ and } B \in \Pi^0_{n+1} \text{ (from Proposition 16(i))} \\ &\iff B \in \Delta^0_{n+1}. \end{aligned}$$

vi.

$$\begin{aligned} A \in \Pi^0_n &\implies A \in \Sigma^0_{n+1} \text{ and } A \in \Pi^0_{n+1} \text{ (from Proposition 16(vi))} \\ &\implies A \in \Delta^0_{n+1} \\ &\implies A \leq_T \emptyset^{(n)} \text{ (from Corollary iv.)} \end{aligned}$$

The proof when $A \in \Sigma^0_n$ is identical.

vii. As mentioned in Chapter 3, it is possible to enumerate the A-computable functions. This gives us a natural enumeration on the A-c.e. sets. As a special case of this, the $\emptyset^{(n-1)}$-c.e. sets can be enumerated. Then (iii) states that these sets are precisely the Σ^0_n sets, as required.

□

4.2 Measures of sets

Theorem 19 (Kurtz, [28]). *i. If S is a Σ^0_n set and $\epsilon \in \mathbb{Q}$, $\mu(S) > \epsilon$ is a Σ^0_n predicate. If T is a Π^0_n set, $\mu(T) < \epsilon$ is a Σ^0_n predicate.*

ii. *If S is a Σ^0_n set and $\epsilon \in \mathbb{Q}$, $\mu(S) \leq \epsilon$ is a Π^0_n predicate. If T is a Π^0_n set then $\mu(T) \geq \epsilon$ is a Π^0_n predicate.*

iii. *If $\{S_i : i \in \omega\}$ is a sequence of Σ^0_n sets and $q \in \mathbb{Q}$, $\sum_i \mu(S_i) \leq q$ is a Π^0_n predicate.*

iv. *If $\{T_i : i \in \omega\}$ is a sequence of Π^0_n sets then $\sum_{i\in\omega} \mu(T_i) \leq q$ is a Π^0_{n+1} predicate.*

v. *Let $\{S_i : i \in \omega\}$ be a sequence of Σ^0_n sets and $q \in \mathbb{Q}$. Suppose that $\sum_i \mu(S_i) < \infty$. The function $(\mu m)\left[\sum_{i\geq m} \mu(S_i) \leq q\right]$ is a $\emptyset^{(n)}$-computable function.*

vi. *Let $\{T_i : i \in \omega\}$ be a sequence of Π^0_n sets and $q \in \mathbb{Q}$. Suppose that $\sum_i \mu(T_i) < \infty$. The function $(\mu m)\left[\sum_{i\geq m} \mu(S_i) \leq q\right]$ is a $\emptyset^{(n+1)}$-computable function.*

Proof. i. The proof is via induction on n.

Base case: Let $n = 1$. Suppose that S is a c.e. set. Let $S[s]$ represent the elements of S at stage s. The measure $\mu(S[s])$ can be computed (roughly speaking, it is the sum of the measures of a prefix-free representation of $S[s]$). Since S is a c.e. set, $\mu(S[s])$ is a non-decreasing function of s. Thus for any rational ϵ,

$$\mu(S) > \epsilon \text{ if and only if } (\exists s)\mu(S[s]) > \epsilon.$$

Since $\mu(S[s]) > \epsilon$ is known to be a computable predicate, this means that $\mu(S) > \epsilon$ is a Σ^0_1 predicate.

If T is a co-c.e. set, then

$$\mu(T) < \epsilon \iff \mu(\overline{T}) > 1 - \epsilon.$$

Since the right-hand side of this equivalence is Σ^0_1 (from above), it follows that the left-hand side is Σ^0_1.

Induction step: Assume that the result holds for $n = k$. Consider a Σ^0_{k+1} set S. Obviously $\overline{S}$ must be a Π^0_{k+1} set, and thus can be written as the intersection of a sequence of Σ^0_k sets: $\overline{S} = \cap_j T_j$. Define $\widehat{T}_m = \cap_{j \le m} T_j$. Since $\lim_{m\to\infty} \widehat{T}_m = \overline{S}$, it follows that

$$\begin{aligned}\lim_{m\to\infty} \mu(\widehat{T}_m) &= \mu(\overline{S}) \\ &= 1 - \mu(S).\end{aligned}$$

Thus

$$\begin{aligned}\mu(S) > \epsilon &\iff \lim_{m\to\infty} \mu(\widehat{T}_m) < 1 - \epsilon \\ &\iff (\exists m)\mu(\widehat{T}_m) < 1 - \epsilon.\end{aligned}$$

Since $\widehat{T}_m$ consists of a finite intersection of Σ^0_k sets, it will also be a Σ^0_k set. Now by the induction hypothesis, since $\widehat{T}_m$ is a Σ^0_k set, the predicate $\mu(\widehat{T}_m) > 1 - \epsilon$ is a Σ^0_k predicate and hence that its complement (namely, $\mu(\widehat{T}_m) < 1 - \epsilon$) is Π^0_k. This implies that $\mu(S) > \epsilon$ is Σ^0_{k+1}.

Similarly, if T is Π^0_{k+1}, it can be written as the intersection of infinitely many Σ^0_k sets, $T = \cap_j S_j$. Let $\widehat{S}_m = \cap_{j \le m} S_j$. Since $\lim_{m\to\infty} \widehat{S}_m = T$ it follows that $\lim_{m\to\infty} \mu(\widehat{S}_m) = \mu(T)$. Since $\mu(\widehat{S}_m)$ is a non-increasing function of m,

$$\begin{aligned}\mu(T) < \epsilon &\iff \lim_{m\to\infty} \mu(\widehat{S}_m) < \epsilon \\ &\iff (\exists m)\mu(\widehat{S}_m) < \epsilon.\end{aligned}$$

As above, $\mu(\widehat{S}_m) < \epsilon$ is a Π^0_k predicate, and so $\mu(T) < \epsilon$ is a Σ^0_{k+1} predicate.

ii. Obviously,

$$\mu(S) \le \epsilon \iff \neg\mu(S) > \epsilon$$

and so the predicate $\mu(S) \le \epsilon$ is Π^0_n. Similarly for T.

iii. Consider

$$\sum_i \mu(S_i) > q \iff (\exists r_0)\ldots(\exists r_t)(r_0+\ldots+r_t > q \wedge \mu(S_0) > r_0 \wedge \ldots \wedge \mu(S_t) > r_t).$$

(i) combined with quantifier contraction on the right-hand side of this equation implies that the predicate on the left-hand side is Σ^0_n. Hence its complement, $\sum_i \mu(S_i) \leq q$, is Π^0_n.

iv. The result follows from the observation that

$$\sum_{i\in\omega} \mu(T_i) > q$$

is a Σ^0_{n+1} predicate. The quantifier contraction used in (iii) doesn't work in this case.

v. Define sets A, B via

$$x \in A \iff (\forall j < x)\Big[\sum_{i\geq j} \mu(S_i) > q\Big]$$
$$x \in B \iff \sum_{i\geq x} \mu(S_i) \leq m.$$

From (iii) and Proposition 16(iv), $A \in \Sigma^0_n$ and $B \in \Pi^0_n$, thus from Theorem 18(vi) we know that $A, B \leq_T \emptyset^{(n)}$. Thus we can define a $\emptyset^{(n)}$-computable function that finds the least m such that $m \in A \cap B$. (As it happens, the construction ensures that there is exactly one m in $A \cap B$, but this doesn't matter for our purposes).

vi. As for (v).

□

Chapter 5

Sets and measure

In this chapter we look at some results on $\mathbb{R}$ that will be useful later in this work. We look at the Lebesgue measure as a means of studying the size of subsets of $\mathbb{R}$, and also look at how the resources of computability theory can be used to deal with real numbers instead of just natural numbers. Because real numbers may not be able to be described in a finite way, computers can at best approximate them. In this way, the real numbers are similar to the infinite sets discussed in Chapter 2.

5.1 The rational and real number fields

The field of real numbers comprises the rational and the irrational numbers. The rational numbers can be created from the natural numbers as follows:

1. Start with the natural numbers $\omega = \{0, 1, 2, 3, \ldots\}$.

2. From the natural numbers we get the integers $\mathbb{Z}$ by including the negative numbers,

$$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3 \ldots\}.$$

Note that for every $x \in \mathbb{Z}$ there is a $y \in \mathbb{Z}$ such that $x + y = 0$. (This y is equal to $-x$ and is called the *additive inverse* of x).

3. From the integers we get the field of rational numbers $\mathbb{Q}$ where

$$\mathbb{Q} \equiv \{\frac{x}{y} : x, y \in \mathbb{Z}, y \neq 0\}.$$

In addition to every rational numbers having an additive inverse, for every non-zero $q \in \mathbb{Q}$ there is a $p \in \mathbb{Q}$ such that $qp = 1$. (This p is equal to $\frac{1}{q}$ and is called the *multiplicative inverse* of q.)

It can be shown that there are real numbers that are not rational (so can't be expressed as a quotient of integers $\frac{x}{y}$). For example, it has been known that $\sqrt{2}$ is irrational for a very long time. Other examples of irrational numbers include π, and e.

Most often, the real numbers are expressed in decimal form. The rational reals will either have a finite decimal expansion, or an infinite expansion that is simply a repetition of a finite sequence, e.g.

$$\frac{1}{7} = 0.\underbrace{142857}142857\ldots.$$

Conversely, irrational numbers are infinite sequences that are not simply made up of a finite repeating block.

5.2 Open and closed intervals

A metric space M is a set of points endowed with a function $\rho(x, y)$ called a *metric*. The metric measures the distance between two points of M. For example, the reals are a metric space with a metric $|x - y|$.

For a given point x, we can talk about the *neighbourhood* of x. This is the set of sets that are in close proximity to x. We can choose precisely how close the points have to be to x. For example, we may choose a δ-neighbourhood, where y is in the neighbourhood of x if $|x - y| < \delta$. A set

E is *open* if for every point $x \in E$ there is some δ-neighbourhood of x that is contained in E. If the complement of a set E is open, then E is *closed*. In particular, among the intervals of $\mathbb{R}$, $(a, b) = \{x : a < x < b\}$ is open in $\mathbb{R}$ and $[a, b] = \{x : a \leq x \leq b\}$ is closed.

5.3 The interval $[0, 1)$

We will restrict our interest to the reals modulo 1, defined as follows:

Definition 11. *Let a be a real number with decimal expansion $a = a_0.a_1a_2a_3\ldots$, where $a_0 \in \mathbb{Z}$ and $a_1, a_2, a_3, \ldots \in \{0, 1, \ldots, 9\}$. We define $a(mod1)$ as the decimal tail of a. That is,*

$$a(mod1) = 0.a_1a_2a_3\ldots.$$

Note that $a(mod1) \in [0, 1)$.

From now on we will assume that any real number is modulo 1. Hence all following work on reals will focus exclusively on the interval $[0, 1)$.

5.4 Lebesgue measure

Let $X \subseteq [0, 1)$. If X is an interval (i.e. $X = \{x : a < x < b\}$ for some real numbers $a < b$) then the size of X is the distance between the endpoints of X: $|X| = |b - a|$. However, we would like to some way of measuring the size X where X is *any* subset of the unit interval. One of the most common measures on $\mathbb{R}$ is the *Lebesgue measure*. The Lebesgue measure uses the known size of the intervals of $\mathbb{R}$ to impose a concept of measure on a general set. This general set (let us call it A) can be approximated by covering it with a set of intervals. The Lebesgue measure of A then depends on the measure of the intervals needed to cover it.

Definition 12 (Lebesgue, [29]). *A (finite or infinite) sequence of intervals I_i are said to cover A if $A \subseteq \bigcup_i I_i$. The* outer Lebesgue measure *of A is written*

$m^*(A)$ and is defined by:

$$m^*(A) = \inf\{\sum_i |I_i| : A \subset \bigcup_i I_i\}.$$

If $A \subseteq [0,1)$, we also define the inner Lebesgue measure *of A to be $1 - m^*(\overline{A})$. If the inner and outer Lebesgue measure of A are equal, we say that A is* Lebesgue measurable *and define the Lebesgue measure of A (written $\mu(A)$) to be the common value.*

Any interval (a, b) is Lebesgue measurable and has

$$\mu((a, b)) = |b - a|.$$

Note that the Greek letter μ is used for both the Lebesgue measure and for the Kleene minimisation function. In places μ appears with two different meanings in the same equation. In order to keep confusion to a minimum, we note that in the vast majority of cases, μ represents the measure and will appear as $\mu(A)$ where A is a set. This use is different from the use as a minimisation function where it appears as $(\mu x)f(x)$ where x is a string or number.

An important feature of sets when we are dealing with random numbers relates to finding very small sets. These are called the *measure zero* sets:

Definition 13. *A set A is called a* measure zero *set or a* null set *if for all $\epsilon > 0$ there exists a sequence of intervals $\{I_n\}$ such that $A \subset I_n$ and $\sum_n |I_n| < \epsilon$.*

For example, the set of rational numbers $\mathbb{Q}$ has measure zero. In fact, any countable set is a measure zero set.

Proposition 20 (First Borel-Cantelli Lemma). *Let $\{X_i : i \in \omega\}$ be a countable set of Lebesgue measurable sets such that*

$$\sum_i \mu(X_i) < \infty.$$

Then the set of elements that are in infinitely many of these X_i has measure 0.

Before proving this we require the following Lemma. For the proof of this Lemma, see the Appendix.

Lemma 21. *Let $\{X_i : i \in \omega\}$ be defined as in Proposition 20. Then for any $\epsilon \in \mathbb{Q}^+$ there is a $k \in \omega$ such that*

$$\sum_{i \geq k} \mu(X_i) < \epsilon.$$

We now turn to the proof of Proposition 20.

Proof. Let $C = \{x : (\exists^\infty i)(x \in X_i)\}$. Lemma 21 ensures us that for any j there is an $n_j \in \omega$ such that

$$\sum_{i \geq n_j} \mu(X_i) < 2^{-j}.$$

Since for any j,

$$C \subseteq \bigcup_{i \geq n_j} X_i,$$

it follows that

$$\begin{aligned} \mu(C) &\leq \mu(\bigcup_{i \geq n_j} X_i) \\ &\leq \sum_{i \geq n_j} \mu(X_i) \\ &< 2^{-j}. \end{aligned}$$

Thus C is a measure zero set. □

The following is a useful result about the closed intervals of $\mathbb{R}$:

Lemma 22. *If $\{I_n : n \in \omega\}$ is a sequence of nested non-empty closed intervals such that $\lim_n \mu(I_n)$, then there is exactly one point which is in all I_n.*

Proof. For each n, $I_n = [a_n, b_n]$ where

$$a_n \leq a_{n+1} \leq b_{n+1} \leq b_n.$$

Consider the sequences $\{a_n : n \in \omega\}$ and $\{b_n : n \in \omega\}$. The first sequence is non-decreasing and is bounded above (by b_1, say). The second sequence is non-increasing and is bounded below (by a_1). Thus each sequence must have a limit. Since $\lim_n(b_n - a_n) = 0$ it follows that

$$\lim_n b_n - \lim_n a_n = 0$$

and so

$$\lim_{n\to\infty} b_n = \lim_{n\to\infty} a_n = L,$$

say. Obviously for all n, $a_n \leq L$ and $b_n \geq L$, so $L \in I_n$. Thus $\mathcal{I} = \bigcap_n I_n$ is non-empty, as $L \in \mathcal{I}$. To prove that L is unique, suppose that has two distinct elements L and M. Then $\mu(\mathcal{I}) \geq |L - M|$, where $|L - M|$ is the standard metric. Since for all n, $\mathcal{I} \subset I_n$, this means that $\mu(I_n) \geq |L - M|$. But by taking $\epsilon = |L - M|$, it can be seen that this implies that

$$\begin{aligned} \lim_n(b_n - a_n) &\geq \epsilon \\ &\neq 0, \end{aligned}$$

which is a contradiction. □

This theorem can be extended to any nested sequence of closed and bounded subsets of the reals. The proof requires the use of the Heine-Borel Theorem.

5.5 Cantor space

Another space of importance is the Cantor space 2^ω of all infinite binary sequences. Any real in $[0, 1)$ can be associated with a binary sequence $x = x_1x_2x_3\ldots \in 2^\omega$ via the function $\lambda : 2^\omega \to [0, 1)$ where

$$\lambda(x) = x_1 2^{-1} + x_2 2^{-2} + x_3 2^{-3} \ldots.$$

Note that $\lambda(1111\ldots)$ is actually equal to 1.0. However, in the spirit of only considering the reals modulo 1, we will assume $\lambda(1111\ldots) = 0$. Note that

λ is not a one-to-one function. For example,

$$\lambda(10000\ldots) = \lambda(01111\ldots) = 0.5.$$

However, if a real is irrational, the associated binary sequence is unique.

The association between the elements of the Cantor space and $(0,1]$ will be frequently used. In fact, throughout most of the following work on the randomness of reals we will view the reals as infinite binary strings.

The basic intervals in 2^ω are defined as follows:

Definition 14. *Given $\sigma \in 2^{<\omega}$, define $[\sigma] \subseteq 2^\omega$ by*

$$[\sigma] = \{x \in 2^\omega : \sigma \sqsubset x\}.$$

The measure of $[\sigma]$ is $\mu([\sigma]) = 2^{-|\sigma|}$. Note that the set

$$\{\lambda(x) : x \in [\sigma]\} = [\lambda(\sigma 0^\omega), \lambda(\sigma 1^\omega)].$$

5.6 Computable reals

Consider how a computer might deal with an (irrational) real number. Because an irrational number cannot be coded in a finite way, it is impossible that a computer could have an exact representation of it. The best it could do would be to approximate it, say by a rational number. As time went on the computer might be able to do a bit more calculation and output another rational number that was a somewhat better approximation, and so on.

Suppose we wanted a real that was computable - as an analogue to our concept of computable sets. Recall that given a computable set, it is possible to state with complete confidence whether or not a given element is in the set. With computable reals, we would like the same level of confidence about how good our approximation is. That is, no matter how accurate we want our rational approximation to be, we can find a rational that approximates the real to the given degree of precision. This is formalised in the following definition:

Definition 15 (Turing, [49]). *A real x is called computable if there is a computable sequence of rationals $q_0, q_1, q_2, \ldots$ and a computable function f such that*

$$|x - q_{f(n)}| \leq 2^{-n}.$$

Let x be a computable real. As mentioned in Chapter 2, x is the characteristic function of a set X, and the n-th bit of x, denoted $x[n]$, is defined by:

$$x[n+1] = \begin{cases} 1 & \text{if } n \in X \\ 0 & \text{otherwise.} \end{cases}$$

x can be equivalently defined by:

$$x = \sum_{n \in X} 2^{-(n+1)}$$

as discussed in Section 5.5. Because reals are so closely related to sets in this way, they will inherit certain properties from them. The following theorem shows how closely computable reals and computable sets are connected:

Theorem 23 (Turing, [49]). *x is a computable real if and only if x is the characteristic function of a computable set X.*

Proof. Suppose that x is the characteristic function of a set X. Define

$$X_n = \{j : j < n \text{ and } j \in X\},$$

and set

$$x_n = \sum_{j \in X_n} 2^{-(j+1)}.$$

Then

$$\begin{aligned}
|x - x_n| &= \sum_{j \in X} 2^{-(j+1)} - \sum_{j \in X_n} 2^{-(j+1)} \\
&= \sum_{j \in X} 2^{-(j+1)} - \sum_{j<n, j \in X} 2^{-(j+1)} \\
&= \sum_{j \geq n, j \in X} 2^{-(j+1)} \\
&\leq \sum_{j>n} 2^{-j} \\
&\leq \sum_{j=n+1}^{\infty} 2^{-j} \\
&\leq 2^{-n},
\end{aligned}$$

so x is computable as the limit of the rational sequence $x_1, x_2, x_3, \ldots$ For the computable selection function the identity function will do.

Conversely, suppose that x is a computable real. We distinguish two cases - depending on whether x is rational or irrational. If x is rational, then it is the limit of the constant sequence of rationals $x, x, x, \ldots$ and so X can be computed from the first term in the sequence. If x is irrational, then clearly there must be an infinite number of zeros in the expansion of x. Suppose that we are at a stage s such that $q_{f(s)}[m] = 0$ and

$$|x - q_{f(s)}| < 2^{-(m+1)}.$$

This ensures that

$$(\forall t > s)(\forall p < m) q_{f(t)}[p] = q_{f(s)}[p]$$

and $x[p] = q_{f(s)}[p]$. Thus at stage s, the first $m-1$ bits of X can be decided. Because there are infinitely many zeros in x, there must be infinitely many stages s where this occurs, and so all of X can be decided. $\square$

5.7 Computably enumerable reals

As well as computable reals, we would like to define a weaker type of real, analogous to the computably enumerable sets. With the c.e. sets, we know that eventually all elements in the set will be generated. However, there is not a given time when we can say for sure whether or not a number is in the set.

Correspondingly, a computably enumerable real x will have a sequence of rationals converging to x, like a computable real. However, unlike a computable real, we don't know how good our approximation is. We know that over some unspecified length of time, the rationals will approach x. However this could take an extremely long time, and we would never know whether we were actually getting quite close, or whether the sequence would suddenly jump somewhere else entirely. Formally this concept is defined as:

Definition 16. *A real x is computably enumerable if there is a computable monotone increasing sequence of rationals with limit x.*

With a computably enumerable set, if a number is in the set then eventually we will find this out. However, we can never be certain that a number will *not* be in the set. Equivalently, if a number a is less than x, where x is computable enumerable, then eventually we will know this. However, if we can never be sure that a number a is greater than x.

It is natural to consider whether every computably enumerable real is the characteristic function of a computably enumerable set, the same way that a computable real is the characteristic function of a computable set. However, this is not the case, as the following proposition shows:

Proposition 24 (Soare, [41]). *There is a c.e. real x that is not the characteristic function of a c.e. set.*

Proof. Let $\{W_e : e \in \omega\}$ be a standard enumeration of the c.e. sets. For our real x to not be the characteristic function of a c.e. set, x must satisfy infinitely many requirements

$$R_e : x \neq \chi_e,$$

where χ_e is the characteristic function of W_e.

Construction. Set $q_0 = 01$. Dovetail on the W_e's. That is, at stage s run $\phi_0, \phi_1, \ldots, \phi_s$ for s steps each. Suppose at the end of stage s there are $e_0, e_1 \ldots e_n$ such that

$$2e_i + 1 \downarrow W_{e_i}[s] - W_{e_i}[s-1].$$

Define the next rational in the sequence,

$$q_{s+1} = q_{s+1}[1]q_{s+1}[2] \ldots q_{s+1}[2s+3]q_{s+1}[2s+4]$$

to be a binary string $2s+4$ bits long such that:

- $q_{s+1}[2s+3] = 0$.
- $q_{s+1}[2s+4] = 1$.
- For all $0 \leq j \leq s$,

$$q_{s+1}[2j+1] = \begin{cases} q_s[2j+1] & \text{if } j \notin \{e_0, e_1, \ldots, e_n\} \\ 1 & \text{otherwise.} \end{cases}$$

$$q_{s+1}[2j+2] = \begin{cases} q_s[2j+2] & \text{if } j \notin \{e_0, e_1, \ldots, e_n\} \\ 0 & \text{otherwise.} \end{cases}$$

- Set $x = \lim_s q_s$.

Verification. x is by definition the limit of a computable sequence of rationals. The sequence $q_0, q_1, \ldots$ is monotonically increasing - at a stage s, there are two possibilities:

1. $\{e_0, e_1, \ldots, e_n\} = \emptyset$. In this case

$$q_{s+1} = q_s 01,$$

which implies that $q_{s+1} > q_s$.

2. $\{e_0, e_1, \ldots, e_n\} \neq \emptyset$. q_{s+1} is the same as q_s01 except at a certain number of pairs of bits which have changed from 01 to 10. Say these bits are at $q_{s+1}[2j]$ and $q_{s+1}[2j+1]$. These bits add $2^{-(2j+1)}$ to q_s and 2^{-2j} to q_{s+1}, so
$$q_{s+1} > q_s01 > q_s.$$

Note that for all i there is a stage t_i such that for all stages
$$(\forall s > t_i) q_s[i] = q_{t_i}[i].$$

Since $x = \lim_s q_s$, it follows that
$$x[i] = q_{t_i}[i].$$

For all i, if $2i+1 \downarrow W_i[s]$, then for all $t > s$, $q_t[2i+2] = 0$ and hence $x[2i+2] = 0$. On the other hand, if $2i+1 \notin W_i$, then for all t, $q_t[2i+2] = 1$ and thus $x[2i+2] = 1$. Either way,
$$x[2i+2] = 1 \text{ if and only if } 2i+1 \in W_i,$$
and so x is not the characteristic function of a c.e. set. □

However, the converse property holds. If x is the characteristic function of a c.e. set, x is definitely a c.e. real:

Proposition 25. *Let X be a c.e. set, and x the characteristic function of X. x will be a c.e. real.*

Proof. If X is a finite set, then as in Theorem 23, the characteristic function of X will be a rational real, thus it is a c.e. real. So, let us assume that X is infinite. Define $X[s]$ to be X at stage s. Let us assume that $X[s]-X[s-1] \neq \emptyset$. Because X is an infinite set, we can ensure this by considering stage s to have been reached when an element enters X for the s-th time. Define a sequence of rationals $q_0, q_1, q_2, \ldots$ such that for all $j \geq 0$:

$$\begin{aligned} q_0[j+1] &= 0 \\ q_s[j+1] &= \begin{cases} q_{s-1}[j+1] & \text{if } j \notin X[s]-X[s-1] \\ 1 & \text{otherwise.} \end{cases} \end{aligned}$$

If the characteristic function of X is x, then

$$x = \lim_s q_s.$$

Since the sequence $q_1, q_2, \ldots$ is computable, and monotonically increasing, x is a c.e. real. □

Part II

Characterizations of randomness

Chapter 6

Probability theory and randomness

6.1 Random sequences

Probability theory is the study of *non-deterministic* processes. These are processes for which the outcome cannot be precisely known beforehand. A classic example of a non-deterministic process is coin-tossing, as it cannot be known for certain whether a coin, when thrown, will land on heads or tails.

There is a general assumption that tossing a coin is a "fair" way of deciding between two courses of action. By "fair", it is meant that the coin is equally likely to land on heads or tails. But how does one translate this intuitive understanding of fairness into a rigorous mathematical definition?

If we wanted to test whether a coin was indeed "fair", we would do so by tossing the coin, not once, but many times. Each time we would note whether the coin landed on heads or tails, perhaps by writing down '1' for a head and '0' for a tail. Then for each n, we let f_n be the number of '1's in the first n bits, i.e. the number of times heads was the outcome. Suppose it happened that as n got larger and larger, $\frac{f_n}{n}$ approached a limiting value. (Empirical observations suggest that this is the case). Let us call this limit the "probability" (of heads occuring), or $p(1)$ for short:

$$p(1) = \lim_{n\to\infty} \frac{f_n}{n}.$$

If $p(1) = \frac{1}{2}$ then we say the coin is "fair": heads and tails occur equally frequently, and hence each have an equal probability.

Suppose we had a coin that was not fair; say $p(1) = \frac{3}{4}$. If a gambler were betting on the outcome of this coin, he could make big money on it over a period of time. If he put money on 1 each time he could expect to win three out of four times. If, on the other hand, he were betting on a fair coin (one for which $p(1) = \frac{1}{2}$) he would have no way of betting that would guarantee him more wins than losses, in the long run. This is an example of a random process - one where there is no way of betting successfully on the outcome.

Suppose we note down the results of a random process (for example, 1's for heads and 0's for tails from a fair coin-tossing experiment). The resultant sequence of ones and zeros will have certain properties because it has been generated by a random process, and we will call it a "random" sequence. As mentioned above, a random binary sequence (representing a fair coin-tossing experiment) has the property that

$$\lim_{n\to\infty} \frac{f_n}{n} = \frac{1}{2}.$$

This fact is known as the *Law of Large Numbers*.

The Law of Large Numbers ensures that if an infinite sequence has many more ones than zeros (or vice versa) then that sequence cannot be random. However, it is not a sufficient condition of randomness, as many non-random sequences have this property as well, such as

$$1010101010101010\ldots.$$

Thus, if we are given a sequence of ones and zeros, the Law of Large Numbers is not be enough to decide if this sequence is random or not. The remainder of this chapter looks at early attempts to find a sufficient condition for random sequences. Note that in general we are interested in *infinite* binary sequences, i.e. sequences that could (in theory) be obtained by an infinite number of coin-tossing experiments, as these can then be associated with the real numbers in the manner discussed in Chapter 5.

6.2 *Kollektivs* and randomness

R. von Mises made the first serious attempt at a mathematical definition of random sequences in 1919. He was attempting to create a set of axioms for probability theory and had chosen "random sequences" as a primitive term in his axiomisation, so it was vital to his case to provide such a definition.

Von Mises was starting from a very empirical viewpoint, using a theory of probability based on observations of physical phenomena, such as coin tossing. In order to formalize this concept, he introduced *kollektivs*, infinite sequences of elements of a sample space. These were the mathematical representation of an infinite series of experiments of, for example, coin tosses. Of course, in order to appear like a genuine coin tossing experiment, this kollektiv would have to have certain properties. Obviously, an infinite sequence such as $11111111\ldots$ would not be a good representation of a coin tossing experiment.

Von Mises recognised that coin tossing (or, indeed, any good random process) has certain properties it must satisfy. He felt that the relative frequency of a given event must approach a limit, as the number of experiments increases, and that there should be no strategy a gambler could use if betting on the process that would be guaranteed to produce a profit over time.

The mathematical definition that is the result of von Mises' attempts to formalize these ideas is given in Definition 17. It is restricted to experiments (such as coin tossing) with only two outcomes, say O_1 and O_0.

Definition 17 (Kollektivs, von Mises, [52]). *An infinite binary sequence* $x_0x_1x_2\ldots$ *is a* Kollektiv *if it satisfies the following properties:*

1. *For all* n,
$$\lim_{n\to\infty}\frac{s_n}{n}=p$$
exists, where $s_n = |\{i \leq n : x_i = 1\}|$.

2. *Call* $\phi : 2^{<\omega} \to \{0,1\}$ *a place-selection rule. An infinite subsequence of* $x_0x_1x_2\ldots$ *is selected via* ϕ *by allowing* x_j *in the subsequence if*
$$\phi(x_0\ldots x_{j-1}) = 1.$$
For every admissible *place-selection rule the corresponding subsequence satisfies Item 1 (above) and the limit is* p*, the same as for the original sequence.*

From this the probability of outcome O_1 occurring is p, and the probability of outcome O_0 occurring is $1-p$.

6.3 Admissible place-selection functions

If *all* place selection functions are deemed admissible, and $x = x_0x_1x_2\ldots$ is any infinite binary sequence, we can define $\phi_{0,x}$ and $\phi_{1,x}$ by

$$\phi_{0,x}(x_0x_1\ldots x_{i-1}) = \begin{cases} 1 & \text{if } x_i = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\phi_{1,x}(x_0x_1\ldots x_{i-1}) = \begin{cases} 0 & \text{if } x_i = 1 \\ 1 & \text{otherwise.} \end{cases}$$

The infinite subsequence selected by $\phi_{0,x}(x)$ has $p = 1$, and that selected by $\phi_{1,x}(x)$ has $p = 0$. i.e. x is not a kollektiv. But x is any binary sequence. So if all these functions are considered to be admissible place-selection functions, no kollektiv can exist. This suggests that a judicious choice of admissible place-selection rules is required for this concept to work. The question is, which functions should be chosen?

In [53], A. Wald proved that kollektivs existed for every *countable* class of place-selection rules. In fact, Wald's proof showed that, given any class $\mathcal{C}$ of place-selection functions the set of kollektivs relative to $\mathcal{C}$ has Lebesgue measure one - so nearly every infinite sequence is a kollektiv. Some mathematicians considered von Mises' probability theory to be fundamentally inconsistent, because of the existence of $\phi_{0,x}$ and $\phi_{1,x}$ (above). Von Mises' disagreed with this view, as he felt that these functions defined in terms of kollektivs are not very natural concepts. Von Mises' lacked the terminology to make this belief concrete until, in 1940, A. Church [11] suggested that computability theory provided the necessary framework to describe admissible functions. In Church's view, the countable set of *computable* selection functions should be taken as the admissible place-selection rules. The class of reals that classify as kollektivs according to the computable selection functions are called *stochastic*.

Definition 18. *A real x is* stochastic, *or* Church random *if and only if x is a kollektiv for the class of computable selection functions.*

However, in 1939 Ville [51] argued that Church randomness, or in fact any definition based on kollektivs, was not an acceptable definition of randomness. Ville proved that for any countable set of admissible place selection functions there is a kollektiv that fails to satisfy the law of the iterated logarithm, a known statistical law.

In 1933, A. Kolmogorov published another axiomisation of probability theory based on measure theoretical principles. This formalization of probability theory became very widely accepted. As it didn't require a definition of random sequences in the way that von Mises' axiomisation did, the interest in finding a definition of random sequences decreased significantly. No major work was done on random sequences between 1940 and the early 1960's.

Chapter 7

Randomness for finite strings

7.1 Function dependent Kolmogorov complexity

In the early 1960's, there was a new development in the study of algorithmic randomness. It was suggested that random strings could be viewed as "chaotic" strings, or strings that have no regular pattern to them. If we were given such a string and told it was the outcome of a coin-tossing experiment, we would not be surprised by this fact. (Whereas if we were given a one-hundred bit string $1111\ldots111$ we would be *very* surprised to be told it represented one hundred coin tosses). This concept of random strings as irregular strings that contain no "surprise value" was by no means original. However, like von Mises' kollektivs, the tools of computability theory were required in order to formalize these intuitive notions.

Let us start by considering an example of what constitutes a non-random string. The million-bit string

$$101010\ldots\ldots\ldots1010$$

is certainly not random. This string could not represent a sequence of a million coin tosses - it is extremely regular, it has properties that are recognisable, and it is easy to describe. In fact it could be fully described as "the million-bit string starting with 1 and consisting of alternate 1's and 0's".

All that is necessary for a computer to output this string is a very simple program, such as:

```
for i=1:500000
print 10.
```

Such a string is "compressible" into a very short description, from which either a computer or human can reproduce the string exactly.

On the other hand, consider a random million-bit string, one that has been generated from a million coin tosses. It is unlikely to have any recognisable pattern. Unlike the previous string, where the regularities were exploited in order to describe it succinctly, there is no shorthand way of describing a random string such that a human (or computer) can reproduce it exactly. The only way that this string could be described is by writing out the whole string in full. This string's shortest description is the string itself. Such a string is "incompressible".

From this idea a definition of randomness can be formulated:

> *A finite string x of length n is random if and only if x has no description that is shorter than n, i.e. x is its own shortest description. x is non-random if there is a description of x that is shorter than x.*

This is the fundamental concept behind Kolmogorov complexity. Kolmogorov complexity was conceived quite independently around the same time by R.J. Solomonoff [43], [44], A.N. Kolmogorov [24], and G.J. Chaitin [7], [8], and makes it possible to quantify the complexity of randomness of finite strings. Like von Mises' kollektivs, Kolmogorov complexity could not immediately provide a successful definition of randomness for infinite sequences. However, both these concepts were instrumental in the development of a definition of randomness that was acceptable.

We now formally define Kolmogorov complexity. A description of a string x is defined to be the input to a computable function, or Turing machine, that results in x:

Definition 19. *Let $f : 2^{<\omega} \to 2^{<\omega}$ be a partial computable function. A string d is a* description *of a string x (relative to f) if and only if*

$$f(d) = x.$$

Then the Kolmogorov complexity of x relative to f is the length of the shortest description:

Definition 20. *Let $f : 2^{<\omega} \to 2^{<\omega}$ be some partial computable function, and let x be a (finite) string. The Kolmogorov complexity relative to f is written $K_f(x)$ and is defined by*

$$K_f(x) = \min(\{|p| : f(p) = x\} \cup \{\infty\}).$$

The ∞ value is included because it is possible that no element of the domain of f is a description of x. In that case $K_f(x) = \infty$.

7.2 Absolute Kolmogorov Complexity

The definition of Kolmogorov complexity is very dependent on the function that it is relative to. It appears that this severely weakens Kolmogorov complexity; it is possible that it might even render it meaningless. After all, for *any* string x, there is a computable function (call it f_x), such that $f_x(p) = x$ for any input p. Thus $K_{f_x}(x) = 1$, which would seem to negate any universal usefulness that the Kolmogorov complexity has. However it turns out that there is an absolute version of Kolmogorov complexity. That is, there is a particular function g, relative to which any string x has the least complexity possible (to within a constant factor):

Theorem 26 (The Invariance Theorem). *There is a function g such that for all partial computable functions f and all strings x,*

$$K_g(x) \leq K_f(x) + \mathcal{O}(1).$$

Proof. Take a Universal Turing Machine U. A UTM has the property that for *all* partial computable functions f there is a p such that for all strings y, $U(\overline{p}y) = f(y)$ (where $\overline{p}y = 1^{|p|}0pn$). Supposing that for some x, $f(y) = x$, we get

$$\begin{aligned} K_U(x) &= |y| + (2|p| + 1) \\ &\leq K_f(x) + \mathcal{O}(1). \end{aligned}$$

□

Thus we can call $K_U(x)$ *the* Kolomogorov complexity $K(x)$, in the knowledge that the complexity of any string x can never be less than this, beyond a constant factor.

7.3 Simple results on Kolmogorov complexity

In this section we look at some trivial, yet useful, results relating to Kolmogorov complexity.

Lemma 27. $K(x) \leq |x| + \mathcal{O}(1)$.

Proof. Define a Turing machine T_{id} via:

$$T_{id}(x) = x.$$

This is an extremely simple Turing machine, which when given input, doesn't do anything at all, but just halts immediately. Obviously, $K_{id}(x) = |x|$, so by the Invariance Theorem:

$$\begin{aligned} K(x) \leq K_{id}(x) + \mathcal{O}(1) \\ \leq |x| + \mathcal{O}(1), \end{aligned}$$

as required. □

Lemma 28. *For any partial computable function ϕ, $K(\phi(x)) \leq K(x) + \mathcal{O}(1)$.*

Proof. Suppose that ϕ is computed by the Turing machine T_n in a standard enumeration of Turing Machines $T_0, T_1, \ldots$, and let U be a universal Turing machine. Suppose that $U(p) = x$ for some p. It is possible to define a Turing machine V by

$$V(1^{|n|}0np) = T_n(U(p)).$$

Then

$$\begin{aligned} V(1^{|n|}0np) &= T_n(x) \\ &= \phi(x). \end{aligned}$$

Hence the Invariance Theorem ensures that the Kolmogorov complexity of $\phi(x)$ is no worse than $|1^{|n|}0np|$:

$$\begin{aligned} K(\phi(x)) &\leq |1^{|n|}0np| \\ &\leq 2|n| + 1 + |p| \\ &\leq K(x) + 2|n| + 1 \\ &\leq K(x) + \mathcal{O}(1), \end{aligned}$$

as required. □

7.4 Randomness of finite strings

Following the philosophy outlined at the beginning of this chapter, the following seems a reasonable definition for a random finite string:

Definition 21. *A string x of length n is incompressible (or random) if $K(x) \geq n$.*

This definition is well-defined, as a simple counting argument ensures that for any n there is at least one random string of length n:

Lemma 29. *For all n, there is an x such that $|x| = n$ and $K(x) \geq n$.*

Proof. For each n there are 2^n binary strings. There are

$$\sum_{i=0}^{n-1} 2^i = 2^n - 1$$

descriptions shorter than n. Thus at least one binary string is incompressible. $\square$

With finite strings the difference between random and non-random strings is not clear-cut. Obviously a string with a Kolmogorov complexity only a very little less than its length can be thought of as *almost* random, as opposed to one that has a significantly shorter length. A finer definition of incompressibility can be considered:

Definition 22. *A string x of length n is c-incompressible if $K(x) \geq n - c$.*

Theorem 30. *There are at least $2^n - 2^{n-c} + 1$ c-incompressible strings of length n.*

Proof. There are 2^n strings of length n. There are

$$\sum_{i=0}^{n-c-1} 2^i = 2^{n-c} - 1$$

descriptions shorter than $n - c$. $\square$

This means that at least half of all strings are 1-incompressible, three-quarters are 2-incompressible, and so on; that is, most strings are very nearly random.

7.5 Initial-segment complexity of infinite strings

It would be desirable to extend the concept of incompressibility from finite sequences to infinite sequences (i.e. real numbers). One way to do this is to consider a real x to be random if *all* the initial segments of x have high

Kolmogorov complexity; that is, there is some constant c such that for all n, $K(x \upharpoonright n) > n - c$. However, it turns out that *no* real is random according to this definition.

To see that this is the case, define a function $\# : \omega \to 2^{<\omega}$ that, given $n \in \omega$, outputs the n-th string in the usual length-lexicographic ordering (as shown in Table 7.1):

$n \in \omega$	$\#(n)$
0	λ
1	0
2	1
3	00
4	01
5	10
6	11
$\vdots$	$\vdots$

Table 7.1: String positions in the length-lexicographic ordering

Define a machine M such that for all strings σ,

$$M(\sigma) = \#(|\sigma|)\sigma.$$

In particular, for any real x, define $m_n = \#(x \upharpoonright n)$ for all n. Then

$$\begin{aligned} M(x[(n+1)\dots n+m_n]) &= \#(|x[(n+1)\dots n+m_n]|)x[(n+1)\dots n+m_n] \\ &= \#(m_n)x[(n+1)\dots n+m_n] \\ &= (x \upharpoonright n)x[(n+1)\dots n+m_n] \\ &= x \upharpoonright (n+m_n). \end{aligned}$$

If we set $t_n = n + m_n$ we notice that

$$\begin{aligned} K_M(x \upharpoonright t_n) &\leq |x[(n+1)\dots n+m_n]| \\ &\leq m_n \\ &\leq t_n - n. \end{aligned}$$

Since this is true for all n, it follows that

$$(\forall n)(\exists t_n)K_M(x \upharpoonright t_n) \leq t_n - n.$$

Hence x cannot be random according to the definition suggested above.

A satisfactory definition of randomness for infinite strings in terms of initial segment complexity was found eventually (we discuss it in Chapter 11), but before that time another characterization of randomness was proposed. We look at that now.

Chapter 8

The first definition

8.1 Martin-Löf randomness

Drawing inspiration from both von Mises' approach to randomness and Kolmogorov complexity, the Swedish mathematician P. Martin-Löf introduced the first widely acceptable notion of randomness in 1966.

Recall that Von Mises' kollektivs were based on the assumption that every random number should satisfy the law of large numbers; that is, the proportion of 1's in any initial segment should approach $\frac{1}{2}$ as the length of the initial segment increases. This should certainly be true, but it is just one statistical law. According to Martin-Löf, this is not sufficient. Random numbers should obey many more laws than this one. For example, it is also the case in any initial segment the number of 11's as a proportion of all two-bit pairs should approach $\frac{1}{4}$, the proportion of 111's should approach $\frac{1}{8}$, and so on. There are many more such rules a truly random string should satisfy. Martin-Löf's concept of randomness was that *every one* of these properties must be satisfied by a random real. This was the difference between his randomness definition and von Mises', and why Martin-Löf's definition is ultimately much more acceptable.

How can all these rules of randomness be satisfied? To see how, let us divide all infinite strings into two types, the "typical" strings, and the "special" strings. The "special" strings have a regularity or a recognisable property, one that a sequence of coin tosses would not be expected to make; that is, they have the "surprise value" discussed in Chapter 7. An example of such a string is one for which every tenth bit is a 1. There are infinitely many strings that satisfy this property. However, the probability that a fair coin would produce any of these strings is zero. Similarly, a sequence that doesn't satisfy the law of large numbers (maybe it has twice as many zeros as ones) will never be produced by tossing a fair coin. If *every* sequence that a fair coin might produce has a certain property, we will call that property a "Law of Randomness". The Law of Large Numbers is one such law. Another law is that it is *not* the case that every tenth bit is a 1.

For any law of randomness, the set of strings X that satisfy this law will be of measure one. Any string that doesn't satisfy this law (i.e. not in X) *cannot* be random. Conversely, all random strings will be contained in X. Since this is true of all measure one sets, any genuinely random string will be in *all* measure one sets representing a law of randomness. The natural definition of randomness stemming from these ideas seems to be:

A real x is random if and only if x is in all *measure one sets.*

Unfortunately, it is not quite this simple. Suppose that we required that a random string be in *all* measure one sets. Then consider any string x. The singleton set $\{x\}$ containing only x is of measure zero, so that the complement of $\{x\}$ is of measure one. So for *any* string x there is a measure one set - namely $\overline{\{x\}}$ - such that x is not in that set. Because of this, *no* string can be in all measure one sets, and so no string can be random according to this definition.

This problem is similar to the problem faced by von Mises' *kollektiv* definition in Chapter 6, and it can be resolved in the same way by requiring the sets involved to be computable before they are considered a suitable candidate. Martin-Löf's definition only looks at whether a string is contained in *effective* tests - measure zero sets that can be generated using computational means. A random sequence is one which fails all such effective tests. The intersection of the complements of all these sets is non-empty; in fact it has measure one, which means that nearly every infinite sequence is effectively random.

Definition 23. *A Martin-Löf test is a computable collection of computably enumerable open sets, $\{U_e : e \in \omega\}$, such that $\mu(U_e) \leq 2^{-e}$. We say a real x withstands a Martin-Löf test if $x \notin \cap_e U_e$.*

A random real will have no "non-typical" attributes and so should withstand all such effective statistical tests.

Definition 24 (Martin-Löf, [35]). *A real x is Martin-Löf random if x withstands all Martin-Löf tests $\{U_e : e \in \omega\}$.*

Observe that a simple way of viewing a Martin-Löf test is as a subset of the usual c.e. sets $W_0, W_1, \ldots$. A computable collection of these sets depends on a (total) computable function that, for any e, selects the index of the c.e. set that is to be U_e. That is, there exists a ϕ_i such that for any e,

$$U_e = W_{\phi_i(e)}.$$

Hence any Martin-Löf test is uniquely defined by the index i (in the standard enumeration) of the corresponding index-selection function.

However, the c.e. sets are collections of finite binary sequences, and when we say that for some real x, $x \in \cap_e U_e$, we are actually suggesting that each U_e is a collection of *infinite* binary sequences. This is the result of some notational laziness, as what we *really* intend is that for all e,

$$U_e = \bigcup_{\sigma \in W_{\phi_i(e)}} [\sigma],$$

where (as we recall from Chapter 5), $[\sigma]$ refers to the set of all infinite extensions of σ. We can safely assume the equivalence of these two notions throughout all the earlier work, but some care will be needed differentiating them in later chapters, particularly Chapter 16.

8.2 The Universal Test

It is frequently convenient to be able to appeal to a single *Universal Martin-Löf test*.

Theorem 31. *There is a Universal Martin-Löf test; that is, there is a Martin-Löf test $\{V_n : n \in \omega\}$ such that x is Martin-Löf random if and only if $x \notin \cap_n V_n$.*

Proof. It is possible to enumerate the Martin-Löf tests, which are simply computable sequences of computably enumerable sets. It is a simple matter to enumerate the computably enumerable sets, discarding any tests if they grow larger than the required measure. Suppose we have our enumeration, so for each i, $\{U_n^i : n \in \omega\}$ is a Martin-Löf test:

$$\begin{array}{cccc} U_0^0 & U_1^0 & U_2^0 & \dots \\ \\ U_0^1 & U_1^1 & U_2^1 & \dots \\ \\ U_0^2 & U_1^2 & U_2^2 & \dots \\ \vdots & \vdots & \vdots & \end{array}$$

Define the Universal Test $\{V_n : n \in \omega\}$ via:

$$V_n = \bigcup_{i=0}^{\infty} U_{i+n+1}^i.$$

For each n, V_n is computably enumerable and furthermore,

$$\begin{aligned}
\mu(V_n) &= \mu(\bigcup_{i=0}^{\infty} U^i_{i+n+1}) \\
&\leq \sum_{i=0}^{\infty} \mu(U^i_{i+n+1}) \\
&\leq \sum_{i=0}^{\infty} 2^{-(i+n+1)} \\
&\leq \frac{2^{-n}}{2} \sum_{i=0}^{\infty} 2^{-i} \\
&\leq 2^{-n},
\end{aligned}$$

so $\{V_n : n \in \omega\}$ is indeed a Martin-Löf test. To show that it is universal, assume firstly that x is not random; that is, that there is a Martin-Löf test $\{U^i_n : j \in \omega\}$ such that $(\forall n)(x \in U^i_n)$. In particular, $(\forall n)(x \in U^i_{i+n+1})$, so that $(\forall n)(x \in V_n)$ and $x \in \cap_n V_n$, as required.

Conversely, suppose that $x \in \cap_n V_n$. Then there is a Martin-Löf test (namely V_n) that x does not withstand, and hence x is not Martin-Löf random, as required. □

8.3 Solovay's characterization

R. Solovay suggested an alternative measure theoretic definition of randomness, using the conditions of the first Borel-Cantelli Lemma (Lemma 20) to create a measure zero set. It turns out that this definition is equivalent to Martin-Löf randomness.

Definition 25 (Solovay, [45]). *A* Solovay test *is a computable collection of computably enumerable sets such that* $\sum_{n \in \omega} \mu(U_n) < \infty$. *A real* x *withstands a Solovay test if* x *is in at most finitely many* U_n.

Theorem 32 (Solovay, [45]). *A real* x *is Martin-Löf random if and only if* x *withstands all Solovay tests* $\{U_n : n \in \omega\}$.

Proof. Suppose that x is not Martin-Löf random, so there is a Martin-Löf test $\{U_n : n \in \omega\}$ such that $(\forall n)(x \in U_n)$. Note that

$$\begin{aligned}\sum_{n \in \omega} \mu(U_n) &\leq \sum_{n \in \omega} 2^{-n} \\ &< \infty,\end{aligned}$$

so $\{U_n : n \in \omega\}$ is a Solovay test. Hence there is a Solovay test (namely $\{U_n\}$) such that x is in infinitely many (in fact, all) U_n.

Conversely, suppose that there is a Solovay test $\{V_n : n \in \omega\}$ with x in infinitely many V_n. Assume without loss of generality that

$$\sum_{n \in \omega} \mu(V_n) \leq 1.$$

For all k, define U_k via

$$U_k = \{y : y \text{ is in at least } 2^k \text{ of the } V_n\}.$$

U_k is a computably enumerable set because each of the V_n are computably enumerable, and note that

$$\begin{aligned}2^k \mu(U_k) &= 2^k \sum_{y \in U_k} 2^{-y} \\ &\leq \sum_{n \in \omega} \mu(V_n) \\ &\leq 1,\end{aligned}$$

so that $\mu(U_k) \leq 2^{-k}$ and $\{U_k : k \in \omega\}$ is a Martin-Löf test. Observe that x is in infinitely many V_n, so that for any k, x is in more than 2^k of the V_n, and hence $x \in U_k$. Thus x is not Martin-Löf random. □

If $\sum_{n \in \omega} \mu(V_n) > 1$, there is an $R \in \omega$ so that

$$\sum_{n \in \omega} \mu(V_n) \leq R.$$

Define $U_k = \{y : y \text{ is in at least } R2^k \text{ of the } V_n\}$. The rest of the proof follows.

Chapter 9

Martingales and randomness

Suppose we have a binary sequence, the bits of which are revealed one at a time. A player P is betting on whether the next bit in the sequence is a 1 or a 0. The gambling system used here is a "double or nothing" system. Say P has capital c, and bets an amount a on an event occuring. If the event occurs, P receives $2a$ back. P's new capital would be $c + a$. On the other hand, if the event does not occurs, the bet is forfeit and P now only has capital $c - a$.

P knows what the first n bits are, and he uses some strategy to decide how much to bet, and whether on 1 or 0. The strategy may be a good strategy to use on that particular string. However, it could also be a poor choice. Let $d(x)$ represent the amount of money that P would have using his strategy to bet on a string that turned out to be x. Suppose this strategy tells him to bet an amount u that the next bit will be 0. If the next bit indeed turns out to be 0, his new capital will be $d(x0) = d(x) + u$. On the other hand, if the next bit turns out to be 1, his new capital would be $d(x1) = d(x) - u$. Note that no matter what the value of u,

$$\begin{aligned} d(x0) + d(x1) &= [d(x) + u] + [d(x) - u] \\ &= 2d(x). \end{aligned}$$

Any function that satisfies the above equation is known as a martingale.

If the sequence that P is betting on is non-random, then our intuitive notion of randomness implies that there will be some regular pattern to it. This pattern could be exploited by P in order to create a successful gambling strategy i.e. one that gains P unlimited amounts of money.

For example, suppose that our non-random string has the property that every tenth bit is a 1. A winning strategy to use on such a string would be to place extremely small bets for nine rounds. Then on the tenth round, all P's capital could be bet on 1. In this round P would practically double his capital. Repeating this strategy over time would allow P to gain unlimited amounts of capital.

Conversely, a truly random string would be lacking in any regularities. It would have no pattern from which P could formulate a strategy that would be guaranteed to increase P's capital. Given a random string, the probability of the next bit being a 1 is equal to the probability of it being a 0. P might win some rounds, and lose some rounds, but would have no way of gaining unlimited capital in such a situation.

This is a way of characterizing randomness:

> *A string x is random if and only if there is no betting strategy which succeeds on x in the long term.*

Martingales are used to formalize this concept.

9.1 Martin-Löf randomness in terms of martingales

Definition 26. *A martingale is a function*

$$d : 2^{<\omega} \to [0, \infty)$$

that for all $x \in 2^{<\omega}$ satisfies the property

$$d(x) = \frac{d(x0) + d(x1)}{2}.$$

If a gambling strategy is successful on a string x, then the corresponding martingale (which represents the amount of capital P will get) will be unbounded, if the game is continued for infinitely many rounds. In that case, we say the martingale *succeeds* on x:

Definition 27. *A martingale d succeeds on $x \in 2^\omega$ if $\limsup_{n\to\infty} d(x \upharpoonright n) = \infty$.*

Schnorr proved in [39], [40] that Martin-Löf random reals correspond to reals that have no *weakly computable* martingale succeed on them, where a weakly computable martingale is defined as follows:

Definition 28. *A martingale d is weakly computable if there is a computable binary function $g : 2^{<\omega} \times \omega \to \mathbb{Q}^+$ such that for all s and for all x,*

$$\lim_s g(x, s) = d(x)$$

and

$$g(x, s) \leq g(x, s + 1).$$

Theorem 33 (Schnorr, [39]). *A real $x \in 2^\omega$ is Martin-Löf random if and only if no weakly computable martingale succeeds on x.*

To prove Theorem 33 we require the following lemma. For the proof of the lemma, see the Appendix.

Lemma 34. *Let d be a martingale. Define the set $F_k = \{x \in 2^{<\omega} : d(x) > k\}$. Define $G_k = \bigcup_{x \in F_k} \{y : y \in [x]\}$, where $[x]$ is the set containing all extensions of x. Then*

$$\mu(G_k) \leq \frac{1}{k} d(\lambda).$$

Now for the proof of Theorem 33.

Proof. Suppose that there is a (weakly computable) martingale d that succeeds on x. Assume without loss of generality that $d(\lambda) \leq 1$. Define

$$S_n = \{\sigma : d(\sigma) \geq 2^n\},$$

and set

$$U_n = \bigcup_{\sigma \in S_n} [\sigma],$$

where $[\sigma]$ contains all extensions of σ. U_n is a c.e. set as d is a weakly computable martingale, and by Lemma 34, $\mu(U_n) \leq 2^{-n}$, so U_n is a Martin-Löf test. Since d succeeds on x,

$$x \in \bigcap_{n \in \omega} U_n$$

by the definition of U_n, so x is not Martin-Löf random.

Conversely, suppose that there is a Martin-Löf test $\{U_n : n \in \omega\}$, such that

$$x \in \bigcap_{n \in \omega} U_n.$$

Each U_n can be defined by $U_n = \bigcup_m [\sigma_{n,m}]$, where $\{\sigma_{n,m} : m \in \omega\}$ is a prefix-free set. We will define a martingale d that succeeds on x. Let

$$d(\tau) = \sum_{n,m} w_{n,m}(\tau)$$

where $w_{n,m}(\tau)$ is the weighting function:

$$w_{n,m}(\tau) = \begin{cases} 1 & \text{if } \sigma_{n,m} \sqsubseteq \tau \\ 2^{-(|\sigma_{n,m}| - |\tau|)} & \text{if } \tau \sqsubset \sigma_{n,m} \\ 0 & \text{otherwise} \end{cases}$$

It is straightforward to check that $w_{n,m}$ satisfies the martingale property, and so d will also satisfy it. By the construction of d, and because each U_n is a c.e. set, d is weakly computable. Furthermore, because for all n, $x \in U_n$ it follows that

$$(\forall n)(\exists m, j) w_{n,m}(x \upharpoonright j) = 1,$$

and so

$$\lim_{n \to \infty} d(x \upharpoonright n) = \infty,$$

as required. $\square$

9.2 The Universal Martingale

When considering Martin-Löf randomness in terms of test sets, there is a single universal Martin-Löf test that contains all random reals. In a similar way there is a universal martingale.

Theorem 35. *There is a weakly computable martingale D such that x is random if and only if* $\limsup_n D(x \upharpoonright n) < \infty$.

Proof. Let $\{V_n : n \in \omega\}$ be the Universal Martin-Löf test as defined in Theorem 31. Applying the process used in the proof of Theorem 33 to $\{V_n\}$, we get a weakly computable martingale D such that if $x \in \cap_n V_n$, then D succeeds on x. But $\cap_n V_n$ includes every non-random real, as $\{V_n\}$ is universal. Hence if x is non-random, then D succeeds on x.

Conversely, if D succeeds on x, then there is a weakly computable martingale (namely D) that succeeds on x, so by definition x is non-random. Combining these two statements gives us

$$x \text{ non-random } \iff \limsup_n D(x \upharpoonright n) = \infty.$$

The theorem is simply the negation of this statement. □

9.3 Supermartingales

A more general version of betting strategy that will be useful in this work is the supermartingale.

Definition 29. *A* supermartingale *is a function* $f : 2^{<\omega} \to \mathbb{R}^+ \cup 0$ *such that for all* $\sigma \in 2^{<\omega}$,

$$f(\sigma 0) + f(\sigma 1) \leq 2f(\sigma).$$

Supermartingales have some attractive features mathematically that martingales do not (largely due to the existence of a *multiplicatively optimal* supermartingale, the existence of which was proved by Schnorr), and in situations it is more convenient to use supermartingales in place of martingales. We can do this with impunity as the following theorem assures us that the random reals are the same whether or not strict equality is insisted upon.

Theorem 36. *For all $x \in 2^\omega$, there is a weakly computable supermartingale d such that $x \in S[d]$ if and only if there is a weakly computable martingale f such that $x \in S[f]$.*

Proof. Suppose firstly that there is a (weakly computable) martingale f such that $x \in S[f]$. Then there is a supermartingale (namely, f) such that $x \in S[f]$.

Conversely, suppose that there is a (weakly computable) supermartingale d such that $x \in S[d]$. Define a martingale f inductively as follows:

$$\begin{aligned} f(\lambda) &= d(\lambda) \\ f(\sigma 0) &= d(\sigma 0) + \frac{2f(\sigma) - [d(\sigma 0) + d(\sigma 1)]}{2} \\ f(\sigma 1) &= d(\sigma 1) + \frac{2f(\sigma) - [d(\sigma 0) + d(\sigma 1)]}{2}. \end{aligned}$$

By construction, f is a martingale. We show via induction that for all $\sigma \in 2^{<\omega}$, $f(\sigma)$ is weakly computable and $f(\sigma) \geq d(\sigma)$:

Base case: $f(\lambda) = d(\lambda)$, by construction.

Induction step: Assume that $f(\sigma) \geq d(\sigma)$, and $f(\sigma)$ is weakly computable. Note that by the induction hypothesis, $f(\sigma 0) \geq d(\sigma 0)$ and $f(\sigma 1) \geq d(\sigma 1)$. Since d and f are weakly computable, and we know that at any stage s, there is some stage $t \geq s$ such that $2f_t(\sigma) \geq d_t(\sigma 0) + d_t(\sigma 1)$, we can wait to enumerate $f(\sigma 0)$ and $f(\sigma 1)$ until such a stage occurs. In this way, $f(\sigma 0)$ and $f(\sigma 1)$ are non-decreasing (thus, weakly computable) as required. Furthermore,

$$\begin{aligned} x \in S[d] &\implies \limsup_{n\to\infty} d(x \upharpoonright n) = \infty \\ &\implies \limsup_{n\to\infty} f(x \upharpoonright n) = \infty \\ &\implies x \in S[f], \end{aligned}$$

as required. □

9.4 Criticism of Martin-Löf randomness

Once Schnorr had proved the theorems above he used them to point out what he considered to be flaws in the definition of Martin-Löf randomness. Schnorr felt that there was no reason why the weakly computable martingales used had to be approximated from below. He argued that martingales that were approximated from above should just as good. However, the classes of random reals differ depending on whether F or $-F$ are weakly computable. (We look more at this in Chapters 10 and 14.) This seemingly meaningless lack of symmetry in the class of Martin-Löf random reals seems to imply that it is not an ideal definition of randomness.

Furthermore, Schnorr believed that Martin-Löf random reals are required to satisfy more laws than are actually necessary. He argued (in [39]) that randomness should not entail "all relevant statistical properties of randomness" but only those properties "that have a physical meaning". If an infinite sequence turns out to be non-random, but does so subtly or so slowly that no "effective procedure" could observe it happening, then this sequence essentially behaves as a random sequence.

In terms of martingales, the interpretation of this argument is that for a sequence to be non-random, not only does a martingale have to succeed on it, but it has to succeed on it quickly enough that it is obviously going to succeed, to anyone observing it using effective (or computable) methods. In Chapter 10 we look at how Schnorr formalises this idea of "effective success".

Chapter 10

Schnorr and computable randomness

10.1 Schnorr randomness

As mentioned at the end of Chapter 9, Schnorr felt that Martin-Löf's formulation of randomness was too strong, and demanded properties of random sequences that were not meaningful in a physical sense. In order to make the definition stronger, he first suggested that the martingales should not just be weakly computable, but should be computable also - i.e. not only should a non-decreasing sequence approach $d(x)$, but that it should do so at a known rate of convergence. Hence he suggested the following definition:

Definition 30. *A function $d : 2^{<\omega} \to \mathbb{R}$ is called* computable *if there is a computable function $g : \omega \times 2^{<\omega} \to \mathbb{Q}^+$ such that for all n,*

$$|d(x) - g(n, x)| < 2^{-n}.$$

However, this is not enough because the martingale, though convergent itself, could still succeed on a real x too slowly to be obvious. Hence the success itself ought to be effective - i.e. the martingale should go to infinity at least as fast as a computable function.

Definition 31. *A function d* effectively succeds *on a real x if there is a computable, monotonic and unbounded function $h : \omega \to \omega$ such that*

$$\limsup_n (d(x \upharpoonright n) - h(n)) \geq 0;$$

that is, $d(x \upharpoonright n) \geq h(n)$ for infinitely many n.

Then the definition of Schnorr randomness uses computable martingales and the effecitve version of success as defined above:

Definition 32 (Schnorr, [39]). *A real x is Schnorr random if no computable martingale effectively succeeds on x.*

Martin-Löf randomness has a characterization in terms of measure zero test sets, and an equivalent characterization can be found for Schnorr.

Definition 33. *A Schnorr test $\{V_n : n \in \omega\}$ is a Martin-Löf test with the property that there is a computable function $f : \omega \to \mathbb{R}^+$ such that $f(n) = \mu(V_n)$ for all n.*

Theorem 37 (Schnorr, [40]). *A real x is Schnorr random if and only if for all Schnorr tests,*

$$x \notin \cap_n V_n.$$

To prove this Theorem we will require the use of the following Lemma. For the proof of this Lemma, see the Appendix.

Lemma 38. *Suppose that f is a computable, unbounded and non-decreasing function and that $\{U_n : n \in \omega\}$ is a Schnorr test. For each n consider each U_n to be a prefix-free collection of strings, $U_n = \{\sigma_{n,i} : i \in \omega\}$. Suppose that each U_n has the property that if*

$$S_n = \sum_{i \in \omega} 2^{-|\sigma_{n,i}|} 2^{f(|\sigma_{n,i}|)}$$

then $S_n \leq 2^{-n}$ and S_n is a computable real. Then there is a computable martingale d such that if $x \in \cap_n U_n$ then $d(x \upharpoonright n) \geq 2^{f(n)}$ infinitely often.

We turn to the proof of Theorem 37.

Proof. The proof of this theorem is clearly very similar to the proof of Theorem 33. The same procedure shows that if a martingale succeeds on x, a test can be built that x does not withstand, and vice versa. All that remains to be shown is that if a computable martingale effectively succeeds on x, the related test is indeed a Schnorr test, and vice versa.

Suppose there is a computable martingale G that effectively succeeds on x. Without loss of generality, assume that $G(\lambda) \leq 1$. As in the case of Martin-Löf randomness, define a test set $\{V_n : n \in \omega\}$ via

$$V_n = \{[\sigma] : G(\sigma) \geq h(|\sigma|) \text{ and } h(|\sigma|) \geq 2^n\}.$$

The arguments used in Theorem 33 ensure that for each n, $\mu(V_n) \leq 2^{-n}$. It also needs to be shown that $\mu(V_n)$ is computable.

For any s, define

$$m_s = \min\{n : h(n) \geq 2^s\}.$$

Then define

$$V_{n,s} = \{[\sigma] : G(\sigma) \geq h(|\sigma|) \wedge h(|\sigma|) \geq 2^n \wedge |\sigma| \leq m_s\}.$$

Note that $|\mu(V_n) - \mu(V_{n,s})| = \mu(X)$ where

$$X = \{[\sigma] : G(\sigma) \geq h(|\sigma|) \geq 2^n \text{ and } |\sigma| > m_s\}.$$

Since h is a non-decreasing function it follows that

$$\begin{aligned} G(\sigma) &\geq h(|\sigma|) \\ &\geq h(m_s) \\ &\geq 2^s. \end{aligned}$$

Applying Lemma 117, we get

$$\begin{aligned} G(\lambda) &\geq \sum_{x \in X} 2^{-|x|} G(x) \\ &\geq \sum_{x \in X} 2^{-|x|} 2^s. \end{aligned}$$

Since $G(\lambda) \leq 1$ this implies that

$$\sum_{x \in X} 2^{-|x|} \leq 2^{-s}$$

or equivalently,

$$\mu(X) \leq 2^{-s}$$

which means that $\mu(V_n)$ is computable as required. Since

$$G(x \upharpoonright j) \geq h(j)$$

infinitely often, it follows that for all n, there is a j such that

$$G(x \upharpoonright j) \geq h(j) \geq 2^n.$$

Thus $[x \upharpoonright j] \subseteq V_n$, and so for all n, $x \in V_n$.

Conversely, suppose there is a Schnorr test $\{V_n : n \in \omega\}$ such that $x \in \cap_n V_n$. We will build a Schnorr test $\{U_n : n \in \omega\}$ and a function f that satisfy the conditions of Lemma 38. Then if $x \in \cap_n U_n$, Lemma 38 ensures that there is a computable martingale that effectively succeeds on x.

Define a prefix-free set V such that $V = \bigcup_n V_n$. Because each V_n has a computable measure, it is possible for any σ to decide if $(\forall n \leq |\sigma|)\ \sigma \in V_n$, and so V is a computable set. We can construct a computable, unbounded, non-decreasing function f such that

$$\sum_{\substack{\sigma \in V \\ |\sigma| \geq n}} 2^{-|\sigma|} \leq 2^{-2f(n)}.$$

Note that because V is a prefix-free set, we can define $f(0) = 0$, Then for any $n > 0$, we know that

$$\sum_{\substack{\sigma \in V \\ |\sigma| \geq n}} 2^{-|\sigma|} \leq 1 - \sum_{\substack{\sigma \in V \\ |\sigma| < n}} 2^{-|\sigma|}.$$

The computability of V means the upper bound on the right-hand side of the equation can be calculated, and f defined appropriately. Then,

$$\begin{aligned}\sum_{\substack{\sigma\in V\\ f(|\sigma|)=m}} 2^{-|\sigma|}2^{f(|\sigma|)} &= 2^m \sum_{\substack{\sigma\in V\\ f(|\sigma|)=m}} 2^{-|\sigma|}\\ &\leq 2^m 2^{-2m}\\ &\leq 2^{-m},\end{aligned}$$

so

$$\begin{aligned}\sum_{\sigma\in V} 2^{-|\sigma|}2^{f(|\sigma|)} &= \sum_{m\in\omega}\sum_{\substack{\sigma\in V\\ f(|\sigma|)=m}} 2^{-|\sigma|}2^{f(|\sigma|)}\\ &\leq \sum_{m\in\omega} 2^{-m}\\ &\leq 2.\end{aligned}$$

Thus

$$\sum_{\sigma\in V} 2^{-|\sigma|}2^{f(|\sigma|)}$$

is finite and bounded (by 2). It is also non-zero as $V \neq \emptyset$. Furthermore,

$$\begin{aligned}\sum_{\substack{\sigma\in V\\ f(|\sigma|)>n}} 2^{-|\sigma|}2^{f(|\sigma|)} &\leq \sum_{m>n} 2^{-m}\\ &\leq 2^{-n},\end{aligned}$$

so if we define $g(n) = (\mu p)[f(p) \geq n+1]$ (which is computable as f is computable and non-decreasing) it follows that

$$\begin{aligned}\sum_{\substack{\sigma\in V\\ |\sigma|\geq g(n)}} 2^{-|\sigma|}2^{f(|\sigma|)} &= \sum_{\substack{\sigma\in V\\ f(|\sigma|)>n}} 2^{-|\sigma|}2^{f(|\sigma|)}\\ &\leq 2^{-n}.\end{aligned}$$

Define $U_n = \{\sigma \in V : |\sigma| \geq g(n)\}$. Then $S_n \leq 2^{-n}$, and

$$\begin{aligned} \mu(U_n) &= \sum_{\substack{\sigma \in V \\ |\sigma| \geq g(n)}} 2^{-|\sigma|} \\ &\leq 2^{-2f(g(n))} \\ &\leq 2^{-2(n+1)} \\ &\leq 2^{-2n-2} \\ &\leq 2^{-n}. \end{aligned}$$

Also note that

$$\begin{aligned} \sum_{\substack{\sigma \in V \\ |\sigma| \geq g(s)}} 2^{-|\sigma|} &\leq \sum_{\substack{\sigma \in V \\ |\sigma| \geq g(s)}} 2^{-|\sigma|} 2^{f(|\sigma|)} \\ &\leq 2^{-s} \end{aligned}$$

so $\mu(U_n)$ is a computable real. This also implies that S_n is a computable real. Hence U_n and f satisfy the conditions of Lemma 38.

Recall that $x \in \cap_n V_n$. For any n, find a j such that $f(j) \geq n+1$. There is a $\sigma \in V_j$ such that $\sigma \sqsubset x$. As $\mu(V_j) \leq 2^{-j}$ this implies that $|\sigma| \geq j$, and so $f(|\sigma|) \geq n+1$ as f is non-decreasing. Because $g(n)$ is defined as the *least* j for which $f(j) \geq n+1$, it follows that $|\sigma| \geq g(n)$. Hence $\sigma \in U_n$ and so for all n, $x \in U_n$. Thus by Lemma 38 there is a computable martingale that effectively succeeds on x. □

Proposition 39. *A real x is Schnorr random if and only if for all Martin-Löf tests $\{U_n : n \in \omega\}$ such that $\mu(U_n) = 2^{-n}$ for all n, $x \notin \cap_n U_n$.*

Proof. Suppose that there is a test $\{U_n : n \in \omega\}$ such that $\mu(U_n) = 2^{-n}$ for all n and $x \in \cap_n U_n$. Obviously, $\{U_n\}$ is a Schnorr test, and so x is not Schnorr random. Conversely, suppose there is a Schnorr test $\{V_n : n \in \omega\}$ such that $x \in \cap_n V_n$ and $f(n) = \mu(V_n)$ for all n. f is computable, so for any n we can computably find $f(n)$, and we know that $f(n) \leq 2^{-n}$. For any $n > 0$, this can be our U_n, except that it needs some "padding" to make sure its measure is exactly 2^{-n}. (For $n = 0$, we have $U_0 = [\lambda]$). At stage 0, we define $U_n[0] = V_n[0] \cup P_0$ where $\mu(P_0) = 2^{-n} - f(n)$ and $V_n[0]$ and P_0 are disjoint. Obviously, P_n represents the padding at stage n, and it is always possible to pick such an P_n. For each stage $s > 0$, set $P_s = P_{s-1}$ and $U_n[s] = U_n[s-1] \cup V_n[s] \cup P_s$ *unless* something has entered $V_n[s]$ that intersects with P_{s-1}. In that case we have to add a bit more padding to P_s to ensure that $\mu(U_n[s])$ remains at 2^{-n}. Note that if $x \in \cap_n V_n$ it follows that $x \in \cap_n U_n$. □

Furthermore, there is a Solovay characterization of Schnorr randomness:

Definition 34. *A* total Solovay test *is a computable collection of c.e. open sets* $\{U_i : i \in \omega\}$ *such that*

$$\sum_{i=0}^{\infty} \mu(U_i) < \infty$$

and is a computable real. A real x withstands a total Solovay test if $x \in U_i$ for only finitely many U_i.

Theorem 40 (Downey and Griffiths, [12]). *A real x is Schnorr random if and only if x withstands all total Solovay tests.*

Proof. Suppose that x is not Schnorr random, so there is a Schnorr test $\{U_n : n \in \omega\}$ such that $x \in \cap_n U_n$. Obviously $\{U_n : n \in \omega\}$ is a total Solovay test, and x is infinitely many (in fact, all) U_n.

Conversely, suppose that there is a total Solovay test $\{V_k : k \in \omega\}$ with x in infinitely many V_k. Use the method in Theorem 32 to build a Martin-Löf test:

$$U_n = \{y : y \text{ is in at least } 2^n \text{ of the } V_k\}.$$

It remains to be shown that this test is a Schnorr test; that is, $\mu(U_n)$ is a computable real for all n.

We know that $\sum_k \mu(V_k)$ is a computable real, so it is possible to enumerate elements into the V_k's until a stage such that

$$\sum_k \mu(V_k) - \sum_k \mu(V_{k,s}) < 2^{-s}.$$

Define

$$X_s = \bigcup_{k \in \omega} \{\sigma : \sigma \in V_k - V_{k,s}\}.$$

Then $\mu(X_s) < 2^{-s}$. Now

$$\begin{aligned} Y_{n,s} &= \{\sigma : \sigma \in U_n - U_{n,s}\} \\ &= \{\sigma : \sigma \in 2^n\, V_k \text{ but } \sigma \notin 2^n\, V_{k,s}\} \\ &\subseteq \bigcup_k \{\sigma : \sigma \in V_k - V_{k,s}\} \\ &\subseteq X_s. \end{aligned}$$

Thus $\mu(Y_{n,s}) \leq \mu(X_s)$ so $\mu(Y_{n,s}) < 2^{-s}$. Since $\mu(U_n) - \mu(U_{n,s}) = \mu(Y_{n,s})$, it follows that $\mu(U_n)$ is a computable real for all n, and so $\{U_n : n \in \omega\}$ is a Schnorr test. Since x is in infinitely many V_k it follows that x is in every U_n, and so x is not Schnorr random. $\square$

10.2 Computable randomness

Computable randomness is an intermediate step between Martin-Löf and Schnorr randomness. It requires that the martingales used be computable, but doesn't require that the rate of success be effective.

Definition 35. *A real x is computably random if and only if for all computable martingales d, d does not succeed on x.*

Theorem 41 (Lutz, [33], Schnorr, [40], van Lambalgen, [50]). *The following are equivalent:*

1. *For all martingales J such that $-J$ is weakly computable,*

$$\limsup_n J(x \upharpoonright n) < \infty.$$

2. *For all rational martingales $G : 2^{<\omega} \to \mathbb{Q}^+$, $\limsup_n G(x \upharpoonright n) < \infty$.*

3. *x is computably random.*

Proof. ($\neg 1 \Rightarrow \neg 2$) Suppose there is a martingale J such that $-J$ is weakly computable and $\limsup_n J(x \upharpoonright n) = \infty$. Note that if $-J$ is weakly computable it is equivalent to J having a computable sequence of approximations $J_s(x)$, such that for all x, $\lim_s J_s(x) = J(x)$ and $J_s(x) \geq J_{s+1}(x)$. We assume that for all x and for all s $J(x) < J_s(x)$. If this is not the case, a new sequence of approximations $J_s(x) + 2^{-s}$ can be used instead. This sequence *will* be strictly greater than $J(x)$, and will still approach J in the limit. We inductively define a rational martingale G such that for all σ, $G(\sigma) > J(\sigma) + 2^{-|\sigma|}$:

Base: Set $G(\lambda) = J_0(\lambda) + 1$. As $J_0 > J$,

$$G(\lambda) > J(\lambda) + 2^{-|\lambda|}.$$

Induction step: Assume that $G(\sigma)$ has been defined and

$$G(\sigma) > J(\sigma) + 2^{-|\sigma|}.$$

We find the least i such that

$$J_i(\sigma 0) + J_i(\sigma 1) < 2(G(\sigma) - 2^{-|\sigma|}).$$

Note that the induction hypothesis ensures that such an i can be chosen. Then set

$$G(\sigma 1) = J_i(\sigma 1) + 2^{-|\sigma 1|}$$

and

$$G(\sigma 0) = 2G(\sigma) - G(\sigma 1).$$

By construction, G is must be a martingale. As $J_i > J$ for all i,

$$G(\sigma 1) > J(\sigma 1) + 2^{-|\sigma 1|}.$$

We must also prove that $G(\sigma 0) > J(\sigma 0) + 2^{-|\sigma 0|}$. Recall that the choice of i ensures that

$$J_i(\sigma 0) + J_i(\sigma 1) < 2(G(\sigma) - 2^{-|\sigma|}).$$

Rearranging this gives

$$\begin{aligned} J_i(\sigma 0) + 3.2^{-|\sigma|-1} &< 2G(\sigma) - J_i(\sigma 1) - 2^{-|\sigma|-1} \\ &< 2G(\sigma) - (J_i(\sigma 1) + 2^{-|\sigma 1|}) \\ &< 2G(\sigma) - G(\sigma 1) \\ &< G(\sigma 0). \end{aligned}$$

As $J < J_i$ for all i, it follows that $J(\sigma 0) + 2^{-|\sigma 0|} < G(\sigma 0)$, as required.

Since $\limsup_n J(x \upharpoonright n) = \infty$ it is also true that

$$\limsup_n G(x \upharpoonright n) = \infty.$$

$(\neg 3 \Rightarrow \neg 1)$ Suppose we have a computable martingale F such that

$$\limsup_n F(x \upharpoonright n) = \infty.$$

F being computable means that there is a computable function h such that for all n and for all x,

$$|h(n,x) - F(x)| \leq 2^{-n}.$$

We will inductively define a new computable function h_1 such that

$$\lim_n h_1(n,x) = F(x)$$

and $h_1(n,x) \geq h_1(n+1,x)$.

Base: For all x set $h_1(0,x) = h(0,x) + 1$.

Induction step: Suppose that $h_1(n,x)$ is defined. Define $h_1(n+1,x)$ by

$$h_1(n+1,x) = \min\{h_1(n,x), h(n+1,x) + 2^{-(n+1)}\}.$$

Note that $h_1(n+1, x) \leq h_1(n, x)$ by construction. To show that

$$\lim_n h_1(n, x) = F(x),$$

firstly observe that for all n, $h_1(n, x) \leq h(n, x) + 2^{-n}$. Also, as

$$|h(n, x) - F(x)| \leq 2^{-n},$$

$h(n, x) + 2^{-n} \geq F(x)$ for all n. Finally, for all n, $h_1(n, x) \geq F(x)$. Thus we have

$$\begin{array}{rcccl} \lim_{n\to\infty} F(x) & \leq & \lim_{n\to\infty} h_1(n, x) & \leq & \lim_{n\to\infty}[h(n, x) + 2^{-n}] \\ F(x) & \leq & \lim_{n\to\infty} h_1(n, x) & \leq & \lim_{n\to\infty} h(n, x) + \lim_{n\to\infty} 2^{-n} \\ F(x) & \leq & \lim_{n\to\infty} h_1(n, x) & \leq & F(x). \end{array}$$

This "squeezing" technique shows that $\lim_{n\to\infty} h_1(n, x) = F(x)$. This proves that F is a martingale and $-F$ is weakly computable, and for which (by assumption) $\limsup_n F(x \upharpoonright n) = \infty$.

($\neg 2 \Rightarrow \neg 3$) Suppose there is a rational martingale G such that

$$\limsup_n G(x \upharpoonright n) = \infty.$$

G is obviously a computable martingale (approximable by the function $g(n, x) = G(x)$ for all n), and so x is trivially not computably random. $\square$

There is also a test set definition of computable randomness.

Theorem 42 (Downey and Griffiths, [13]). *A Martin-Löf test* $\{U_n : n \in \omega\}$ *is* computably graded *if there is a computable function* $f : 2^{<\omega} \times \omega \to \mathbb{R}$ *such that for all* $n \in \omega$ *and* $\sigma \in 2^{<\omega}$,

$$\mu(U_n \cap [\sigma]) \leq f(\sigma, n),$$

and for any finite prefix-free set of strings $\{\sigma_i : i \in I\}$,

$$\sum_{i \in I} f(\sigma_i, n) \leq 2^{-n}$$

and

$$\sum_{i \in I} f(\sigma_i, n) \leq f(\tau, n),$$

where $\cup_{i \in I}[\sigma_i] \subseteq [\tau]$. *A real* x *is computably random if and only if* x *withstands all computably graded tests.*

Proving this Theorem requires the following Lemma. The proof of this Lemma follows the proof of Lemma 117.

Lemma 43. *Let* X *be a prefix-free set such that* $X \subset [\sigma]$. *Then*

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

We now turn to the proof of the Theorem.

Proof. Firstly, suppose that x is not computably random, so there is a computable martingale G such that $\limsup_n G(x \upharpoonright n) = \infty$. Assume without loss of generality that $G(\lambda) \leq 1$. Define a Martin-Löf test $\{U_n : n \in \omega\}$ by $U_n = \{[\sigma] : G(\sigma) \geq 2^n\}$. For any string σ, $U_n \cap [\sigma]$ represents the strings extending σ where the martingale G is greater than 2^n. Let X_n be a prefix-free set of strings such that $[X_n] = U_n \cap [\sigma]$. Using Lemma 43,

$$\begin{aligned}
\mu(U_n \cap [\sigma]) &= \mu(X_n) \\
&= \sum_{x \in X_n} 2^{-|x|} \\
&= 2^{-n} \sum_{x \in X_n} 2^{-|x|} 2^n \\
&\leq 2^{-n} \sum_{x \in X_n} 2^{-|x|} G(x) \\
&\leq 2^{-n} 2^{-|\sigma|} G(\sigma).
\end{aligned}$$

Set $f(\sigma, n) = 2^{-n} 2^{-|\sigma|} G(\sigma)$ and $\mu(U_n \cap [\sigma]) \leq f(\sigma, n)$ as required for a computably graded test. Furthermore, if $\{\sigma_i : i \in I\}$ then using Lemma 117 gives us

$$\begin{aligned}
\sum_{i=0}^{I} f(\sigma_i, n) &= \sum_{i \in I} 2^{-n} 2^{-|\sigma_i|} G(\sigma_i) \\
&= 2^{-n} \sum_{i \in I} 2^{-|\sigma_i|} G(\sigma_i) \\
&\leq 2^{-n} G(\lambda) \\
&\leq 2^{-n}.
\end{aligned}$$

Finally, if

$$\cup_{i \in I} [\sigma_i] \subseteq [\tau],$$

Lemma 43 gives us

$$\begin{aligned}
\sum_{i \in I} f(\sigma_i, n) &= 2^{-n} \sum_{i \in I} 2^{-|\sigma_i|} G(\sigma_i) \\
&\leq 2^{-n} 2^{-|\tau|} G(\tau) \\
&\leq f(\tau, n).
\end{aligned}$$

Hence we know that $\{U_n : n \in \omega\}$ satisfies all the requirements of a computably graded test. Furthermore, because $\limsup_n G(x \upharpoonright n) = \infty$, it follows that $(\forall j)(\exists n_j) G(x \upharpoonright n_j) \geq 2^j$ and so for all j, $x \in U_j$.

Conversely, suppose there is a computably graded test $\{V_n : n \in \omega\}$ such that $x \in \cap_n V_n$. Let us assume without loss of generality that for all n, $V_{n+1} \subseteq V_n$. For any string σ let $P(\sigma)$ be the set of all prefix-free sets of strings $\{\sigma_i : i \in I\}$ such that $\cup_{i \in I} [\sigma_i] = [\sigma]$. Then define

$$J(\sigma) = 2^{|\sigma|} \sum_{n=0}^{\infty} \inf_{P(\sigma)} \sum_{i \in I} f(\sigma_i, n).$$

Clearly any cover of $[\sigma]$ that isn't σ itself will be a combination of a cover of $[\sigma 0]$ and a cover of $[\sigma 1]$. Suppose firstly that the only cover of $[\sigma]$ that minimizes f is σ itself. That means for any other cover $\{\sigma_i : i \in I\}$ such that $\cup_i [\sigma_i] = [\sigma]$, $f(\sigma, n) < \sum_i f(\sigma_i, n)$. This contradicts the definition of a computably graded test, and hence σ *cannot* be its own least cover. The least cover must be a combination of a cover of $\sigma 0$ and a cover of $\sigma 1$, and so

$$\begin{aligned}\inf_{P(\sigma)} \sum_i f(\sigma_i, n) &= \inf_{P(\sigma 0) \oplus P(\sigma 1)} \sum_i f(\sigma_i, n) \\ &= \inf_{P(\sigma 0)} \sum_i f(\sigma_i, n) + \inf_{P(\sigma 1)} \sum_j f(\sigma_j, n)\end{aligned}$$

by the rules of infimums. This implies that

$$\begin{aligned}2J(\sigma) &= 2\sum_{n=0}^{\infty} 2^{|\sigma|} \inf_{P(\sigma)} \sum_i f(\sigma_i, n) \\ &= \sum_{n=0}^{\infty} 2^{|\sigma|+1} \Big(\inf_{P(\sigma 0)} \sum_i f(\sigma_i, n) + \inf_{P(\sigma 1)} \sum_j f(\sigma_j, n) \Big) \\ &= \sum_{n=0}^{\infty} 2^{|\sigma 0|} \inf_{P(\sigma 0)} \sum_i f(\sigma_i, n) + \sum_{n=0}^{\infty} 2^{|\sigma 1|} \inf_{P(\sigma 1)} \sum_j f(\sigma_j, n) \\ &= J(\sigma 0) + J(\sigma 1),\end{aligned}$$

implying that J is a martingale.

Furthermore, J has a non-increasing approximation to it. Define $P(\sigma, s)$ to be the set of all prefix-free sets of strings $\{\sigma_i : i \in I\}$ such that

$$\cup_{i \in I} [\sigma_i] = [\sigma]$$

and for all i, $|\sigma_i| \leq |\sigma| + s$. Then set

$$J_s(\sigma) = \sum_{n=0}^{s} 2^{|\sigma|} \inf_{P(\sigma,s)} \sum_i f(\sigma_i, n) + 2^{|\sigma|} 2^{-s}.$$

It is clear that $\lim_s J_s(\sigma) = J(\sigma)$. Also,

$$\begin{aligned}
J_{s+1}(\sigma) &= \sum_{n=0}^{s+1} 2^{|\sigma|} \inf_{P(\sigma,s+1)} \sum f(\sigma_i, n) + 2^{|\sigma|}2^{-(s+1)} \\
&= \sum_{n=0}^{s} 2^{|\sigma|} \inf_{P(\sigma,s+1)} \sum f(\sigma_i, n) \\
&\quad + 2^{|\sigma|} \inf_{P(\sigma,s+1)} \sum f(\sigma_i, s+1) + 2^{|\sigma|}2^{-(s+1)} \\
&\leq \sum_{n=0}^{s} 2^{|\sigma|} \inf_{P(\sigma,s+1)} \sum f(\sigma_i, n) + 2^{|\sigma|}2^{-(s+1)} + 2^{|\sigma|}2^{-(s+1)} \\
&\leq \sum_{n=0}^{s} 2^{|\sigma|} \inf_{P(\sigma,s+1)} \sum f(\sigma_i, n) + 2^{|\sigma|}2^{-s} \\
&\leq \sum_{n=0}^{s} 2^{|\sigma|} \inf_{P(\sigma,s)} \sum f(\sigma_i, n) + 2^{|\sigma|}2^{-s} \\
&\leq J_s(\sigma),
\end{aligned}$$

so the J_s's represent a non-increasing approximation to J. Since for all m, $x \in \cap_n V_m$, there must be a k_m such that $[x \upharpoonright k_m] \subseteq V_m$. As the V_n are nested, this implies that $(\forall n \geq m)[x \upharpoonright k_m] \subseteq V_n$. Thus we have

$$\begin{aligned} J(x \upharpoonright k_m) &= 2^{k_m} \sum_{n=0}^{\infty} \inf_{P(x \upharpoonright k_m)} \sum_i f(\sigma_i, n) \\ &\geq 2^{k_m} \sum_{n=0}^{m} \inf_{P(x \upharpoonright k_m)} \sum_i f(\sigma_i, m) \\ &\geq 2^{k_m} \sum_{n=0}^{m} \inf_{P(x \upharpoonright k_m)} \sum_i \mu(V_m \cap [\sigma_i]) \\ &\geq 2^{k_m} \sum_{n=0}^{m} \inf_{P(x \upharpoonright k_m)} \sum_i 2^{-|\sigma_i|} \\ &\geq 2^{k_m} \sum_{n=0}^{m} 2^{-|x \upharpoonright k_m|} \\ &\geq 2^{k_m} \sum_{n=0}^{m} 2^{-k_m} \\ &\geq \sum_{n=0}^{m} 1 \\ &\geq m+1. \end{aligned}$$

Since for all n there is a k_n such that $J(x \upharpoonright k_n) \geq n+1$, it follows that $\limsup_k J(x \upharpoonright k) = \infty$. Hence x is not computably random. $\square$

10.3 Universal tests of randomness

Both Schnorr randomness and computable randomness have a disadvantage compared to Martin-Löf randomness in that there are no universal tests for either.

Theorem 44. *There is no universal computable martingale; that is, there is no martingale F such that x is random if and only if $\limsup_n F(x \upharpoonright n) < \infty$.*

Proof. Lemma 78 states that for any computable martingale, and any finite string y there is a computable real x extending y such that

$$\limsup_n F(x \upharpoonright n) < \infty.$$

This would be true for the universal computable martingale, if it existed. This would imply that there is a computable real x that is computably random. But Lemma 80 ensures that there is no computable real that is computably random. Hence there can be no universal test. $\square$

This Theorem eliminates the possibility of a universal martingale for either Schnorr randomness or computable randomness. Also, if there is no universal martingale test, there can obviously be no universal test set either, as Theorems 42 and 37 would allow us to build corresponding universal martingales. But this contradicts Theorem 44, and so cannot be possible.

Chapter 11

Prefix-free complexity

Although Kolmogorov complexity provided a good theoretic basis of randomness for finite strings, it had not been successful when applied to infinite strings as there are no infinite strings that have high Kolomogorov complexity for *all* initial segments. This problem was finally solved by considering the prefix-free Kolmogorov complexity (called the prefix-free complexity for short).

Recall that a prefix-free set is one for which no element is the prefix of another element.

Definition 36. *A function f is prefix-free if and only if* $\mathrm{dom} f$ *is a prefix-free set of strings. Similarly, a Turing machine M is prefix-free if and only if* $\mathrm{dom} M$ *is a prefix-free set of strings.*

11.1 Prefix-free complexity

The prefix-free complexity was discovered independently by Chaitin [9], Levin [30], and others. More information on the early development of prefix-free complexity can be found in [31]. The definition of the relative prefix-free complexity follows the definition of relative Kolmogorov complexity. The one difference is that the prefix-free complexity must use a prefix-free function to interpret descriptions.

Definition 37. *Let* $f : 2^{<\omega} \to 2^{<\omega}$ *be a prefix-free partial computable function. Let* x *be a (finite) string. The prefix-free complexity of* x *relative to* f *is written* $H_f(x)$ *and is defined by*

$$H_f(x) = \min(\{|p| : f(p) = x\} \cup \{\infty\}).$$

Recall that the existence of an absolute measure of Kolmogorov complexity was proved in Theorem 26. We would hope that $H(x)$ would have a similar Invariance Theorem. However, Theorem 26 relies on the existence of the Universal Turing Machine, so we first need to prove the existence of a Universal Prefix-free Turing Machine.

Theorem 45. *There is an enumeration of all and only the prefix-free Turing machines.*

Proof. Let $T_1, T_2, \ldots$ be an enumeration of the Turing machines. For each i, we will change T_i to a prefix-free Turing machine $\widehat{T_i}$. Furthermore, if T_i is already prefix-free, then $\widehat{T_i} = T_i$. Since all the prefix-free Turing machines are included in the enumeration of the Turing machines, the enumeration $\widehat{T_1}, \widehat{T_2}, \ldots$ will contain all (and only) the prefix-free Turing machines. The prefix-free machines behave as follows: suppose that $\widehat{T_i}$ halts on a string x. If $\widehat{T_i}$ is run on a string y that is a superstring of x, $\widehat{T_i}$ will only read the first $|x|$ bits of y, and will output $\widehat{T_i}(x)$. We call such an x a *halting input* and consider $\mathrm{dom}\widehat{T_i}$ to be the set of all halting inputs. Define $\widehat{T_i}(x)$ as follows:

1. Set $p := \lambda$.

2. For each $q \in 2^{<\omega}$, dovetail on $T(pq)$. Let r be the first string q such that $T(pq)$ halts.

3. If $r = \lambda$, then p is a halting input of $\widehat{T_i}$. Output $\widehat{T_i}(p) = T(p)$ and halt. If not, read the next input bit of x (say x_j), set $p := px_j$ and return to the previous step. If there are no further bits of x, set $\widehat{T_i}(x) \uparrow$.

The following lemmas ensure that $\widehat{T}_1, \widehat{T}_2, \ldots$ is an enumeration of the prefix-free Turing machines. □

Lemma 46. *For each i, $\widehat{T}_i$ is a prefix-free machine.*

Proof. The construction ensures that if $\widehat{T}_i$ halts on a string x it can never halt on a superstring or substring of x. □

Lemma 47. *If T_i is a prefix-free machine, then $\widehat{T}_i = T_i$.*

Proof. Suppose $x \in \mathrm{dom} T_i$. Thus there is no substring $y \sqsubset x$ such that $T_i(y) \downarrow$. For each $j \leq |x|$, define x_j as the first j bits of x. If we run $\widehat{T}_i(x)$, we dovetail on $T_i(\lambda r)$ for all $r \in 2^{<\omega}$, then on $T_i(x_1 r)$, and so on. For each j, $T_i(x_j r)$ will halt for a certain r (if nothing else we know that $T_i(x_j r) \downarrow$ when $x_j r = x$). However, it will never halt for $r = \lambda$ until $x_j = x$ because $\mathrm{dom} T_i$ is prefix-free. Note that because T_i is a prefix-free machine, if $T_i(xr)$ halts, then $r = \lambda$. Hence $\widehat{T}_i(x) \downarrow = T_i(x)$, and

$$x \in \mathrm{dom} T_i \Rightarrow x \in \mathrm{dom} \widehat{T}_i.$$

Conversely, if $x \in \mathrm{dom}\widehat{T}_i$, then by the definition of $\widehat{T}_i$, $T_i(x) \downarrow$ and

$$\widehat{T}_i(x) = T_i(x),$$

as required. □

As for the standard Universal Turing Machine, it is possible to build a Universal Prefix-free Machine from the enumeration of the prefix-free machines $\widehat{T}_1, \widehat{T}_2, \ldots$. We choose a prefix-free set $P_1, P_2, \ldots$ to provide a way to index the prefix-free machines and define U as $U(P_i x) = \widehat{T}_i(x)$. As $\mathrm{dom}\widehat{T}_i$ is a prefix-free set, and so is the set of indices, it follows that $\mathrm{dom}(U)$ is prefix-free as required.

Theorem 48 (The Invariance Theorem). *There is a function g such that for all partial computable functions f and all strings x,*

$$H_g(x) \leq H_f(x) + \mathcal{O}(1).$$

Proof. As for Theorem 26. □

Thus $H_g(x)$ can be called *the* prefix-free complexity. We will write it $H(x)$. We require the following result:

Lemma 49. *For any partial computable function f, if $f(\sigma)\downarrow$ then $H(f(\sigma)) \leq H(\sigma) + \mathcal{O}(1)$.*

Proof. Let U be a universal prefix-free machine. Simulate U until a stage s where $U_s(p)\downarrow= x$ for some p. At that point, compute $f(x)$, and if $f(x)\downarrow$, define $M(p) = f(x)$. Then $\mathrm{dom}(M) = \mathrm{dom}(U)$ so M is a prefix-free machine. Furthermore, for some $p \in \mathrm{dom}(U)$, $|p| = H(x)$, and so for any x for which $f(x)$ terminates:

$$\begin{aligned} H(f(x)) &\leq H_M(f(x)) + \mathcal{O}(1) \\ &\leq |p| + \mathcal{O}(1) \\ &\leq H(x) + \mathcal{O}(1). \end{aligned}$$

The constant factor $\mathcal{O}(1)$ contains the coding constant of M in U, and the finite number of bits needed to describe how to compute f. □

11.2 The Kraft inequality

Demanding that a set be prefix-free puts a severe restriction on the strings that can be used. For instance, putting a string x into a prefix-free set A bars *any* substring or superstring of x from ever entering A. Proved in 1949, the Kraft Inequality is a very important result quantifying exactly how big prefix-free sets can be.

Theorem 50 (The Kraft Inequality, Kraft, [25]). *Let $l_1, l_2 \ldots$ be a (finite or infinite) sequence of natural numbers. There is a prefix-free set A whose elements have precisely these lengths if and only if*

$$\sum_n 2^{-l_n} \leq 1.$$

Proof. One way to understand how the Kraft Inequality works is to recall that finite binary strings σ have a correspondence with subsets of the real interval $[0, 1)$ defined by

$$f : 2^{<\omega} \to \{[a, b] : a, b \in \mathbb{Q}\}$$

such that $a = \lambda(\sigma)$ and $b = \lambda(\sigma) + 2^{-|\sigma|}$. The length of this interval is $2^{-|\sigma|}$. If we have a set of prefix-free strings, none of the corresponding intervals will overlap. Thus the sum of the lengths of the intervals is less than the length of the real interval (that is, less than 1), and so the Kraft Inequality holds. Conversely, if we have a set of lengths such that

$$\sum_n 2^{-l_n} \leq 1,$$

we can choose a prefix-free set whose elements have these lengths. Given l_n, add the length-lexicographically least string of length l_n that is still available (in the sense that none of its substrings or superstrings have already been allocated) to the set. By construction this set is prefix-free. If we relate these strings to their corresponding real intervals, the Kraft Inequality ensures that their (non-overlapping) intervals will all fit into $[0, 1)$ and hence it is always possible to choose a finite binary string of the right length. $\square$

Note that although *all* prefix-free sets satisfiy Kraft's Inequality, it is possible for a set that is not prefix-free to satisfy it also. An example of this is the set $\{0, 00, 11\}$, or even the infinite set $\{0, 01, 011, 0111, \ldots\}$.

If we are effectively given a series of lengths that satisfy Kraft's Inequality, it is possible to effectively produce a prefix-free set whose elements have these lengths. A particularly useful application of this is the effective Kraft-Chaitin algorithm, a method of building prefix-free machines where we want to have a input string of a particular length mapping to a given output string:

Suppose we had an effective list $< n_k, \sigma_k >$ of pairs of strings and natural numbers, such that the n_k satisfied Kraft's Inequality:

$$\sum_{k \in \omega} 2^{-n_k} \leq 1.$$

Then one could define a prefix-free machine M such that for each k there is a string τ_k with length $|\tau_k| = n_k$ and $M(\tau_k) = \sigma_k$.

11.3 Prefix-free complexity and infinite random strings

For our purposes, the important advantage that H has over K, is that reals exists with high complexity on all initial segments. Furthermore, the reals with high initial-segment complexity are precisely the Martin-Löf random reals.

Theorem 51 (Schnorr, see [10]). *A real x is Martin-Löf random if and only if there is a constant c such that for all n,*

$$H(x \upharpoonright n) \geq n - c.$$

Before we prove this theorem, we state two lemmas that are used in the proof of this, and other, theorems. These lemmas are proved in the Appendix.

Lemma 52. *Define $U_k = \{\sigma : H(\sigma) \leq |\sigma| - k\}$. (Here H can refer to absolute or to function-dependent prefix-free complexity). Then $\mu(U_k) \leq 2^{-k}$.*

Lemma 53. *Let U_k be a test set such that $\mu(U_k) \leq 2^{-k}$. U_k can be considered as a collection of extensions of strings,*

$$U_k = \bigcup_{i \in \omega} [\sigma_{k,i}],$$

where $\{\sigma_{k,i} : i \in \omega\}$ is a prefix-free set of strings. Then one can define a prefix free machine M that for all $k, i \in \omega$ maps a string of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$.

Now we turn to the proof of Theorem 51.

Proof. Suppose first of all that for all constants k there is an n such that $H(x \upharpoonright n) < n - k$. Define

$$U_k = \{x : (\exists n) H(x \upharpoonright n) < n - k\}.$$

By Lemma 52, $\mu(U_k) \leq 2^{-k}$. Also, U_k is computably enumerable, so

$$\{U_k : k \in \omega\}$$

is a Martin-Löf test. Furthermore, $(\forall k)(\exists n) H(x \upharpoonright n) < n - k$, so

$$(\forall k)(x \in U_k),$$

and x is not Martin-Löf random, as required.

Conversely, suppose that x is not Martin-Löf random, so that for all k, $x \in U_k$. Using the notation and arguments from Lemma 53, we have a prefix-free machine M mapping strings of length $|\sigma_{2k+2,i}| - k - 1$ to $\sigma_{2k+2,i}$, where $\cup_{i \in \omega}[\sigma_{k,i}]$ is a prefix-free cover of U_k.

$$\begin{aligned}
\text{Now, } (\forall k)(x \in U_k) &\Rightarrow (\forall k)(\exists i)(x \in [\sigma_{2k+2,i}]) \\
&\Rightarrow (\forall k)(\exists i, j)(x \upharpoonright j = \sigma_{2k+2,i}) \\
&\Rightarrow (\forall k)(\exists i, j)(H_M(x \upharpoonright j) = j - k - 1) \\
&\Rightarrow (\forall k)(\exists j)(H(x \upharpoonright j) < j - k),
\end{aligned}$$

from the Invariance Theorem. ☐

11.4 Prefix-free complexity and Schnorr randomness

Schnorr randomness also has a characterization in terms of prefix-free complexity. Instead of requiring that a infinite string x have complex initial segments with relation to *all* prefix-free machines, it only requires that x have complex initial segments with relation to a subset of the prefix-free machines, the so-called *computable* machines.

Definition 38 (Downey and Griffiths, [12]). *A prefix free machine M is called* computable *if and only if*

$$\sum_{\sigma \in \mathrm{dom} M} 2^{-|\sigma|}$$

is a computable real.

Theorem 54 (Downey and Griffiths, [12]). *A real x is Schnorr random if and only if for all computable machines M, there is a constant c such that for all n,*

$$H_M(x \restriction n) \geq n - c.$$

The proof of this theorem of course largely follows the equivalent proof for Martin-Löf randomness, with the added proviso that the measures remain computable. Let

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

where m is a computable real, and define

$$\sum_{\sigma \in \mathrm{dom} M_s} 2^{-|\sigma|} = m_s,$$

where $\mathrm{dom} M_s$ is the set of elements that M has halted on by stage s.

Proof. Suppose that there is a computable machine M such that for every constant c there is an n, such that

$$H_M(x \restriction n) < n - c.$$

Define

$$U_k = \{x : (\exists n) H(x \restriction n) < n - k\}.$$

As in Theorem 51, $\{U_k : k \in \omega\}$ will be a Martin-Löf test, and x will not withstand it. It remains to check that the U_k's form a Schnorr test; that is, for all k, $\mu(U_k)$ is a computable real.

For $\mu(U_k)$ to be a computable real requires that at stage s,

$$|\mu(U_k) - \mu(U_{k,s})| < 2^{-s}.$$

To ensure this, run M until $|m - m_s| < 2^{-s+k}$. Since

$$\sigma \in U_k \Leftrightarrow H(\sigma) < |\sigma| - k$$

it follows that

$$\begin{aligned}
|\mu(U_k) - \mu(U_{k,s})| &= \sum_{\sigma \in U_k - U_{k,s}} 2^{-|\sigma|} \\
&< \sum_{\sigma \in U_k - U_{k,s}} 2^{-H(\sigma)} 2^{-k} \\
&< 2^{-k} \sum_{\sigma \in U_k - U_{k,s}} 2^{-H(\sigma)} \\
&< 2^{-k} \sum_{\sigma \in \mathrm{dom} M - \mathrm{dom} M_s} 2^{-|\sigma|} \\
&< 2^{-k} |m - m_s| \\
&< 2^{-k} 2^{-s+k} \\
&< 2^{-s},
\end{aligned}$$

as required.

Conversely, suppose that x is not Schnorr random, so that there is a Schnorr test $\{U_k : k \in \omega\}$ such that $x \in \cap_k U_k$. As for Theorem 51, we can build a prefix-free machine M relative to which x has initial segments with arbitrarily low complexity. Recall that M maps strings of length $|\sigma_{2k+2,i}| - k - 1$ to $|\sigma_{2k+2,i}|$ for all $k, i \in \omega$. It remains to ascertain that M is a computable machine, that is, that

$$m = \sum_{\sigma \in \mathrm{dom} M} 2^{-|\sigma|}$$

is a computable real. Note that

$$\begin{aligned}
m &= \sum_{\sigma \in \mathrm{dom} M} 2^{-|\sigma|} \\
&= \sum_{k,i} 2^{-(|\sigma_{2k+2,i}| - k - 1)} \\
&= \sum_{k} \mu(U_{2k+2}) 2^{k+1}.
\end{aligned}$$

Suppose we wish to calculate m to within 2^{-s}. To do so, enumerate the Schnorr tests until a stage t when for all $k \leq s$,

$$|\mu(U_{2k+2}) - \mu(U_{2k+2,t})| < \frac{2^{-s-k-2}}{(s+1)}.$$

This is possible because $\mu(U_k)$ is computable for each k. Define m_s to be the measure of the domain of M at this stage.

$$\begin{aligned}
|m - m_s| &= |\sum_k \mu(U_{2k+2})2^{k+1} - \sum_{k \leq s} \mu(U_{2k+2,t})2^{k+1}| \\
&< \sum_{k>s} \mu(U_{2k+2})2^{k+1} + \sum_{k \leq s} 2^{k+1} \frac{2^{-s-k-2}}{(s+1)} \\
&< \sum_{k>s} 2^{-2k-2}2^{k+1} + \sum_{k \leq s} \frac{2^{-s}}{2(s+1)} \\
&< \sum_{k>s} \frac{2^{-k}}{2} + \frac{(s+1)2^{-s}}{2(s+1)} \\
&< \frac{2^{-s}}{2} + \frac{2^{-s}}{2} \\
&< 2^{-s}.
\end{aligned}$$

□

Chapter 12

A random number

12.1 Halting probabilities and randomness

The machine definition of Martin-Löf randomness provides an explicit example of a random number. This random number is Known variously as Chaitin's number, the halting probability, or simply as Ω. It is defined as the measure of the domain of a prefix-free Universal Turing machine:

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

Theorem 55 (Chaitin, [9]). *Ω is a Martin-Löf random real.*

Proof. Let U be a universal self-delimiting machine and let $p_1, p_2, \ldots$ be an enumeration of the c.e. set $\mathrm{dom}(U)$. We compute Ω by adding $2^{-|p_i|}$ to Ω as each p_i is enumerated. Let $\Omega_s = \sum_{U(\sigma)[s]\downarrow} 2^{-|\sigma|}$.

There are 2^n stringsof length n, so if $\Omega_s \upharpoonright n \neq \Omega_{s+1} \upharpoonright n$ more than 2^n times, then either Ω must have decreased (which is impossible as Ω is the measure of a c.e. set) or Ω has exceeded 1 (which again is impossible as Ω is the measure of a prefix-free set). Thus $\Omega_s \upharpoonright n$ can change at most finitely many times, and there must be a stage j such that

$$\Omega \upharpoonright n = \Big(\sum_{i=1}^{j} 2^{-|p_i|}\Big) \upharpoonright n.$$

If we *knew* what $\Omega \upharpoonright n$ was we could calculate what j is, by running the enumeration of p_i's until the right-hand sum reaches $\Omega \upharpoonright n$. Hence there is a partial recursive function $\psi : 2^{<\omega} \to \omega$ such that

$$\Omega \upharpoonright n = \sum_{i=1}^{\psi(\Omega \upharpoonright n)} 2^{-|p_i|}.$$

Obviously,

$$(\forall i > \psi(\Omega \upharpoonright n))|p_i| > n,$$

because otherwise p_i would still have an effect on $\Omega \upharpoonright n$. Thus, any finite binary string x not in the Ω-computable set

$$S_{\Omega \upharpoonright n} = \{U(p_i) : i \leq \psi(\Omega \upharpoonright n)\},$$

will have the property that $H(x) > n$. Define a partial recursive function

$$\Psi : 2^{<\omega} \to 2^{<\omega}$$

that on input $\Omega \upharpoonright n$ selects a string x not in the set $S_{\Omega \upharpoonright n}$. Perhaps Ψ chooses $U(p_j)$ where $j = (\mu i)\big[U(p_i) \notin S_{\Omega \upharpoonright n}\big]$. We know that

$$H(\Psi(\Omega \upharpoonright n)) > n.$$

It follows that

$$n < H(\Omega \upharpoonright n) + c_\Psi,$$

from Lemma 49. Since this is true for all n, Ω must be random. $\square$

Not only does this theorem provide us with an explicit example of a random number, it shows us that it is possible to have a real number that is both computably enumerable and random, as Ω satisfies both these requirements:

Proposition 56. *Ω is a computably enumerable real.*

Proof. If Ω is the halting probability associated with a given universal self-delimiting machine U, let us define $\mathrm{dom}(U_s)$ as the set $\{\sigma : U(\sigma)[s] \downarrow\}$. Without loss of generality we may assume that $|\mathrm{dom}(U_{s+1}) - \mathrm{dom}(U_s)| \neq \emptyset$. Hence define

$$\Omega_s = \sum_{\sigma \in \mathrm{dom}(U_s)} 2^{-|\sigma|}.$$

The sequence $\Omega_0, \Omega_1, \ldots$ is a computable sequence of rationals, and it is increasing by construction. Obviously $\lim_s \Omega_s = \Omega$, and hence Ω must be a c.e. real. $\square$

The halting probability Ω has a very close relationship with the halting problem. It is often said that Ω "codes" the halting problem, meaning that Ω contains all the information necessary to answer the halting problem correctly.

If Ω is known, the question of whether or not $U(\sigma)$ halts can be answered by dovetailing on U. Compute Ω_s (as defined above). If $|\sigma| = n$, continue dovetailing until a stage t such that $\Omega_t \upharpoonright n = \Omega \upharpoonright n$. At this stage, nothing of length n or shorter can enter $\mathrm{dom}(U)$. By inspection of the elements that have entered $\mathrm{dom}(U)$ by stage s, the question "does $U(\sigma)$ halt?" can be decided.

U is a *universal* prefix free machine and so for any prefix free function ϕ_x and input y,

$$\phi_x(y) = U(\bar{x}y).$$

Because the question "$U(\sigma) \downarrow$?" can be decided from Ω, Ω can answer any question of the form "$\phi_x(y) \downarrow$?" In this way, Ω can be used to decide the generalized halting problem for prefix free functions.

12.2 Solovay reducibility and relative randomness

In this section we introduce a new reducibility, Solovay reducibility, that provides a way to measure the relative randomness of two reals. Solovay reducibility is similar to m-reducibility, but for reals instead of sets. Whilst $B \leq_m A$ if there is a computable function f that produces the elements of B given the elements of A, $y \leq_S x$ if there is a computable function f, given a rational approximation to x, produces a rational approximation to y, that converges at least as fast.

Definition 39 (Solovay, [45]). *Let x and y be c.e. reals. We say that y is Solovay reducible to x, or that x dominates y, if there is a constant c and a partial computable function f such that*

$$(\forall q < x)\big[y - f(q) \leq c(x - q)\big].$$

If this is the case we write $y \leq_S x$.

The following Lemma provides another characterization of Solovay reducibility which will be useful in later proofs.

Lemma 57. *Let x and y be c.e. reals. x dominates y if and only if there are computable sequences of rationals $\{x_i : i \in \omega\}$ and $\{y_i : i \in \omega\}$ that converge to x and y respectively, and a constant d such that $d(x - x_i) \geq (y - y_i)$ for all $i \in \omega$.*

Proof. Suppose that y is Solovay reducible to x. Thus there is a constant c and a computable function f such that for all rationals $q < x$,

$$y - f(q) \leq c(x - q).$$

Let $\{x_i : i \in \omega\}$ be a computable sequence of rationals converging to x. (Since x is a c.e. real, such a sequence will exist). Since x_i is a rational and $x_i < x$ for all i, it follows that

$$y - f(x_i) \leq c(x - x_i),$$

and $\{f(x_i) : i \in \omega\}$ is a computable sequence of rationals converging to y, as required.

Conversely, suppose that $\{x_i : i \in \omega\}$ and $\{y_i : i \in \omega\}$ converging to x and y respectively, such that

$$d(x - x_i) \geq y - y_i$$

for each $i \in \omega$. Take any $q < x$, and compute the least i such that $x_i \geq q$. (Since the sequence of x_i's converge to x, such an i will exist). Define $f(q) = y_i$. Thus

$$\begin{aligned} d(x - q) &\geq d(x - x_i) \\ &\geq (y - y_i) \\ &\geq y - f(q), \end{aligned}$$

as required. $\square$

Solovay reducibility allows us to grade the reals more finely than just saying that they are random or non-random. Solovay reducibility can be used to compare two numbers to decide if one is "more" random than the other. In the case of finite strings, the Kolmogorov complexity of the two strings can be compared to see whether one is more compressible than the other. For infinite strings, if the initial segments of y are more compressible that the initial segments of x, that would imply that x is more random than y. The following theorem shows that Solovay reducibility provides a measure of relative randomness:

Theorem 58 (Solovay, [45]). *Let x and y be c.e. reals. If $y \leq_S x$, then for all n,*

$$H(y \upharpoonright n) \leq H(x \upharpoonright n) + \mathcal{O}(1).$$

Before proving this Theorem we require the following Lemma. For the proof of the Lemma, see the Appendix.

Lemma 59. *Let x, y be binary strings such that $|x| = |y| = n$ and $|x - y| \leq 2^{k-n}$ where $k \in \omega$. There is a constant c_k (depending on k) such that $H(y) \leq H(x) + c_k$.*

We now turn to the proof of Theorem 58.

Proof. Suppose that $y \leq_S x$, so there is a computable function f and a constant c such that

$$(\forall q < x)\big[y - f(q) \leq c(x - q)\big].$$

In particular, since for all n, $x \upharpoonright n$ is a rational and $x \upharpoonright n \leq x$,

$$\begin{aligned} y - f(x \upharpoonright n) &\leq c(x - x \upharpoonright n) \\ &\leq c2^{-n}. \end{aligned}$$

Since $y \upharpoonright n \leq y$ and $f(x \upharpoonright n) \leq f(x \upharpoonright n) \upharpoonright n + 2^{-n}$ it follows that

$$(y \upharpoonright n) - (f(x \upharpoonright n) \upharpoonright n) \leq (c+1)2^{-n},$$

so from Lemma 59, there is a constant k such that for all n,

$$H(y \upharpoonright n) \leq H(f(x \upharpoonright n) \upharpoonright n) + k.$$

Then by applying Lemma 49 it follows that for all n,

$$H(y \upharpoonright n) \leq H(x \upharpoonright n) + k,$$

as required. ☐

However, the converse of this theorem is not true. In [3], the authors construct two c.e. reals x and y such that $H(y \upharpoonright n) \leq H(x \upharpoonright n) + \mathcal{O}(1)$ but $y \not\leq_S x$.

12.3 Ω and Solovay completeness

As usual, we define a c.e. real x to be *Solovay complete* if for all c.e. reals y, $y \leq_S x$. As we see in the following theorem, Ω is Solovay complete. Because of this, the Solovay complete reals are often referred to as the Ω-*like* reals.

Theorem 60 (Kučera and Slaman, [27]). *Let x be a c.e. real. Suppose that x is random. Then x is Solovay complete; i.e. for any c.e. real y, $y \leq_S x$.*

Proof. Because x and y are c.e. reals, we have computable sequences of rationals $\{x_i : i \in \omega\}$ and $\{y_i : i \in \omega\}$ converging to x and y respectively. For simplicity assume that both x and y are irrational, and so without loss of generality that we may assume that $y_i < y_{i+1}$ for all $i \in \omega$.

We define a Martin-Löf test $\{U_n : n \in \omega\}$. At stage 0, for all n let $U_n = \emptyset$. For $s > 0$, call s an "n-stage" if $x_s \notin U_n^s$. If s is an n-stage, set

$$U_n^{s+1} = U_n^s \cup (x_s, x_s + 2^{-n}(y_s - y_{s'})),$$

where s' is the last n-stage ($s' = 0$ if there is no previous n-stage). Each U_n is c.e. as x and y are c.e. reals, and (letting $t_0 = 0$ and $t_1, t_2 \ldots$ denote the n-stages)

$$\begin{aligned}
\mu(U_n) &= \sum_{i \geq 1} \mu(x_{t_i}, x_{t_i} + 2^{-n}(y_{t_i} - y_{t_{i-1}})) \\
&= \sum_{i \geq 1} 2^{-n}(y_{t_i} - y_{t_{i-1}}) \\
&\leq 2^{-n}(y - y_0) \\
&\leq 2^{-n},
\end{aligned}$$

as $(y - y_0) \leq 1$. Thus $\{U_n : n \in \omega\}$ is a Martin-Löf test.

By assumption x is random, so $x \notin \cap_n U_n$ and there must be an n such that $x \notin U_n$. This means that for any n-stage t_i there is another n-stage $t_{i+1} > t_i$ such that

$$x_{t_{i+1}} > x_{t_i} + 2^{-n}(y_{t_i} - y_{t_{i-1}}).$$

For any s, let t_j be the last n-stage before s ($t_j = 0$ if there are no such n-stages). Consider the sum

$$\sum_{i>j}^{\infty}(x_{t_{i+1}} - x_{t_i}).$$

From above it follows that

$$\sum_{i>j}^{\infty}(x_{t_{i+1}} - x_{t_i}) > \sum_{i>j}^{\infty} 2^{-n}(y_{t_i} - y_{t_{i-1}})$$

and so

$$x - x_{t_{j+1}} > 2^{-n}(y - y_{t_j}).$$

Because the sequences $\{x_i : i \in \omega\}$ and $\{y_i : i \in \omega\}$ are increasing it follows that $x_s \leq x_{t_{j+1}}$ and $y_s > y_{t_j}$ so

$$x - x_s > 2^{-n}(y - y_s),$$

and hence $y \leq_S x$. □

The converse is also true, giving an exact coincidence between the c.e. random reals and Solovay complete reals: reals:

Corollary 61 (Solovay, [45], Kučera and Slaman, [27]). *Let x be a c.e. real. Then x is random if and only if x is Solovay complete.*

Proof. Theorem 60 ensures that if x is a c.e. random real, then x is Solovay complete. Conversely, let x be Solovay complete; that is, it dominates *all* c.e. reals. In particular, x will dominate Ω. From Theorem 58 this means that $(\forall n) H(\Omega \restriction n) \leq H(x \restriction n) + \mathcal{O}(1)$. Since Ω is random, we know that $(\forall n) H(\Omega \restriction n) \geq n - \mathcal{O}(1)$. Combining these two facts gives us, for all n

$$\begin{aligned} H(x \restriction n) + \mathcal{O}(1) &\geq H(\Omega \restriction n) \\ &\geq n - \mathcal{O}(1). \end{aligned}$$

Hence $H(x \restriction n) \geq n - \mathcal{O}(1)$ for all n, and so x is random, as required. □

The final amazing result about the Ω-like reals is that *every one* of them is in fact the halting probability of a universal prefix-free machine. We showed in Theorem 55 that any halting probability is a c.e. random real; now we see that the c.e. random reals are precisely the halting probabilities.

Theorem 62 (Calude, Hertling, Khoussainov, Wang, [4]). *A real is Ω-like if and only if it is the halting probability of a Universal prefix-free machine.*

To facilitate the proving of this theorem, we require the following Lemma. For the proof of this Lemma, see the Appendix.

Lemma 63 ([4]). *Let x be a c.e. real, and $Y = \{y_0, y_1, y_2, \ldots\}$ be an infinite prefix-free c.e. set. Let*

$$y = \sum_{n \in \omega} 2^{-|y_n|}.$$

If $y \leq_S x$, then we can construct an infinite prefix-free c.e. set X, a partial recursive function f and a constant $c > 0$ such that

$$x = \sum_{\sigma \in X} 2^{-|\sigma|},$$

$\mathrm{dom}(f) = X$, $\mathrm{ra}(f) = Y$, *and* $(\forall \sigma \in X)(|\sigma| \leq |f(\sigma)| + c)$.

We now turn to the proof of Theorem 62.

Proof. Let α be an Ω-like (that is, a Solovay complete) real. Since α dominates *all* c.e. reals, α will dominate Ω in particular, where Ω is the halting probability of a Universal Prefix-free Machine. The domain of U is an infinite, prefix-free, c.e. set, so by Lemma 63 there is an infinite prefix-free c.e. set A such that

$$\alpha = \sum_{\sigma \in A} 2^{-|\sigma|}.$$

There is also a partial recursive function f such that the domain of f is A, and the range of f is $\mathrm{dom}(U)$. Furthermore, for all $\sigma \in A$, $|\sigma| \leq |f(\sigma)| + c$, for some $c > 0$.

Define a machine V such that for any σ,

$$V(\sigma) = U(f(\sigma)).$$

Clearly

$$\begin{aligned} \mathrm{dom}(V) &= \{\sigma : f(\sigma) \in \mathrm{dom}(U)\} \\ &= \mathrm{dom}(f) \\ &= A. \end{aligned}$$

This implies that V is a prefix-free machine, since we have already established that A is a prefix-free set. We also require V to be a universal machine. Recall that U is a universal (prefix-free) machine and so has the property that for every (prefix-free) machine M, there is a constant c_M such that

$$(\forall x \in \mathrm{dom}(M))(\exists y \in \mathrm{dom}(U))U(y) = M(x) \text{ and } |y| \leq |x| + c_M,$$

or put another way

$$(\forall x \in \mathrm{dom}(M))(\exists z \in \mathrm{dom}(V))V(z) = U(f(z)) = M(x) \text{ and } |z| \leq |f(z)| + c.$$

Since

$$\begin{aligned} |z| &\leq |f(z)| + c \\ &\leq |x| + c_M + c, \end{aligned}$$

it follows that V must be a universal machine. Note that by construction α is the halting probability of V, so we are done. □

12.4 Other completeness results

Theorem 64 (Downey). *If*

$$(\forall c)(a.a.\ n)H(x \upharpoonright n) > H(n) + c,$$

then for any c.e. set B, $B \leq_{sw} x$.

Proof. Take any c.e. set B, and let x be as in the statement of the Theorem. For simplicity, let n_c be such that given c,

$$(\forall n \geq n_c)H(x \upharpoonright n) > H(n) + c.$$

We will assume that we are dealing with $n \geq n_c$. We define a reduction Γ^x such that for all n, $\Gamma^x(n) = B[n]$. We also define a prefix-free machine M.

We say that n *requires attention* at stage s if $n \in B$ and $\Gamma^{x_s}(n) = 0$. For all n, let $\Gamma^{x_0}(n) = 0$. At stage $s > 0$, we do the following (in the order prescribed below):

1. If n requires attention, and $x_s \upharpoonright n \neq x_{s-1} \upharpoonright n$, then for all $u \geq s$, set $\Gamma^{x_u}(n) = 1$. Then n no longer requires attention, and

$$\begin{aligned} \Gamma^x(n) &= \lim_{s\to\infty} \Gamma^{x_s}(n) \\ &= 1. \end{aligned}$$

2. If n requires attention, and

$$H_s(x_s \upharpoonright n) > H_s(n) + c$$

   where c is the coding constant for M given by the Recursion Theorem, enumerate the requirement $< H_s(n), x_s \upharpoonright n >$ into M. The construction ensures that $\mathrm{dom}(M) \subseteq \mathrm{dom}(U)$ where U is the universal prefix-free machine, and so M is a prefix-free machine.

Clearly, if $B[n] = 0$ then $\Gamma^x(n) = B[n]$. Furthermore, if $n \downarrow B_s$ and there is a stage $t \geq s$ such that $x_t \upharpoonright n \neq x_s \upharpoonright n$, then $\Gamma^x(n) = B[n]$. We will prove that such a stage exists for all $n \in B$.

Suppose for contradictive purposes that n requires attention at stage s, and

$$(\forall t \geq s)\big[x_t \upharpoonright n = x_s \upharpoonright n\big].$$

By assumption, for any n there will be a stage t such that

$$H_t(x_t \upharpoonright n) > H_t(n) + c.$$

The construction enumerates $< H_t(n), x_t \upharpoonright n >$ into M, and in doing so, forces U to enumerate $< H_t(n) + c, x_t \upharpoonright n >$. Thus

$$H_{t+1}(x_t \upharpoonright n) \leq H_t(n) + c.$$

Since $x_u \upharpoonright n = x_t \upharpoonright n$ for all $y \geq t$, and the prefix-free complexity of a string cannot *increase* over time (recalling that the prefix-free complexity is the *shortest* description of a string), it follows that

$$H_u(x_u \upharpoonright n) \leq H_t(n) + c$$

and so

$$H(x \upharpoonright n) \leq H_t(n) + c.$$

Since we know that $H(x \upharpoonright n) > H(n) + c$ there must be a stage $u \geq t$ such that $H_u(n) \leq H_t(n)$, and $H(x \upharpoonright n) > H_u(n) + c$. However, at this point the construction acts again, and enumerates another requirement into M, so that $H_{u+1}(x \upharpoonright n) \leq H_u(n) + c$. Thus

$$\begin{aligned} H(x \upharpoonright n) &= \lim_{s\to\infty} H_s(x \upharpoonright n) \\ &\leq \lim_{s\to\infty} H_s(n) + c \\ &\leq H(n) + c. \end{aligned}$$

But this is impossible because we know that $H(x \upharpoonright n) > H(n) + c$. Hence we must conclude that $x \upharpoonright n$ cannot remain the same for all time; there must be a stage where Case 1 applies to n, and so $\Gamma^x(n) = 1$ as required. Thus for all $n \geq n_c$, $\Gamma^x(n) = B[n]$.

To correct the finite mistakes below n_c, define for all $\sigma \in 2^{<\omega}$ a corresponding function Φ^x_σ such that

$$\Phi^x_\sigma(n) = \begin{cases} \sigma[n] & \text{if } n \leq |\sigma| \\ \Gamma^x(n) & \text{otherwise.} \end{cases}$$

Then one of these functions will be a correct reduction from x to B. The construction ensures that if $\alpha_s \upharpoonright n = \alpha \upharpoonright n$, then $\Gamma^{\alpha_s \upharpoonright n} = \Gamma^{\alpha \upharpoonright n}$ and so Γ^α is an sw reduction. □

Corollary 65. *For any c.e. set B, $B \leq_{sw} \Omega$.*

We require the following Lemma, which is a simplication of work found in [9]. For the proof of this Lemma, see the Appendix.

Lemma 66. *For any d, $n > H(n) + d$ for almost all n.*

We now turn to the proof of Corollary 65.

Proof. From Theorem 64, we only have to show that for all c, and for almost all n,

$$H(\Omega \upharpoonright n) > H(n) + c.$$

Since Ω is random, we know that there exists a c such that for all n,

$$(\exists c)(\forall n) H(\Omega \upharpoonright n) \geq n - c.$$

From Lemma 66, we have

$$\begin{aligned}(\forall d)(\text{for almost all } n) H(n) + d - c &< n - c \\ &< H(\Omega \upharpoonright n),\end{aligned}$$

as required. □

From this result it automatically follows that for any c.e. set B, $B \leq_{wtt} \Omega$, and $B \leq_T \Omega$. These results were originally proved in [5] and [26] respectively.

Theorem 67 (Bennett, [2]). *If $K \leq_{tt} x$, then x is not random.*

We follow the proof given in [5].

Proof. Let $K \leq_{tt} x$, so there is a computable sequence of functions

$$\{F_i : i \in \omega\}$$

such that for all i, $F_i : 2^{r_i} \to \{0, 1\}$ and

$$K[i] = F_i(x \upharpoonright r_i).$$

We inductively build a Martin-Löf test $\{V_n : n \in \omega\}$ such that $x \in \cap_n V_n$, which demonstrates that x is not random. We define a machine $M : 2^{<\omega} \to (2^{<\omega})^{<\omega}$ via

$$(\forall z \in \omega) M(z) = \{u : F_z(u) = 0\}.$$

Then consider the set

$$S_1 = \{z : \mu(M(z)) \geq \frac{1}{2}\}.$$

S_1 is a c.e. set and so there is a $z_1 \in \omega$ such that $S_1 = W_{z_1}$ where W_e is the e-th c.e. set in the standard enumeration.

Because we are assuming that $K \leq_{tt} x$, it follows that

$$z_1 \in K \iff F_{z_1}(x \upharpoonright r_{z_1}) = 1.$$

Define

$$V_1 = \{u : |u| = r_{z_1}, F_{z_1}(u) = 1 \Leftrightarrow \mu(M(z_1)) \geq \frac{1}{2}\}.$$

Note that if $\mu(M(z_1)) \geq \frac{1}{2}$ then

$$\begin{aligned} \mu(V_1) &= \mu(\{u : F_{z_1}(u) = 1\}) \\ &= 1 - \mu(\{u : F_{z_1}(u) = 0\}) \\ &= 1 - \mu(M(z_1)) \\ &\leq \frac{1}{2}, \end{aligned}$$

and if $\mu(M(z_1)) < \frac{1}{2}$ then

$$\begin{aligned} \mu(V_1) &= \mu(\{u : F_{z_1}(u) = 0\}) \\ &= \mu(M(z_1)) \\ &< \frac{1}{2}. \end{aligned}$$

$M(z_1)$ is a computable set, so V_1 is a computable set. Finally, note that

$$\begin{aligned} F(x \upharpoonright r_{z_1}) = 1 &\iff z_1 \in K \\ &\iff z_1 \in W_{z_1} \\ &\iff \mu(M(z_1)) \geq \frac{1}{2}, \end{aligned}$$

so $x \upharpoonright r_{z_1} \in V_1$.

Now, suppose that V_n and z_n have been defined such that V_n is a finite, computable set with $\mu(V_n) \leq 2^{-n}$, and $x \upharpoonright r_{z_n} \in V_n$. As above, let

$$S_{n+1} = \{z : \mu(M(z) \cap V_n) \geq \frac{1}{2}\mu(V_n)\}.$$

This will be a c.e. set so we can choose z_{n+1} such that $S_{n+1} = W_{z_{n+1}}$. By padding we can ensure that $z_{n+1} \notin \{z_1, \ldots, z_n\}$, and of course

$$z_{n+1} \in K \Longleftrightarrow F_{z_{n+1}}(x \upharpoonright r_{z_{n+1}}) = 1.$$

Define

$$V_{n+1} = \{u : |u| = r_{z_{n+1}}, u \upharpoonright r_{z_n} \in V_n, F_{z_{n+1}}(u) = 1 \Leftrightarrow z_{n+1} \in S_{n+1}\}.$$

V_{n+1} is a finite, computable set, $\mu(V_{n+1}) \leq 2^{-(n+1)}$ and

$$\begin{aligned} z_{n+1} \in S_{n+1} &\Longleftrightarrow z_{n+1} \in W_{z_{n+1}} \\ &\Longleftrightarrow z_{n+1} \in K \\ &\Longleftrightarrow F_{z_{n+1}}(x \upharpoonright r_{z_{n+1}}) = 1, \end{aligned}$$

so $x \upharpoonright r_{z_{n+1}} \in V_{n+1}$. Thus x is not random, as required. $\square$

Since Ω is random, it follows that $K \not\leq_{tt} \Omega$, and so Ω cannot be tt-complete, even for the c.e. sets. However, the above results show that Ω is sw, wtt, and Turing complete for the c.e. sets. The following result shows that Ω is also Turing complete for the c.e. reals.

Theorem 68. *Let x, y be c.e. reals. Then*

$$y \leq_S x \Longrightarrow y \leq_T x.$$

Proof. Suppose that $y \leq_S x$, so there is a constant c and a computable function f such that

$$(\forall q < x)\big[y - f(q) \leq c(x - q)\big].$$

Assume that y is irrational, (if y is rational then y is computable, and the result is obvious) and let $k \in \omega$ such that $c \leq 2^k$. Then

$$(\forall n)\big[y - f(x \upharpoonright n) \leq 2^{k-n}\big].$$

Because y is irrational it is possible to find, for any n, an $m > n$ and a stage $j \geq m$ such that that $f(x \upharpoonright j)[m] = 0$ and

$$y - f(x \upharpoonright j) \leq 2^{-(m+1)}.$$

Then

$$(\forall p < m)\Big[y[p] = f(x \upharpoonright j)[p]\Big].$$

Thus one can define an x-computable function ϕ^x so that on input n, $\phi^x(n)$ finds the least j, m such that $j \geq m > n$ and $f(x \upharpoonright j)[m] = 0$. Then $\phi^x(n) = f(x \upharpoonright j)[n]$. $\square$

Chapter 13

Kurtz randomness

Drawing inspiration from the study of genericity, Stuart Kurtz introduced a new type of randomness in [28]. Whereas Martin-Löf randomness as defined to be those reals not in any effective measure zero set Kurtz defined a random real as one that was in every effective measure one set. This produces a very different class of random reals.

Definition 40 (Kurtz, [28]). *A real x is* Kurtz random *(or* weakly random*) if for every computably enumerable set $U \subseteq 2^\omega$ such that $\mu(U) = 1$, $x \in U$.*

Like Martin-Löf and Schnorr randomness, Kurtz randomness has a characterization in terms of test sets, martingales, Solovay tests, and prefix-free machines.

13.1 Test set characterization

Since Kurtz randomness requires that random reals are in all measure one sets, and the complement of a measure one set is a measure zero set, it seems natural that Kurtz randomness would have a natural characterization in terms of test sets (which require that random reals are in *no* measure zero set). The precise relationship is as follows:

Theorem 69 (Wang, [54]). *A Kurtz null test is a computable set $\{W_i : i \in \omega\}$ where for each i, W_i is a finite, computable set of extensions of strings such that $\mu(W_i) \leq 2^{-i}$. x is Kurtz random if and only if for all Kurtz null tests,*

$$x \notin \cap_i W_i.$$

Proof. Suppose that x is not Kurtz random, as witnessed by the Kurtz test U. From U we can construct a Kurtz null test, $\{W_i : i \in \omega\}$. Enumerate U until a stage s_1 occurs where

$$\mu(U_{s_1}) > \frac{1}{2}$$

and so, obviously,

$$\mu(\overline{U_{s_1}}) \leq \frac{1}{2}.$$

(Because $\mu(U) = 1$, such a stage *will* occur). Define

$$W_1 = \overline{U_{s_1}}.$$

Similarly define $W_n = \overline{U_{s_n}}$ for all $n > 1$, where s_n is the first stage after s_{n-1} where

$$\mu(U_{s_n}) > (1 - 2^{-n}).$$

Each W_n is a finite set of strings, is uniformly computable by the above algorithm, and has measure $\mu(W_n) \leq 2^{-n}$; thus the set $\{W_n : n \in \omega\}$ is a Kurtz null test. Because $x \notin U$, $x \in \overline{U_{s_n}}$ for all n, so $x \in \cap_n W_n$.

Conversely, suppose we have a Kurtz null test $\{W_i : i \in \omega\}$ with

$$x \in \cap_i W_i.$$

We define a Kurtz test as follows:

1. Set $U_0 = \emptyset$.
2. At stage $n \geq 1$, compute W_n and define $U_n = \overline{W_n}$.
3. Set $U = \bigcup_n U_n$.

U is a c.e. open set with $\mu(U) = 1$, and $x \notin U$. Thus x is not Kurtz random, as required. □

Without loss of generality, we can assume that for any i, all strings in W_i are of the same (computable) length $g(i)$. If this is not the case, $g(i)$ can be defined as the length of the longest string in W_i. We then replace any other string σ in W_i by the $2^{g(i)-|\sigma|}$ strings of length $g(i)$ that cover $[\sigma]$.

We can also assume that if $\{W_i : i \in \omega\}$ is a Kurtz null test, then for all i, $W_{i+1} \subseteq W_i$. If not, define a new null test $\widehat{W_i}$ by

$$\widehat{W_i} = \cap_{j \leq i} W_j.$$

Since

$$\cap_i \widehat{W_i} = \cap_i W_i$$

this makes no difference to the Kurtz random reals.

13.2 Martingale characterization

A martingale version of Kurtz randomness can also be formulated. This has interesting parallels to the martingale characterization of Schnorr randomness. Compare Definition 32 to the corresponding martingale characterization of Kurtz randomness:

Definition 41. *A function d* strongly succeeds *on a real x if there is a computable, monotonic and unbounded function $h : \omega \to \omega$ such that*

$$\liminf_n (d(x \upharpoonright n) - h(n)) \geq 0;$$

that is, $d(x \upharpoonright n) \geq h(n)$ for almost all n.

Theorem 70 (Wang, [54]). *A real x is Kurtz random if and only if no computable martingale strongly succeeds on x.*

Note that saying that $d(x \upharpoonright n) \geq h(n)$ for almost all n is equivalent to saying that $d(x \upharpoonright n) \geq h(n)$ for *all* n. Obviously, the second statement implies the first. Conversely, if $d(x \upharpoonright n) \geq h(n)$ for almost all n, this means there is a number n_1, such that for all $n \geq n_1$, $d(x \upharpoonright n) \geq h(n)$. Replace $h(n)$ by the similarly computable, unbounded and nondecreasing function $\widehat{h}(n)$:

$$\widehat{h}(n) = \begin{cases} h(n) & \text{if } n \geq n_1 \\ 0 & \text{otherwise.} \end{cases}$$

Then for all n, $d(x \upharpoonright n) \geq \widehat{h}(n)$. Hence it can be assumed without loss of generality that $d(x \upharpoonright n) \geq h(n)$ for all n.

We now turn to the proof of Theorem 70.

Proof. Suppose that x is not Kurtz random; as witnessed by the Kurtz null test $\{W_i : i \in \omega\}$. Assume that for all i, each string in W_i has length $g(i)$. Define

$$d(\tau) = \sum_{n,m} w_{n,m}(\tau)$$

where $w_{n,m}(\tau)$ is the weighting function:

$$w_{n,m}(\tau) = \begin{cases} 1 & \text{if } \sigma_{n,m} \sqsubseteq \tau \\ 2^{-(|\sigma_{n,m}|-|\tau|)} & \text{if } \tau \sqsubset \sigma_{n,m} \\ 0 & \text{otherwise.} \end{cases}$$

It is straightforward to check that $w_{n,m}$ satisfies the martingale property, and so d will also satisfy it. d is computable, since we can compute $d(\tau)$ to 2^{-s}, by finding k where $g(k) > |\tau| + s$, and setting

$$d_s(\tau) = \sum_{n \leq k,m} w_{n,m}(\tau).$$

Then

$$\begin{aligned}
|d(\tau) - d_s(\tau)| &= \sum_{n>k,m} w_{n,m}(\tau) \\
&\leq \sum_{n>k} 2^{-(g(n)-|\tau|)} \\
&\leq 2^{|\tau|} \sum_{n>k} 2^{-n} \\
&\leq 2^{|\tau|} 2^{-k} \\
&\leq 2^{-s},
\end{aligned}$$

as required. d strongly succeeds on x as for any n,

$$\begin{aligned}
d(x \upharpoonright n) &\geq \sum_{j:g(j)\leq n} 1 \\
&\geq |\{j : g(j) \leq n\}|.
\end{aligned}$$

If we let $h(n) = |\{j : g(j) \leq n\}|$, we have an unbounded, non-decreasing function that is defined for all n. $h(n)$ is computable, because for any n, $g(n) \geq n$. Hence it is only necessary to calculate the first n (computable) terms of g before $h(n)$ is known. Furthermore, $d(x \upharpoonright n) \geq h(n)$ for all n, as required.

Conversely, suppose that there is a computable martingale d and computable, unbounded nondecreasing function h such that

$$(\forall n)(d(x \upharpoonright n) \geq h(n)).$$

For each k, define c_k to be the least n such that $h(n) \geq 2^{k+1}$. Because of the nature of h, c_k exists and can be computed for each k. Define

$$U_k = \{\sigma : |\sigma| \leq c_k \text{ and } d(\sigma) > 2^k\}.$$

Each U_k is a finite set, is computable as d is a computable martingale, and has measure $\mu(U_k) \leq 2^{-k}$, so the set $\{U_k : k \in \omega\}$ is a Kurtz null test. We have that for all n,

$$d(x \upharpoonright n) \geq h(n).$$

In particular, for all k:

$$\begin{aligned} d(x \upharpoonright c_k) &\geq h(c_k) \\ &\geq 2^{k+1}. \end{aligned}$$

Thus $x \in \cap_k U_k$, and so x is not Kurtz random. $\square$

13.3 Computably layered machines

We can define Kurtz randomness in terms of initial-segment complexity.

Definition 42. *An computably layered machine is a prefix-free machine where there is a related computable function*

$$f : \omega \to (2^{<\omega})^{<\omega}$$

such that

- $\bigcup_i f(i) = \mathrm{dom} M$.
- *If* $\gamma \in f(i+1)$, *then* $(\exists \tau \in f(i))$, *such that* $M(\tau) \sqsubseteq M(\gamma)$.
- *If* $\gamma \in f(i)$, *then* $|M(\gamma)| = |\gamma| + i + 1$.

The idea of a computably layered machine is that each layer of the *domain*, $f(i)$, provides a layer of the *range*, and the range elements just become more refined as i increases.

Theorem 71 (Downey, Griffiths, Reid, [14]). *x is Kurtz random if and only if for all computably layered machines M,*

$$(\exists d)(\forall n)(K_M(x \upharpoonright n) \geq n - d).$$

Proof. Suppose that x is not Kurtz random, as witnessed by a Kurtz null test $\{W_i : i \in \omega\}$. Without loss of generality suppose that $\{W_i : i \in \omega\}$ is nested and

$$(\forall i)\sigma \in W_i \Rightarrow |\sigma| = g(i),$$

where g is a monotonically increasing function. By appealing to Lemma 52 there is a prefix-free machine mapping strings of length $(|\sigma_{2n+2,i}| - (n+1))$ to $\sigma_{2n+2,i}$ for $n \in \omega$, where

$$\bigcup_{i=0}^{m_n} [\sigma_{n,i}] = W_n.$$

Let us denote this machine M.

We define f by

$$f(n) = \{\tau : (\exists i)(M(\tau) = \sigma_{2n+2,i})\}.$$

This satisfies all the requirements for f:

- f is computable; to calculate $f(n)$, firstly compute all $\sigma_{2n+2,i}$. $f(n)$ consists of all strings that map to these σ. For each i, the string mapping to $\sigma_{2n+2,i}$ has length

  $$|\sigma_{2n+2,i}| - (n+1),$$

  so by emulating the the effective Kraft-Chaitin algorithm that was used to produce M we can list the domain elements.

- By the definition of f,

  $$\mathrm{dom}(M) = \bigcup_{n \in \omega} f(n).$$

- Since the Kurtz null test is nested, it follows that for all τ in $f(n+1)$, there exists a $\widehat{\tau}$ in $f(n)$ such that $[M(\tau)] \subseteq [M(\widehat{\tau})]$, that is,

  $$M(\widehat{\tau}) \sqsubseteq M(\tau).$$

- Suppose $\tau \in f(n)$. This implies that $M(\tau) = \sigma_{2n+2,i}$ for some i, and hence (by the definition of M) that

  $$\begin{aligned} |\tau| &= |\sigma_{2n+2,i}| - (n+1) \\ &= |M(\tau)| - (n+1) \end{aligned}$$

  as required.

Thus M is a computably layered machine. Since $(\forall n)(x \in W_n)$, it follows that $(\forall n)(\exists i)(\sigma_{2n+2,i} = x \upharpoonright k)$, where $k = |\sigma_{2n+2,i}|$. By the definition of M is follows that

$$(\forall n)(\exists k)(K_M(x \upharpoonright k) < k - n).$$

Conversely, suppose that given x there is some computably layered machine M with a corresponding computable function f such that:

$$(\forall d)(\exists n)(K_M(x \upharpoonright n) < n - d).$$

Define

$$W_k = \{[\sigma] : K_M(\sigma) < |\sigma| - k)\}.$$

Suppose $\sigma \in W_k$. Then $K(\sigma) < |\sigma| - k$, so if $M(\tau) = \sigma$, then

$$|\tau| < |M(\tau)| - k,$$

so τ cannot be an element of $f(j)$ for any $j < k$. Thus

$$W_k = \bigcup_{\substack{j \geq k \\ \gamma \in f(j)}} [M(\gamma)].$$

However, suppose $\gamma \in f(j)$ where $j > k$. Then by iterating the definition of f, it follows that there is a $\tau \in f(k)$ such that $M(\tau) \sqsubseteq M(\gamma)$. So the definition of W_k simplifies to:

$$W_k = \bigcup_{\gamma \in f(k)} [M(\gamma)].$$

Since $f(k)$ is finite and computable, W_k must be finite and computable, and since from Lemma 52 we have that for all k, $\mu(W_k) \leq 2^{-k}$, $\{W_k : k \in \omega\}$ is a Kurtz null test. Since

$$(\forall k)(\exists n)(K_M(x \upharpoonright n) < n - k)$$

it follows that $(\forall k)(x \in W_k)$, and so x is not Kurtz random. $\square$

13.4 Computable machines and Kurtz randomness

It is an indication of the close relationship between Schnorr and Kurtz randomness that this same class of machines can be used to characterize both types of randomness. Recall that Schnorr randomness can be characterized in terms of a certain class of machines known as *computable* machines. A real x is *Schnorr* random if and only if for all such computable machines M, there is a constant c such that for all n, $K_M(x \upharpoonright n) \geq n - c$. Computable machines also characterize Kurtz randomness according to the following theorem:

Theorem 72 (Downey, Griffiths, Reid, [14]). *x is* not *Kurtz random if and only if there is a computable machine M and a computable function $f : \omega \to \omega$ such that:*

$$(\forall d)(K_M(x \upharpoonright f(d)) < f(d) - d).$$

Proof. Suppose that x is not Kurtz random, so there is a Kurtz null test $\{W_n : n \in \omega\}$, such that $(\forall n)(x \in W_n)$. Each W_n is a finite, computable set of extensions of strings: $W_n = \{[\sigma_{n,i}] : i \leq m_n\}$. Assume that

$$(\forall n)(\forall i \leq m_n)(|\sigma_{n,i}| = g(n))$$

where $g : \omega \to \omega$ is a computable function. As in Lemma 53, the effective Kraft-Chaitin algorithm ensures there is a prefix-free machine M mapping strings of length $|\sigma_{2n+2,i}| - (n+1)$ to $\sigma_{2n+2,i}$, for each $n \in \omega$ and $i \leq m_n$.

Let

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

Also let

$$m_s = \sum_{\sigma \in \mathrm{dom} M_s} 2^{-|\sigma|}$$

where

$$\sigma \in \mathrm{dom} M_s \iff M(\sigma) \in W_{2n+2} \text{ and } n \leq s.$$

Then

$$\begin{aligned}
|m - m_s| &= \sum_{\sigma \in \mathrm{dom} M} \sum_{n>s} \sum_{M(\sigma) \in W_{2n+2}} 2^{-|\sigma|} \\
&= \sum_{n>s} \sum_{i \le m_n} 2^{-|\sigma_{2n+2,i}|} 2^{n+1} \\
&= \sum_{n>s} \mu(W_{2n+2}) 2^{n+1} \\
&\le \sum_{n>s} 2^{-2n-2} 2^{n+1} \\
&\le \sum_{n>s} 2^{-n-1} \\
&< 2^{-s},
\end{aligned}$$

so M is a computable machine.

Since $(\forall n)(x \in W_{2n+2})$, for each n there is an $i \le m_n$ such that

$$x \in [\sigma_{2n+2,i}],$$

or put another way,

$$(\forall n)(x \upharpoonright g(2n+2) = \sigma_{2n+2,i}).$$

Thus

$$(\forall n) K_M(x \upharpoonright g(2n+2)) < g(2n+2) - n,$$

as required.

Conversely, suppose we have a computable machine M and a computable function f such that $(\forall d) K(x \upharpoonright f(d)) < f(d) - d$, and we wish to show x is not Kurtz random. Let

$$V_d = \{[\sigma] : |\sigma| = f(d), K_M(\sigma) < f(d) - d\}$$

It is a standard result that $\mu(V_d) \le 2^{-d}$ (see Lemma 52). Furthermore, as there are $2^{f(d)}$ strings of length $f(d)$, V_d cannot contain more than $2^{f(d)}$ strings, and hence must be finite. V_d is computable; because M is a computable machine, it can be run until all strings in the domain of M that have length $f(d) - d$ or less are known, and at that stage it is known which strings are in V_d. Hence V_d is a Kurtz null test. By the condition on x it follows that for all d,

$$x \upharpoonright f(d) \in V_d.$$

so $x \in \cap_d V_d$ and therefore x is not Kurtz random. ☐

13.5 Solovay tests and Kurtz randomness

A variant of the Solovay characterization of randomness can be applied to Kurtz randomness as follows:

Definition 43. *A Kurtz-Solovay test is a pair (f, V) consisting of a computable function $f : \omega \to \omega$ and a computable collection of finite, computable sets*

$$\{V_i : i \in \omega\}$$

where the sum

$$\sum_{i=0}^{\infty} \mu(V_i)$$

is finite and a computable real. x fails *a Kurtz-Solovay test if for all n, x is in at least n of $V_0, \ldots, V_{f(n)}$.*

Theorem 73 (Downey, Griffiths, Reid, [14]). *A real x is Kurtz random if and only if there is no Kurtz-Solovay test (f, V) that x fails.*

Proof. Suppose that x is not Kurtz random, so there is some Kurtz null test $\{W_i : i \in \omega\}$ such that $x \in \cap_i W_i$. This null test is finite and computable, and since

$$\sum_{i=0}^{\infty} \mu(W_i) \leq \sum_{i=0}^{\infty} 2^{-i},$$

the sum of the measures is finite and a computable real, since $\mu(W_i)$ is computable for all i. x is in all of the W_i, so if f is defined to be the identity function, then x is in n of

$$W_0, \ldots, W_{f(n)}$$

for each n, and hence x fails the Kurtz-Solovay test (id, W).

Conversely, suppose that x fails a Kurtz-Solovay test, (f, V). Assume without loss of generality that the sum of the measures is no greater than 1. Define

$$S_k = \{y : y \text{ is in } 2^k \text{ of } V_0, \ldots, V_{f(2^k)}\}$$

and let

$$W_k = \bigcup_{y \in S_k} [y].$$

Now

$$\mu(W_k) = \sum_{y \in S_k} 2^{-|y|},$$

and because each of these y appears in a set V_n at least 2^k times, we know that

$$\begin{aligned} \sum_{n=0}^{\infty} \mu(V_n) &\geq \sum_{y \in S_k} 2^k 2^{-|y|} \\ &\geq 2^k \mu(W_k). \end{aligned}$$

Since, by assumption, $\sum_{n=0}^{\infty} \mu(V_n) \leq 1$, it follows that

$$1 \geq 2^k \mu(W_k),$$

and hence that

$$\mu(W_k) \leq 2^{-k}.$$

W_k will contain finitely many strings as at most it can contain the strings in the (finite) sets $V_0, \ldots, V_{f(2^k)}$. W_k is computable as the sets $V_0, \ldots, V_{f(2^k)}$ are all computable. Hence W_k is a Kurtz null test. Since x fails the Kurtz-Solovay test (f, V), this means that for all n, x is in at least n of

$$V_0, V_1, \ldots, V_{f(n)}.$$

In particular this means that for all k, x is at least 2^k of

$$V_0, V_1, \ldots, V_{f(2^k)}.$$

Hence for all k, $x \in W_k$, and so x is not Kurtz random, as required. □

We observe that given any Kurtz-Solovay test, (f, V), a new Kurtz-Solovay test $(id, \widehat{V})$ can be defined via

$$\widehat{V}_k = V_{f(k)} \cup \ldots \cup V_{f(k+1)-1}.$$

13.6 Kurtz null tests and Schnorr randomness

Schnorr randomness can be characterised in terms of Kurtz null tests.

Definition 44. *A Kurtz array is a uniform collection of Kurtz null tests*

$$\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geq 1\}$$

with the property that

$$(\forall j)(\forall n)(\mu(W_{j,n}) \leq 2^{-j}2^{-n}).$$

Theorem 74 (Downey, Griffith, Reid, [14]). *x is Schnorr random if and only if for all Kurtz arrays $\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geq 1\}$,*

$$(\exists n)(\forall j)(x \notin W_{j,n}).$$

Proof. Suppose x is not Schnorr random, so there is a Schnorr test

$$\{U_n : n \in \omega\},$$

such that $(\forall n)(x \in U_n)$. We define a Kurtz array such that

$$(\forall n)(\exists j)(x \in W_{j,n}),$$

as follows:

Let U_n^s denote U_n at stage s. Define s_j to be the least stage s such that

$$\mu(U_n^s) \geq \frac{2^{j+1}-1}{2^{j+1}}\mu(U_n).$$

Since $\mu(U_n)$ is computable, $\mu(U_n^s)$ is a computable value.

Define

$$W_{j,n} = U_n^{s_j} - U_n^{s_{j-1}}$$

for all n and all $j \geq 1$. All the sets $W_{j,n}$ are finite and computable. Furthermore,

$$\begin{aligned} \mu(W_{j,n}) &= \mu(U_n^{s_j} - U_n^{s_{j-1}}) \\ &= \mu(U_n^{s_j}) - \mu(U_n^{s_{j-1}}), \end{aligned}$$

because $U_n^{s_{j-1}} \subset U_n^{s_j}$. Thus

$$\begin{aligned} \mu(W_{j,n}) &\leq \mu(U_n^{s_j}) - \frac{2^j - 1}{2^j}\mu(U_n) \\ &\leq \mu(U_n) - \frac{2^j - 1}{2^j}\mu(U_n) \\ &\leq \mu(U_n)[1 - 1 + 2^{-j}] \\ &\leq \mu(U_n)2^{-j} \\ &\leq 2^{-n}2^{-j}, \end{aligned}$$

so the sets $W_{j,n}$ form a Kurtz array. Since $(\forall n)(x \in U_n)$ and $\cup_j W_{j,n} = U_n$, it follows that $(\forall n)(\exists j)(x \in W_{j,n})$, as required.

Conversely, suppose we have a Kurtz array

$$\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geq 1\}$$

and x such that $(\forall n)(\exists j)(x \in W_{j,n})$. We define $U_n = \cup_{j\in\omega} W_{j,n}$. Then certainly $(\forall n)(x \in U_n)$. If we show that $\{U_n : n \in \omega\}$ is a Schnorr test, we are done. Note that

$$\begin{aligned} \mu(U_n) &= \sum_{\substack{j\in\omega \\ j\geq 1}} \mu(W_{j,n}) \\ &\leq \sum_{\substack{j\in\omega \\ j\geq 1}} 2^{-j}2^{-n} \\ &\leq 2^{-n}, \end{aligned}$$

as required. Furthermore, $\mu(U_n)$ is computable, since to compute it to within 2^{-s} can be done from the first $s+1$ (computable) sets $\{W_{j,n} : j \in \omega\}$. Hence U_n is a Schnorr test, and x is not Schnorr random. $\square$

Part III

The classes of random numbers

Chapter 14

The basic classes

In this section we consider the classes generated from the characterizations of randomness defined in Part II. Let **M-RAND** denote the class of Martin-Löf random reals; that is, $x \in$ **M-RAND** if and only if x is Martin-Löf random. We know that **M-RAND** is not empty. (It was proved in Chapter 12 that $\Omega \in$ **M-RAND**). However, recalling that the random sequences are the "typical" sequences - non-random strings are those that have some special property - we would expect just about every real to be random. And indeed the set **M-RAND** does contains all but a measure zero set of reals.

Theorem 75. $\mu($**M-RAND**$) = 1$.

Proof. Let $\{U_n : n \in \omega\}$ be a Universal Martin-Löf test. Recall that

$x \in$ **M-RAND** if and only if $x \notin \cap_n U_n$.

Since

$$\sum_n \mu(U_n) < \infty$$

the First Borel-Cantelli Lemma (Proposition 20) ensures that $\mu(\cap_n U_n) = 0$. Since the complement of every measure zero set is a measure one set, it follows that

$$\mu(\{x : x \notin \cap_n U_n\}) = 1$$

and hence that **M-RAND** has measure one. □

Let **COMP-RAND** denote the class of computably random reals, **S-RAND** denote the class of Schnorr random reals, and **K-RAND** denote the class of Kurtz random reals. From the above theorem we can conclude that

$$\mu(\textbf{COMP-RAND}) = \mu(\textbf{S-RAND}) = \mu(\textbf{K-RAND}) = 1,$$

because each of these sets contain **M-RAND** as a subset. In fact:

Theorem 76. **M-RAND** $\subseteq$ **COMP-RAND** $\subseteq$ **S-RAND** $\subseteq$ **K-RAND**.

Proof.

1. Let $x \in$ **M-RAND**. This means no weakly computable martingale succeeds on x. Since if a martingale is computable then it is certainly weakly computable, this implies that no computable martingale succeeds on x. Hence $x \in$ **COMP-RAND**.

2. Let $x \in$ **COMP-RAND**. This means no computable martingale succeeds on x (in any manner at all). Thus is is certainly impossible that a computable martingale could succeed *effectively* on x. Hence $x \in$ **S-RAND**.

3. Let $x \in$ **S-RAND**. Then, by definition, no computable martingale F has the property that there is a computable, nondecreasing, unbounded function h such that $F(x \upharpoonright n) \geq h(n)$ infinitely often. Thus it is certainly the case that no computable martingale is greater than such an $h(n)$ almost always. Hence $x \in$ **K-RAND**.

□

Furthermore, in each case the inclusion is proper; that is:

$$\textbf{M-RAND} \subset \textbf{COMP-RAND} \subset \textbf{S-RAND} \subset \textbf{K-RAND}.$$

Hence Martin-Löf, computable, Schnorr and Kurtz randomness are distinct concepts, as depicted in Figure 14.1. In the remainder of this chapter we prove these claims.

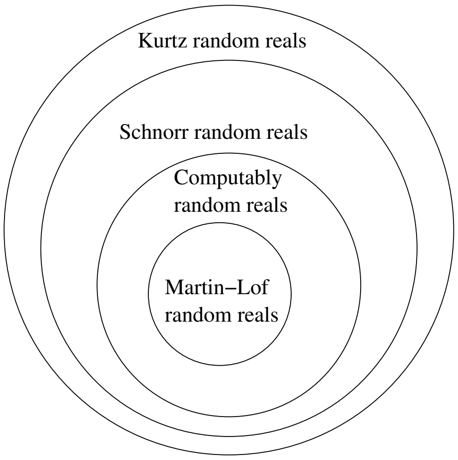

Figure 14.1: Relationships between randomness classes

14.1 Martin-Löf randomness and computable randomness

Theorem 77 (Schnorr, [39], [40]). M-RAND $\subset$ COMP-RAND.

Theorem 76 ensures that **M-RAND** $\subseteq$ **COMP-RAND**. It remains to show that the inclusion is proper; that is, there is a computably random real that is not Martin-Löf random.

First we state some useful lemmas. For the proofs of these lemmas, see the Appendix.

Lemma 78. *Let F be a computable martingale, and $\sigma \in 2^{<\omega}$ a finite string. There is a computable real y extending σ such that for all $n \geq |\sigma|$,*

$$F(y \restriction n) \leq F(\sigma).$$

Lemma 79 (Modification of Lemma 78). *Let $F_0, F_1, \ldots, F_m$ be a sequence of computable martingales, $c_0, c_1, \ldots, c_m$ a sequence of constants, and $\sigma \in 2^{<\omega}$ a finite string. Then there is a computable real y extending σ such that for all $n \geq |\sigma|$,*

$$\sum_{i=0}^{m} c_i F_i(y \restriction n) \leq \sum_{i=0}^{m} c_i F_i(\sigma).$$

Lemma 80. *If a real is computable then it is not Martin-Löf random.*

Lemma 81. *For all i, let f_i be a function such that for all $x \in \mathrm{dom}(f_i)$,*

$$f_i(x) \in \mathbb{R}^+ \cup \{0\}.$$

Then

$$\limsup_n \sum_{i=0}^{\infty} f_i(n) < \infty \Rightarrow (\forall i) \limsup_n f_i(n) < \infty.$$

We now turn to the proof of Theorem 77.

Proof. Let $\{F_i : i \in \omega\}$ be a (non-effective) enumeration of the computable martingales. We assume without loss of generality that $F_i(\lambda) \leq 1$ for all i. Also, let G be the universal Martin-Löf martingale. We inductively define a real z that is computably random, but not Martin-Löf random. Our induction hypothesis is that so far we have $z \upharpoonright n_k$, which has been defined to have the following properties:

1. $G(z \upharpoonright n_k) > k$.
2. For all $j \leq n_k$,
$$\sum_{i=0}^{k} 2^{-n_i - i} F_i(z \upharpoonright j) \leq \sum_{i=0}^{k} 2^{-i+1}.$$

Base step: Let $k = 0$, $n_0 = 0$. It is obvious that these properties are satisfied.

Induction step: Because F_{k+1} is a martingale, for all $j \leq n_k$,

$$F_{k+1}(z \upharpoonright j) \leq 2^{n_k},$$

and hence
$$2^{-n_k-(k+1)} F_{k+1}(z \upharpoonright j) \leq 2^{-(k+1)}.$$

Then for *any* $n_{k+1} \geq n_k$, we have

$$2^{-n_{k+1}-(k+1)} F_{k+1}(z \upharpoonright j) \leq 2^{-(k+1)}.$$

By combining this fact with the induction hypothesis we get

$$(\forall j \leq n_k) \sum_{i=0}^{k+1} 2^{-n_i - i} F_i(z \upharpoonright j) \leq \sum_{i=0}^{k+1} 2^{-i+1}.$$

Then by Lemma 79 there is a computable real y extending $z \upharpoonright n_k$ such that for all $j > n_k$,

$$\sum_{i=0}^{k+1} 2^{-n_i - i} F_i(y \upharpoonright j) \leq \sum_{i=0}^{k+1} 2^{-n_i - i} F_i(z \upharpoonright n_k).$$

Because y is computable, Lemma 80 ensures that there must be a value $n_{k+1} > n_k$ where $G(y \upharpoonright n_{k+1}) > k + 1$. We choose this to be our n_{k+1} and define $z \upharpoonright n_{k+1} = y \upharpoonright n_{k+1}$. Then the induction hypothesis is satisfied for $k+1$, as required. Furthermore, $G(z \upharpoonright n)$ is unbounded as n goes to infinity, so z cannot be Martin-Löf random. On the other hand, we have that for all $m \in \omega$, and for all $j \leq n_m$,

$$\sum_{i=0}^{m} 2^{-n_i - i} F_i(z \upharpoonright j) \leq \sum_{i=0}^{m} 2^{-i+1}.$$

Note that as $m \to \infty$, $n_m \to \infty$ as n_m is a strictly increasing sequence of natural numbers. Hence as we let m go to infinity we get

$$\begin{aligned}(\forall j) \sum_{i=0}^{\infty} 2^{-n_i - i} F_i(z \upharpoonright j) &\leq \sum_{i=0}^{\infty} 2^{-i+1} \\ &\leq 4.\end{aligned}$$

Thus

$$\limsup_j \sum_{i=0}^{\infty} 2^{-n_i - i} F_i(z \upharpoonright j) < \infty,$$

so by Lemma 81, for all i,

$$\limsup_j F_i(z \upharpoonright j) < \infty.$$

Thus z is computably random, as required. □

14.2 Computable randomness and Schnorr randomness

Theorem 82 (Wang, [54], [55]). COMP-RAND $\subset$ S-RAND.

Theorem 76 ensures that **COMP-RAND** $\subseteq$ **S-RAND**. It remains to show that the inclusion is proper. To do this we build a real that is Schnorr random but not computably random.

This proof is more difficult than the previous proof for two reasons. The first reason is that in Theorem 77 it was only necessary to show that no computable martingale succeeded on x. In this theorem each martingale is paired with a computable bounding function that it may not exceed. This adds some extra technicalities.

However, the real added difficulty is the lack of a universal computable martingale. In the proof of Theorem 77 it was only necessary to show that a real x existed that satisfied the requirements. Here, we also need to construct a computable martingale that succeeds on x. The construction of x does not need to be computable, but the construction of the martingale does. We cannot just state that a martingale exists, as at each stage in the proof, we need to make a decision on what values the martingale will take.

With this in mind, instead of using an imaginary, non-effective enumeration of all the computable martingale tests we use a completely computable enumeration of all pairs of computable functions $\{(F_i, h_i) : i \in \omega\}$, where $F_i : 2^{<\omega} \to \mathbb{Q}^+ \cup \{0\}$ and $h_i : \omega \to \omega$. (Theorem 41 ensures that we only need to consider the *rational* martingales). Each of these function pairs has the potential to be a martingale test, however we cannot know for sure whether they are. Then at each stage, we define more bits of our computable martingale F. Our (non-computable) definition of x then depends on whether or not (F_i, h_i) is in fact a martingale test.

We call a martingale test (F, h) *standard* if $F(\lambda) \leq 1$. Within our effective sequence of pairs of computable functions $(F_0, h_0), (F_1, h_1), \ldots$, there is a (non-effective) subsequence $(F_{n_0}, h_{n_0}), (F_{n_1}, h_{n_1}), \ldots$ of all the standard martingale tests. From this we define the following (non-computable) function Φ by

$$\Phi(\sigma) = \sum_{i=0}^{\infty} 2^{-n_i} F_{n_i}(\sigma).$$

Φ is a martingale as for all i, F_{n_i} is a martingale. The construction also requires an additional partial recursive function $d : 2^{<\omega} \to 2^{<\omega}$.

Construction.

Stage 0*:* Set

$$F(\lambda) = F(0) = F(1) = 1,$$

and

$$d(\lambda) = \lambda, d(0) = 0, d(1) = 1.$$

Stage $s > 0$*:* Consider each string σ such that $|\sigma| = s$ and both $F(\sigma 0)$ and $F(\sigma 1)$ are as yet undefined. Then act depending on which of the following three cases applies.

- **Case 1:** If $d(\sigma)$ is undefined, set

$$F(\sigma 0) = F(\sigma 1) = F(\sigma).$$

- **Case 2:** If for all $j \leq |d(\sigma)|$ such that $d(\sigma)[j] = 1$, there exists a $k \leq |\sigma|$ such that $h_{j-1}(k)[s] \downarrow$ and

$$2^{(j-1)+|d(\sigma)|+3} F(\sigma) < h_{j-1}(k)[s],$$

then set

$$\begin{aligned} F(\sigma 0) &= 0 \\ F(\sigma 1) &= 2F(\sigma) \\ F(\sigma 10) &= 2F(\sigma) \\ F(\sigma 11) &= 2F(\sigma), \end{aligned}$$

and

$$\begin{aligned} d(\sigma 1) &= d(\sigma) \\ d(\sigma 10) &= d(\sigma)0 \\ d(\sigma 11) &= d(\sigma)1. \end{aligned}$$

- **Case 3:** Otherwise, set

$$F(\sigma 0) = F(\sigma 1) = F(\sigma),$$

and

$$d(\sigma 0) = d(\sigma 1) = d(\sigma).$$

We define a real x inductively as follows:
Stage 0*:*

$$x \restriction 1 = \begin{cases} 1 & \text{if } (F_0, h_0) \text{ is a standard martingale test} \\ 0 & \text{otherwise.} \end{cases}$$

Stage $s > 0$*:* If Case 2 applies to $x \restriction s$, let $k = |d(x \restriction s)|$ and define

$$x \restriction (s+2) = \begin{cases} (x \restriction s)11 & \text{if } (F_k, h_k) \text{ is a standard martingale test} \\ (x \restriction s)10 & \text{otherwise.} \end{cases}$$

If Case 3 applies to $x \restriction s$ then set

$$x \restriction (s+1) = \begin{cases} (x \restriction s)1 & \text{if } \Phi((x \restriction s)1) \leq \Phi((x \restriction s)0) \\ (x \restriction s)0 & \text{otherwise.} \end{cases}$$

Verification. We prove that x is Schnorr random, but not computably random.

i. *For all* n*, both* $x \restriction n$ *and* $d(x \restriction n)$ *are defined.*

Proof. Proof by induction.

Base case: Let $n = 1$. From the construction, $x \restriction 1$ is defined, and as both $d(0)$ and $d(1)$ are defined, so is $d(x \restriction 1)$.

Induction step: Suppose that $x \upharpoonright k$ and $d(x \upharpoonright k)$ are defined. Thus Case 1 will never apply to $x \upharpoonright k$, and so $x \upharpoonright (k+1)$ is well-defined. If Case 2 applies to $x \upharpoonright k$, then the construction ensures that $d(x \upharpoonright (k+1))$ and $d(x \upharpoonright (k+2))$ are defined, as required.

If Case 3 applies to $x \upharpoonright k$ then $d(x \upharpoonright (k+1))$ is well-defined by the construction. $\square$

ii. *For all n, and for all $j \leq |d(x \upharpoonright n)|$),*

$$d(x \upharpoonright n)[j] = 1 \iff (F_{j-1}, h_{j-1}) \text{ is a standard martingale test.}$$

Proof. This is proved by induction on n.

Base case: Let $n = 1$. Then

$$\begin{aligned} d(x \upharpoonright 1) = 1 \quad & \text{if and only if} \quad x \upharpoonright 1 = 1 \\ & \text{if and only if} \quad (F_0, h_0) \text{ is a standard martingale test.} \end{aligned}$$

Induction step: Assume that for all $j \leq |d(x \upharpoonright k)|$, $d(x \upharpoonright k)[j] = 1$ if and only if (F_{j-1}, h_{j-1}) is a standard martingale test.

If Case 2 applies to $x \upharpoonright k$ then $d(x \upharpoonright (k+1)) = d(x \upharpoonright k)$, so the result holds for $x \upharpoonright (k+1)$.

Let $p = |d((x \upharpoonright k)|$. Then

$$d(x \upharpoonright (k+2)) = \begin{cases} d((x \upharpoonright k)1 & \text{if } (F_p, h_p) \text{ is a stand. mart. test} \\ d((x \upharpoonright k)0 & \text{if } (F_p, h_p) \text{ is not a stand. mart. test.} \end{cases}$$

Thus $d(x \upharpoonright (k+2))[p+1] = 1$ if and only if (F_p, h_p) is a standard martingale test. The induction hypothesis ensures that the claim holds for all other $j \leq d(x \upharpoonright (k+2))$.

If Case 3 applies to $x \upharpoonright k$ then $d(x \upharpoonright (k+1)) = d(x \upharpoonright k)$ and so we are done. $\square$

iii. $\lim_{n\to\infty} F(x \upharpoonright n) = \infty$.

Proof. Since $F(x \upharpoonright 1) = 1$ and

$$(\forall s)F(x \upharpoonright (s+1)) \geq F(x \upharpoonright s),$$

if we can show that

$$(\exists^\infty s)F(x \upharpoonright (s+1)) > F(x \upharpoonright s),$$

then we are done. This occurs whenever Case 2 applies to $x \upharpoonright s$, so we have to show that Case 2 applies to $x \upharpoonright s$ infinitely often.

We showed above that for all s, if $d(x \upharpoonright s)[j] = 1$, (F_{j-1}, h_{j-1}) is a standard martingale test, and so h_{j-1} is an unbounded, non-decreasing function. Thus for each such j there is a t_j and a k_j such that

$$h_{j-1}(k_j)[t_j] \downarrow> 2^{(j-1)+|d(x\upharpoonright s)|+3}F(x \upharpoonright s).$$

Go to the first stage $s_0 > s$ where for all $j < |d(x \upharpoonright s)|$ such that

$$d(x \upharpoonright s)[j] = 1,$$

there is a $k_j < s_0$ such that

$$h_{j-1}(k_j)[s_0] \downarrow> 2^{(j-1)+|d(x\upharpoonright s)|+3}F(x \upharpoonright s).$$

s_0 is chosen to be the *least* such stage, so Case 3 applies to all stages between s and s_0. Thus $F(x \upharpoonright s_0) = F(x \upharpoonright s)$ and $d(x \upharpoonright s_0) = d(x \upharpoonright s)$. Thus Case 2 applies to $x \upharpoonright s_0$, and

$$F(x \upharpoonright s_0 + 1) = 2F(x \upharpoonright s_0) = 2F(x \upharpoonright s).$$

Since such an s_0 can be found for any stage s, it follows that F must be unbounded on x. $\square$

iv. *x is not computably random.*

Proof. The construction ensures that F is a computable martingale. $\square$

v. *For all* $n \in \omega$, $\Phi(x \upharpoonright n) \leq 2^{|d(x \upharpoonright n)|+1} F(x \upharpoonright n)$.

Proof. This is proved by induction on n.

Base case: Let $n = 0$. Then

$$\begin{aligned}
\Phi(\lambda) &= \sum_{i=0}^{\infty} 2^{-n_i} F_{n_i}(\lambda) \\
&\leq \sum_{i=0}^{\infty} 2^{-n_i} \\
&\leq 2 \\
&\leq 2^{|d(\lambda)|+1} F(\lambda).
\end{aligned}$$

Induction step: Assume that for $n = k$, $\Phi(x \upharpoonright k) \leq 2^{|d(x \upharpoonright k)|+1} F(x \upharpoonright k)$. If Case 2 applies to $x \upharpoonright k$ then

$$\begin{aligned}
|d(x \upharpoonright (k+1))| &= |d(x \upharpoonright k)| \\
F(x \upharpoonright (k+1)) &= 2F(x \upharpoonright k).
\end{aligned}$$

Because Φ is a martingale, $\Phi(x \upharpoonright (k+1)) \leq 2\Phi(x \upharpoonright k)$, and so by application of the induction hypothesis

$$\Phi(x \upharpoonright (k+1)) \leq 2^{|d(x \upharpoonright (k+1))|+1} F(x \upharpoonright (k+1)).$$

Furthermore,

$$\begin{aligned}
|d(x \upharpoonright (k+2))| &= |d(x \upharpoonright k)| + 1 \\
F(x \upharpoonright (k+2)) &= 2F(x \upharpoonright k),
\end{aligned}$$

and again the induction hypothesis ensures that

$$\Phi(x \upharpoonright (k+2)) \leq 2^{|d(x \upharpoonright (k+2))|+1} F(x \upharpoonright (k+2)).$$

If Case 3 applies to $x \upharpoonright k$, then

$$\begin{aligned} |d(x \upharpoonright (k+1))| &= |d(x \upharpoonright k)| \\ F(x \upharpoonright (k+1)) &= F(x \upharpoonright k), \end{aligned}$$

and $x \upharpoonright (k+1)$ is defined so that $\Phi(x \upharpoonright (k+1))$ is minimal. Because Φ is a martingale, for any string σ the least of $\Phi(\sigma 0)$ and $\Phi(\sigma 1)$ must be no greater than $\Phi(\sigma)$. Thus

$$\begin{aligned} \Phi(x \upharpoonright (k+1)) &\leq \Phi(x \upharpoonright k) \\ &\leq 2^{|d(x \upharpoonright k)|+1} F(x \upharpoonright k) \\ &\leq 2^{|d(x \upharpoonright (k+1))|+1} F(x \upharpoonright (k+1)), \end{aligned}$$

from the induction hypothesis. □

vi. *For each i, if (F_i, h_i) is a standard martingale test, then*

$$2^{i+|d(x \upharpoonright n)|+1} F(x \upharpoonright n) < h_i(n)$$

for almost all n.

Proof. Since Case 2 applies to $x \upharpoonright n$ for infinitely many n, it follows that $\lim_n |d(x \upharpoonright n)| = \infty$. Thus for any i there is a stage s such that $|d(x \upharpoonright s)| > i$. Because Case 2 applies to x infinitely many times, we can find a $t > s$, such that Case 2 applies to $x \upharpoonright t$. In particular, this means that

$$2^{i+|d(x \upharpoonright t)|+3} F(x \upharpoonright t) < h_i(m_i)$$

where $m_i < t$, and so

$$2^{i+|d(x \upharpoonright t)|+3} F(x \upharpoonright t) < h_i(t)$$

as h_i is non-decreasing. We prove by induction that the claim holds for any $n \geq t$.

Base case: Let $n = t$. Then it is clear from the choice of t that

$$2^{i+|d(x\restriction t)|+1}F(x \restriction t) < h_i(t).$$

Induction step: Assume that

$$2^{i+|d(x\restriction k)|+1}F(x \restriction k) < h_i(k)$$

where $k \geq t$. If Case 2 applies to $x \restriction k$, then since (F_i, h_i) is a standard martingale test we know that there is some $p \leq k$ such that

$$2^{i+|d(x\restriction k)|+3}F(x \restriction k) < h_i(p)[k].$$

Thus

$$\begin{aligned} 2^{i+|d(x\restriction(k+1))|+1}F(x \restriction (k+1)) &= 2.2^{i+|d(x\restriction k)|+1}F(x \restriction k) \\ &< h_i(p) \\ &< h_i(k+1). \end{aligned}$$

Similarly,

$$\begin{aligned} 2^{i+|d(x\restriction(k+2))|+1}F(x \restriction (k+2)) &= 4.2^{i+|d(x\restriction k)|+1}F(x \restriction k) \\ &< h_i(p) \\ &< h_i(k+2). \end{aligned}$$

If Case 3 applies to $x \restriction k$, then

$$\begin{aligned} 2^{i+|d(x\restriction(k+1))|+1}F(x \restriction (k+1)) &= 2^{i+|d(x\restriction k)|+1}F(x \restriction k) \\ &< h_i(k) \\ &< h_i(k+1), \end{aligned}$$

from the induction hypothesis. $\square$

vii. *For each i, if (F_i, h_i) is a standard martingale test, then F_i does not effectively succeed on x. That is,*

$$F_i(x \upharpoonright n) < h_i(n),$$

for almost all n.

Proof. Suppose that (F_i, h_i) is a standard martingale test. We showed above that there is an $s \in \omega$ such that for all $n \geq s$,

$$2^{i+|d(x\upharpoonright n)|+1} F(x \upharpoonright n) < h_i(n).$$

Thus for all $n \geq s$,

$$\begin{array}{rcll} F_i(x \upharpoonright n) & \leq & 2^i \Phi(x \upharpoonright n) & \text{by definition of } \Phi. \\ & \leq & 2^i 2^{|d(x\upharpoonright n)|+1} F(x \upharpoonright n) & \text{by (v).} \\ & \leq & h_i(n). & \text{by (vi).} \end{array}$$

□

Thus we may conclude that Theorem 82 holds.

14.3 Schnorr randomness and Kurtz randomness

Theorem 83 (Kurtz, [28]). S-RAND $\subset$ K-RAND.

Here we follow the method used in Kautz [23]. This is a miniaturized and slightly altered version of the proof, which appear in a more original form in Chapter 15. (See Theorem 90). We need to build a Schnorr test $\{S_e : e \in \omega\}$ and a real x such that $x \in \cap_e S_e$ and x is in no co-c.e. measure zero set. This latter condition ensures that x must be in every c.e. set of measure one and hence that x is Kurtz random. The proof requires a finite injury priority argument.

We enumerate the co-c.e. sets as $\{P_e : e \in \omega\}$, so that if W_e is a c.e. set, then $P_e = \overline{W_e}$. If $W_e[s]$ is the c.e. set at stage s, then let $P_e[s] = \overline{W_e[s]}$.

Obviously, $P_e[s+1] \subseteq P_e[s]$, and $\lim_{s\to\infty} \mu(P_e[s]) = \mu(P_e)$. We wish to build a string $\tau = \lim_s \tau[s]$ such that $(\forall e)(\tau \in S_e)$, and the following infinite requirements are satisfied:

$$R_e : \mu(P_e) = 0 \to \tau \notin P_e.$$

The R_e have the following priority: $R_0, R_1, \ldots$.

At stage s, a set P_e may have a finite string, denoted $\sigma_e[s]$, assigned to it as a *witness*. Furthermore, P_e has a corresponding theshold function $t_e[s]$. If $\mu(P_e[s]) \geq t_e[s-1]$ then we know that $\mu(P_e[s]) \neq 0$. However, if at some stage $\mu(P_e[s])$ drops below the threshold functions, P_e may be a measure zero set. In this case we must act to ensure τ will not enter P_e.

Definition 45. *We say that P_e* requires attention *at stage s if one of the following conditions hold:*

1. *P_e has no witness.*

2. *P_e has a witness and $\mu(P_e[s]) < t_e[s-1]$.*

Construction. At stage 0,

- $S_i[0] = \emptyset$ for all i.

- $P_i[0]$ has no witness for all i.

- $\tau[0] = \lambda$.

- $t_i[0]$ is undefined for all i.

At stage $s > 0$, find the least e such that P_e requires attention. If P_e requires attention via case 1, assign a witness $\sigma_e[s]$ to P_e. This witness must be such that $\tau[s-1] \sqsubset \sigma_e[s]$ and $|\sigma_e[s]| \geq \max(2e, s)$.

- Set $S_e[s] = S_e[s-1] \cup [\sigma_e[s]]$. For all $i \neq e$, set $S_i[s] = S_i[s-1]$.

- For all $i \neq e$, $\sigma_i[s] = \sigma_i[s-1]$.

- Set $\tau[s] = \sigma_e[s]$.
- Set $t_e[s] = 2^{-(|\sigma_e[s]|+1)}$. For all $i \neq e$, $t_i[s] = t_i[s-1]$.

If P_e requires attention via case 2, unassign all witnesses from P_i, where $i \geq e$. Define a new witness $\sigma_e[s]$ for P_e, such that $\sigma_e[s-1] \sqsubset \sigma_e[s]$ and $P_e[s] \cap [\sigma_e[s]] = \emptyset$. The construction ensures that

$$\begin{array}{rcl} \mu(P_e[s]) & < & t_e[s-1] \\ & < & 2^{-(|\sigma_e[s-1]|+1)} \\ & < & \frac{1}{2}\mu([\sigma_e[s-1]]), \end{array}$$

so it will always be possible to choose a suitably long string that satisfies both conditions.

- For all i, set $S_i[s] = S_i[s-1]$.
- For all $i < e$, $\sigma_i[s] = \sigma_i[s-1]$. For all $i > e$, $\sigma_i[s]$ is undefined.
- Set $\tau[s] = \sigma_e[s]$.
- Set $t_e[s] = 0$. For all $i < e$, $t_i[s] = t_i[s-1]$. For all $i > e$, $t_i[s]$ is undefined.

Verification. Define $\tau = \lim_s \tau[s]$.

- *For all e, P_e requires attention only finitely many times. Precisely, it can require attention up to 2^e times via case 1 and up to 2^e times via case 2.*

  Proof. Assume that for all $k \leq e$, the lemma holds. Consider P_{e+1}. After each time one of the P_k requires attention via case 2, P_{e+1} may require attention via case 1. Furthermore, before any of the P_k require attention via case 2, P_{e+1} may also require attention via case 1 once. The total number of times it may require attention via case 1 is therefore

$$\begin{aligned} 1+\sum_{k=0}^{e} 2^k &= 1+(2^{e+1}-1) \\ &= 2^{e+1}, \end{aligned}$$

as required. Furthermore, each time a new witness is assigned to P_{e+1}, it can require attention via case 2 at most once. Thus P_{e+1} only requires attention via case 2 at most 2^{e+1} times, as required. □

- *For all e, $t_e = \lim_s t_e[s]$ and $\sigma_e = \lim_s \sigma_e[s]$ exist.*

  Proof. Let s be a stage such that for all $k \leq e$, P_k will not require attention at any stage $t \geq s$. Then P_e must have a witness, $\sigma_e[s]$, otherwise P_e would require attention via case 1. Since no P_k requires attention after stage s, this witness is immortal. Similarly, $t_e[s]$ only changes when there is some P_k that requires attention. Since this never happens after stage s, it follows that for all $t \geq s$, $t_e[t] = t_e[s]$; that is, $\lim_t t_e[t] = t_e[s]$. □

- *Suppose that P_e requires attention at stage s. Then for all $i > e$, $\sigma_i[s]$ is undefined.*

  Proof. This is obvious from the construction. □

- *Let σ_e be P_e's immortal witness, and let t be a stage after which P_e does not require attention. Then for all $s > t$, $\sigma_e \sqsubset \sigma_{e+1}[s]$.*

  Proof. As t is the final stage where P_e requires attention, $\sigma_e = \sigma_e[t]$ and $\sigma_{e+1}[s]$ is undefined. Hence P_{e+1} will require attention via case 1 at stage $t+1$, and $\sigma_{e+1}[t+1]$ is chosen such that $\tau[t] \sqsubset \sigma_{e+1}[t+1]$. Since by construction $\tau[t] = \sigma_e[t]$, it follows that $\sigma_e[t] \sqsubset \sigma_{e+1}[t+1]$. If P_{e+1} never requires attention again, the proof is done. Otherwise, there may be a stage $s > t + 1$ such that P_{e+1} requires attention via case 2. If that is so, $\sigma_{e+1}[s]$ will be P_{e+1}'s immortal witness, and as by construction it is chosen to be a superstring of $\sigma_{e+1}[t + 1]$, the proof holds in this case also. $\square$

- *Let σ_e and t be as above. If $i > e$, then for all $s > t$ such that $\sigma_i[s]$ is defined, $\sigma_e \sqsubset \sigma_i[s]$.*

  Proof. By induction. $\square$

- $\tau = \lim_s \tau[s]$ *exists.*

  Proof. Let t be a stage after which P_e does not require attention. At any stage $s > t$, the least i such that P_i requires attention will be greater than e. $\tau[s]$ is defined as $\sigma_i[s]$, and as $\sigma_e \sqsubset \sigma_i[s]$, it follows that for all $s > t$, $\sigma_e \sqsubset \tau[s]$. By construction, $\lim_e |\sigma_e| \to \infty$, so for any n there is a stage t such that

  $$(\forall s > t)(\tau[s] \upharpoonright n = \tau[t] \upharpoonright n).$$

  Consequently, $\tau \upharpoonright n = \tau[t] \upharpoonright n$. $\square$

- $\{S_e : e \in \omega\}$ *is a Schnorr test.*

  Proof. For each e, S_e is a c.e. set defined in the construction above. Elements only enter S_e when P_e requires attention via case 1, which occurs at most 2^e times. The element which enters S_e at this stage, $\sigma_e[s]$, is chosen to be of length greater than $2e$. It follows that

  $$\begin{aligned} \mu(S_e) &\leq 2^e 2^{-2e} \\ &\leq 2^{-e}, \end{aligned}$$

  so $\{S_e : e \in \omega\}$ is a Martin-Löf test. Furthermore, $\sigma_e[s]$ is also of length greater than s, so after stage s, $\mu(S_e)$ can increase by at most

  $$\sum_{t>s} 2^{-t} = 2^{-s}.$$

Hence $\mu(S_e)$ is a computable function, and $\{S_e : e \in \omega\}$ is a Schnorr test as required. $\square$

- *$\tau \in \cap_e S_e$ and hence, τ is not Schnorr random.*

  Proof. For any e, let σ_e be P_e's immortal follower. If σ_e was defined via case 1, then by construction $\sigma_e \downarrow S_e$. If σ_e was defined via case 2, then by construction there is a $\hat{\sigma}_e \sqsubset \sigma_e$ such that $\hat{\sigma}_e \downarrow S_e$. In either case, $[\sigma_e] \subseteq S_e$. Since $\tau \in [\sigma_e]$, it follows that $\tau \in S_e$, as required. $\square$

- *For all e, R_e is satisfied.*

  Proof. Let stage t be the first stage such that for all $k < e$, P_k will never again require attention. Thus $\sigma_e[t-1]$ is undefined, and P_e requires attention via case 1. We set $t_e[t] > 0$ and $\tau[t] = \sigma_e[t]$.

  If P_e never requires attention again, then

  $$(\forall s > t)\mu(P_e[s]) > t_e[t].$$

  Since $t_e[t]$ is non-zero, this means that $\lim_s \mu(P_e[s]) > 0$, and R_e is trivially satisfied.

  It is possible that P_e may require attention once more after stage t, if $\mu(P_e[s])$ has drops below the threshold level $t_e[s-1] = t_e[t]$. In this case it is possible that P_e *is* a measure zero set, but $\tau[s]$ is defined so that $\tau[s] \notin P_e[s]$. This is the final time that P_e requires attention, so $\tau \sqsubset \tau[s]$. For all $r > s$, $P_e[r] \subseteq P_e[s]$, so τ can never enter P_e. $\square$

Thus τ is a Kurtz random real, and we may conclude that Theorem 83 holds.

Chapter 15

n-randomness

The randomness test sets considered in Part II are based on computably enumerable, or Σ_1^0 sets. The c.e. sets are just the first level in the arithmetic hierarchy, and a more general definition of randomness could be arrived at simply by by allowing a test to be a collection of Σ_n^0 sets, rather than Σ_1^0 sets.

Definition 46 (Solovay, [10]). *A real x is n-random if for all computable collections of Σ_n^0 sets, $\{U_k : k \in \omega\}$ such that $\mu(U_k) \leq 2^{-k}$, $x \notin \cap_k U_k$.*

Martin-Löf randomness is the $n = 1$ case of this and so is often referred to as 1-randomness in the literature. There is a corresponding characterization involving Solovay-style tests:

Theorem 84. *A Solovay n-test is a computable collection of Σ_n^0 sets*

$$\{U_n : n \in \omega\}$$

such that $\sum_{n\in\omega} \mu(U_n) < \infty$. A real x is n-random if and only if for any Solovay n-test $\{U_n : n \in \omega\}$, x is in at most finitely many U_n.

Proof. As for Theorem 32. Note that if we define

$$U_k = \{y : y \text{ is in at least } 2^k \text{ of the } V_i\},$$

then

$$x \in U_k \iff (\exists n_1)(\exists n_2)\ldots(\exists n_{2^k})\Big[x \in V_{n_1} \text{ and } x \in V_{n_2} \text{ and } \ldots \text{ and } x \in V_{n_{2^k}}\Big]$$

and thus since each of the V_i are Σ^0_n sets, quantifier contraction ensures that U_k is a Σ^0_n set also. □

Recall that a function d is A-computable if there is a Turing machine ϕ with oracle A that computes d. We extend this definition in the following way:

Definition 47. *A function d is A-c.e. if there is an A-computable binary function g such that*

$$(\forall s)g(x,s) \leq g(x,s+1)$$

and

$$\lim_s g(x,s) = d(x).$$

Theorem 85. *A real x is n-random if and only if no $\emptyset^{(n-1)}$-c.e. martingale d succeeds on x.*

Proof. Follows Theorem 33. □

Similarly, Kurtz randomness is based on a Kurtz test - a computably enumerable set. This can be extended to a Kurtz n-test, by requiring that the set be Σ^0_n:

Definition 48 (Kurtz, [28]). *A weak n-test is a Σ^0_n set U such that $\mu(U) = 1$. A string x is weakly n-random if for all weak n-tests U, $x \in U$.*

Thus Kurtz randomness as discussed in Chapter 13 is simply weak 1-randomness. The definitions of Schnorr and computable randomness can be extended to n-random versions in a similar way.

15.1 The n-randomness hierarchy

Let n-**RAND** be the class of all n-random reals, and n-**W-RAND** be the class of all weakly n-random reals. Since for any n, a Σ^0_n set can be considered a Σ^0_{n+1} set by the addition of a dummy variable, it follows that the class

$$n+1\textbf{-RAND} \subseteq n\textbf{-RAND}.$$

The same applies to weak n-randomness. Furthermore, as the name weak n-randomness suggests, n-**RAND** is contained in n-**W-RAND**.

Theorem 86 (Kurtz,[28]). n-**RAND** $\subseteq$ n-**W-RAND**.

Proof. Suppose that x is not weakly n-random, so there is a Σ^0_n set U such that $\mu(U) = 1$, and $x \notin U$. This implies that $x \in \overline{U}$. $\overline{U}$ is a Π^0_n set with measure zero, so according to Proposition 16(v), we can write

$$\overline{U} = \bigcap_j T_j$$

where for all j, T_j is a Σ^0_{n-1} set. We can also define another sequence of sets

$$\widehat{T_i} = \bigcap_{j\leq i} T_j,$$

which from Proposition 16(iii) are also Σ^0_{n-1}. Note that $\lim_i \widehat{T_i} = \overline{U}$ and so $\lim_i \mu(\widehat{T_i}) = 0$. We define a collection of sets $\{V_i : i \in \omega\}$ where for each i,

$$y \in V_i \iff (\exists p)\Big[y \in T_p \wedge (\forall j < p)\big[\mu(T_j) > 2^{-i}\big] \wedge \mu(T_p) \leq 2^{-i}\Big].$$

Clearly, $\{V_i : i \in \omega\}$ is an n-test, and $\overline{U} = \cap_i V_i$. Since $x \in \overline{U}$ this means that $x \in \cap_i V_i$, and so x is not n-random. $\square$

Furthermore, the weak n-random classes and the weak n-random classes are nested within each other.

Theorem 87 (Kurtz, [28]). *For any n, $n+1$-**W-RAND** $\subseteq$ n-**RAND**.*

Proof. Suppose that x is not n-random, so there is an n-test such that

$$x \in \cap_k U_k.$$

For each k, U_k is a Σ^0_n set, so from Proposition 16(v), $\cap_k U_k$ is a Π^0_{n+1} set. It is also (from Lemma 20) a measure zero set. Hence its complement is a Σ^0_{n+1} measure one set, i.e. a weak $n+1$-test. Note that x is in $\cap_k U_k$, so there is is a weak $n+1$-test that x is not in. Hence x is not weakly $n+1$-random. ☐

Thus we have that

$$n+1\textbf{-W-RAND} \subseteq n\textbf{-RAND} \subseteq n\textbf{-W-RAND}$$

for all n.

Each of these inclusions is proper. We prove this but first we prove a Lemma that these proofs require.

Lemma 88 (Kautz, [23]). *For any Σ^0_n set S and Π^0_n set T, for any rational $\epsilon > 0$ there is a Σ^0_n set S_2, a Π^0_n set T_1, a $\Pi^{\emptyset^{(n)}}_{n-1}$ set S_1 and a $\Sigma^{\emptyset^{(n)}}_{n-1}$ set T_2 such that*

- $S_1 \subseteq S \subseteq S_2$, $T_1 \subseteq T \subseteq T_2$.
- $\mu(S) - \mu(S_1) < \epsilon$, $\mu(T) - \mu(T_1) < \epsilon$
- $\mu(S_2) - \mu(S) < \epsilon$, $\mu(T_2) - \mu(T) < \epsilon$.

Proof. This proof uses induction on n.

Base: Assume S is a Σ^0_1 set. Setting $S_2 = S$ will do. To find a suitable S_1, enumerate S until a stage s when $\mu(S) - \mu(S_s) < \epsilon$. Finding this stage is a $\mathbf{0}'$ problem (as knowing the halting problem makes this decidable). So $S_1 = S_s$ is a set that is computable in $\emptyset'$. Thus given $\emptyset'$, S_1 is computable and so $S_1 \in \Pi^{\emptyset'}_0$. Define $T_1 = T$. To find T_2, find $\overline{T}_1$ (as above because $\overline{T}$ is a Σ^0_1 set) and let T_2 be the complement of this.

Induction step: Assume that the Lemma holds for $n = k$ and consider $n = k+1$. Consider our Σ^0_{k+1} set S. By Proposition 16(v), S can be represented by a countable sequence of Π^0_k sets T^i such that

$$S = \bigcup_{i \in \omega} T^i.$$

From the induction hypothesis we know that each T^i has a corresponding Π^0_k set T^i_2 such that $T^i \subseteq T^i_2$. Setting $S_2 = \bigcup_{i\in\omega} T^i_2$ ensures that $S \subseteq S_2$, and S_2 is Σ^0_{k+1}. Furthermore,

$$\begin{aligned} \mu(S_2) - \mu(S) &\leq \sum_{i\in\omega} \mu(T^i_2) - \mu(T^i) \\ &\leq \sum_{i\in\omega} \frac{\epsilon}{2^{i+1}} \\ &\leq \epsilon. \end{aligned}$$

To find S_1, find j such that

$$\mu(\bigcup_{i>j} T^i) \leq \frac{\epsilon}{2}.$$

A similar argument as used in Theorem 19(vi) ensures that this is computable in $\emptyset^{(k+1)}$. Now $\bigcup_{i\leq j} T^i$ is a Π^0_k set so by the induction hypothesis there is a set $S_1 \subseteq \bigcup_{i\leq j} T^i$ such that

$$\mu(\bigcup_{i\leq j} T^i) - \mu(S_1) \leq \frac{\epsilon}{2}.$$

S_1 is a $\Pi^{\emptyset^{(k+1)}}_k$ set and

$$\begin{aligned} \mu(S) - \mu(S_1) &= \mu(\bigcup_{i\in\omega} T^i) - \mu(S_1) \\ &\leq \mu(\bigcup_{i>j} T^i) + \mu(\bigcup_{i\leq j} T^i) - \mu(S_1) \\ &\leq \frac{\epsilon}{2} + \frac{\epsilon}{2} \\ &\leq \epsilon, \end{aligned}$$

as required.

For any Π^0_{k+1} set T, let S be the (Σ^0_{k+1}) complement of T, with S_1 and S_2 as defined above. Then let

$$T_1 = \overline{S_2}$$

and

$$T_2 = \overline{S_1}.$$

These will satisfy the requirements for T. □

We now show that the n-random and weak n-random classes are proper subsets of one another.

Theorem 89 (Kurtz,[28]). $n+1$-**W-RAND** $\subset$ n-**RAND**.

Enumerate the sequences of Σ^0_n sets as $\{\{S_{i,j} : i \in \omega\} : j \in \omega\}$. There is a corresponding list of requirements

$$R_j : \sum_i \mu(S_{i,j}) \leq 1 \Rightarrow x \text{ is in at most finitely many } S_{i,j}.$$

We wish to create a sequence of nested non-empty $\emptyset^{(n)}$-computable sets $\{C_s : s \in \omega\}$ such that $\lim_s \mu(C_s) = 0$. We will define a real x that is in the intersection of the C_s (a Π^0_{n+1} set with measure zero), and hence is not weakly $n+1$-random. We have to ensure that for each j, R_j is satisfied for x.

Construction. At stage -1, set $\sigma_{-1} = \lambda$ and $C_{-1} = [\lambda]$.

At stage $s \geq 0$, consider C_{s-1}. As this is a $\emptyset^{(n)}$-computable set, we can compute the elements given $\emptyset^{(n)}$. Because $C_{s-1} \subseteq [\sigma_{s-1}]$, and $\mu(C_{s-1}) > 0$, there will be an $x \in 2^{<\omega}$ such that $\sigma_{s-1}x \in C_{s-1}$. Find the first x (in length-lexicographic order, say) such that this is the case, and define $\sigma_s = \sigma_{s-1}(x[1])$. (That is, $\sigma_s = \sigma_{s-1}0$ or $\sigma_{s-1}1$). Define $C'_{s-1} = C_{s-1} \cap [\sigma_s]$. Ask

$$\text{“is } \sum_i \mu(S_{i,s}) \leq 1\text{?”}.$$

Theorem 19(iii) ensures that this is a $\emptyset^{(n)}$-computable predicate. If the answer is no, set $C_s = C'_{s-1}$. If the answer is yes, find $q \in \mathbb{Q}$ such that $\mu(C'_{s-1}) \geq q$ (this is computable in $\emptyset^{(n)}$) and $k(s)$ such that

$$k(s) = (\mu m)\Big[\sum_{i \geq m} \mu(S_{i,s}) < \frac{q}{2}\Big].$$

This is a $\emptyset^{(n)}$-computable function (from Theorem 19(v)) and so we can find a $\emptyset^{(n)}$-computable set $C_s \neq \emptyset$ such that

$$C_s \subseteq C'_{s-1} \cap \overline{\bigcup_{i \geq k(s)} S_{i,s}}.$$

$k(s)$ has been chosen to ensure that a non-empty C_s can be found.

Verification. Let $C = \cap_s C_s$. Let $x = \lim_s \sigma_s$.

- *x is well-defined.*

  Proof. For each $s \geq 0$, $\sigma_{s-1} \sqsubset \sigma_s$ and $|\sigma_s| = |\sigma_{s-1}| + 1$ so

  $$\lim_s |\sigma_s| \to \infty.$$

  □

- *For each s, C_s is a $\emptyset^{(n)}$-computable set.*

  Proof. Follows from the construction. □

- *C is a measure zero Π^0_{n+1} set.*

  Proof. Since each C_s is a $\emptyset^{(n)}$-computable set, and $C = \cap_s C_s$, it follows that C is a $\Pi_1^{\emptyset^{(n)}}$ set; that is, C must be a Π^0_{n+1} set. Futhermore, $\lim_s \mu([\sigma_s]) \to 0$. Since $\mu(C_s) \leq \mu([\sigma_s])$ for all s, it follows that $\lim_s \mu(C_s) \to 0$. Since C is the intersection of the sets C_s, $(\forall s) C \subseteq C_s$, and so C must be a measure zero set. □

- *$x \in C$.*

  Proof. Each C_s is a collection of extensions of strings, and so is a closed set.It is also bounded, as it is contained within $[\sigma_s]$. Furthermore, $C_{s+1} \subseteq C_s$. Thus there is a unique real in the intersection C. Clearly, this real is x, as $\cap_s C_s \subseteq \cap_s [\sigma_s]$ and x is the unique real in $\cap_s [\sigma_s]$. □

Thus x is not weakly $n+1$-random.

- *For all j, R_j is satisfied.*

Proof. If $\sum_i \mu(S_{i,j}) > 1$, the requirement is fulfilled trivially. Otherwise, we have $\sum_i \mu(S_{i,j}) \leq 1$. In that case, the construction picks C_j and m so that

$$C_j \subseteq \overline{\bigcup_{i \geq m} S_{i,j}}.$$

Now $x \in C_j$, so it follows that x cannot be in any $S_{i,j}$ where $i \geq m$. At most x can be in m of the $S_{i,j}$. $\square$

- *If $\{T_i : i \in \omega\}$ is a sequence of Σ^0_n sets and $\sum_i \mu(T_i) < \infty$, then x will be in at most finitely many T_i.*

  Proof. From Lemma 21, there is a stage r such that $\sum_{i \geq r} \mu(T_i) \leq 1$. All R_j are satisfied, so if we consider the set $\{T_i : i \geq r\}$, x can only be in finitely many members of it, say t. So x can be in at most $t + r$ members of $\{T_i : i \in \omega\}$. $\square$

Hence x is n-random.

Theorem 90 (Kautz, [23]). n-**RAND** $\subset$ n-**W-RAND**.

To prove this is true for a general n we need to build an Martin-Löf n-test $\{S_e : e \in \omega\}$ and a real x such that $x \in \cap_e S_e$ and x is in no Π^0_n class of measure zero. This latter condition ensures that x must be in every Σ^0_n set of measure one and hence that x is weakly n-random. This proof requires a finite injury priority argument.

We enumerate the Π^0_n sets, $\{P_e : e \in \omega\}$, as follows: we enumerate the sequences of Σ^0_{n-1} sets, $\{\{V_i, e : i \in \omega\} : e \in \omega\}$ and as in Proposition 16(v) we consider P_e to be the intersection of Σ^0_{n-1} sets, $P_e = \cap_i V_i, e$. Define $P_e[s] = \cap_{i<s} V_i$. As $P_e[s]$ is a finite intersection of Σ^0_{n-1} sets, Proposition 16(iii) implies that $P_e[s]$ is itself a Σ^0_{n-1} set. Note that $P_e[s+1] \subseteq P_e[s]$, so $\mu(P_e[s+1]) \leq \mu(P_e[s])$, and $\lim_s \mu(P_e[s]) = \mu(P_e)$.

We will build for each P_e a sequence of non-empty Π^0_{n-1} sets $T_e[s]$ such that if $T_e = \lim_s T_e[s]$, $T_{e+1} \subseteq T_e$ and define our requirements R_e by

$$R_e : \mu(P_e) = 0 \Rightarrow P_e \cap T_e = \emptyset.$$

Since $\lim_e T_e = 0$, we may define x to be the unique real in $\cap_e T_e$. The R_e have the following priority: $R_0, R_1, \ldots$.

At stage s, P_e has a corresponding function $t_e[s]$. This function represents a threshold that the measure of P_e may not go lower than. If it does, then there is a possibility that P_e may be a measure zero set. In that case it is necessary to ensure that our random real x will not enter P_e.

Definition 49. *Say that e requires attention at stage $s+1$ if*

- *$t_e[s]$ is undefined.*
- *$t_e[s]$ is defined and $\mu(P_e[s+1]) \leq t_e[s]$.*

Construction.

At stage 0:

- $S_i[0] = \emptyset$ for all $i \neq 0$. Set $S_0[0] = 2^\omega$.
- $T_i[0] = \emptyset$ for all $i \neq 0$. Set $T_0[0] = 2^\omega$.
- $t_i[0]$ is undefined for all $i \neq 0$. Set $t_0[0] = \frac{1}{2}$.

At stage $s > 0$:

Find the least e that requires attention. As $P_e[s]$ is a Π^0_{n-1} set, finding if $\mu(P_e[s]) \leq t_e[s]$ is a Σ^0_{n-1} predicate and hence is computable in $\emptyset^{(n-1)}$. If e requires attention via case 1, then $e > 0$ and $t_{e-1}[s-1]$ is defined (as otherwise e would not be the least requiring attention), so $\mu(T_{e-1}[s-1]) > 0$. Thus there is a rational $q > 0$ such that $\mu(T_e[s]) > q$.

Lemma 91. *For any r there is a string σ_r (where $|\sigma_r| = r$) such that $\mu(T_e[s] \cap [\sigma_r]) > \frac{q}{2^r}$.*

Proof. Proof by induction. If $r = 0$, then $\mu(T_e[s] \cap [\lambda]) = \mu(T_e[s]) > \frac{q}{2^0}$. Assume it is true for $r = k$, so $\mu(T_e[s] \cap [\sigma_k]) > \frac{q}{2^k}$. Consider $k+1$. Since it follows that at least half of the measure of $T_e[s] \cap [\sigma_k]$ must be in one of $\sigma_k 0$ and $\sigma_k 1$, the lemma holds. $\square$

Thus we can find a σ and $\epsilon > 0$ such that $\mu(T_{e-1}[s-1] \cap [\sigma]) > \epsilon$ and $|\sigma| \geq 2e$. Furthermore, since $T_{e-1}[s-1]$ is a Π^0_{n-1} set and so its elements can be computed given $\emptyset^{(n-1)}$, such a σ and ϵ can be found given $\emptyset^{(n-1)}$.

- Set $S_e[s] = S_e[s-1] \cup [\sigma]$. For all $i \neq e$, set $S_i[s] = S_i[s-1]$.
- Set $T_e[s] \subseteq T_{e-1}[s-1] \cap [\sigma]$ to be a Π^0_{n-1} set such that $\mu(T_e[s]) \geq \frac{\epsilon}{2}$. Lemma 88 ensures that this can be done. For all $i \neq e$, set $T_i[s] = T_i[s-1]$.
- Set $t_e[s] = \frac{\epsilon}{4}$. For all $i \neq e$, $t_i[s] = t_i[s-1]$.

If e requires attention via case 2 then because

$$\mu(P_e[s]) \leq t_e[s-1] \leq \frac{\mu(T_e[s-1])}{2},$$

it follows that

$$\mu(T_e[s-1] \cap \overline{P_e[s]}) \geq \frac{\mu(T_e[s-1])}{2}.$$

Because $P_e[s]$ is a Σ^0_{n-1} set (and thus $\overline{P_e[s]}$ is a Π^0_{n-1} set) and $T_e[s-1]$ is a Π^0_{n-1} set, Lemma 88 states that we can compute a Π^0_{n-1} set $T_e[s]$ such that $T_e[s] \subseteq T_e[s-1] \cap \overline{P_e[s]}$ and

$$\mu(T_e[s] \geq \frac{\mu(T_e[s-1])}{4}.$$

- For all e, set $S_e[s] = S_e[s-1]$.
- For all $i \neq e$, set $T_i[s] = T_i[s-1]$.
- Set $t_e[s] = -1$. For all $i < e$, $t_i[s] = t_i[s-1]$. For all $i > e$, $t_i[s]$ is undefined.

Verification.

- *If $t_e[s]$ is defined then $\mu(T_e[s]) > 0$ and $t_e[s] \leq \frac{\mu(T_e[s])}{2}$.*

Proof. Proof by induction. Assume that for all $s' < s$, this result holds. At stage s, either e requires attention via case 1, or via case 2, or e doesn't require attention at all. If e requires attention via case 1, then $\mu(T_e[s]) \geq \frac{\epsilon}{2}$ for some $\epsilon > 0$, and $t_e[s] = \frac{\epsilon}{4}$; that is, $t_e[s] \leq \frac{\mu(T_e[s])}{2}$. If e requires attention via case 2, then

$$\begin{aligned} \mu(T_e[s]) &\geq \frac{\mu(T_e[s-1])}{4} \\ &> 0, \end{aligned}$$

by the induction hypothesis. Also, $t_e[s] = -1$ so clearly $t_e[s] \leq \frac{\mu(T_e[s])}{2}$. If e doesn't require attention then $T_e[s] = T_e[s-1]$ and either $t_e[s] = t_e[s-1]$ in which case the induction hypothesis ensures that the result holds, or $t_e[s]$ is undefined in which case the lemma is fulfilled trivially. □

- *Every e requires attention at most finitely often. In fact, e requires attention via case 1 at most 2^e times and via case 2 at most 2^e times.*

  The argument here is the same one used in Theorem 83. This result implies that $t_e = \lim_s t_e[s]$, $T_e = \lim_s T_e[s]$, and $S_e = \lim_s S_e[s]$ all exist. Furthermore, it is clear that each $e > 0$ requires attention via case 1 at least once.

- *For all e, S_e is a Σ_n^0 set, and $\mu(S_e) \leq 2^{-e}$.*

  Proof. The string σ that enters S_e at stage s is computable from $\emptyset^{(n-1)}$. Thus $S_e[s]$ can be computed from $\emptyset^{(n-1)}$, and $S_e = \bigcup_s S_e[s]$ i.e.

  $$x \in S_e \iff (\exists s)(x \in S_e[s]).$$

  $x \in S_e[s]$ is a $\emptyset^{(n-1)}$-computable relation, and so S_e is a $\Sigma_1^{\emptyset^{(n-1)}}$ set. Hence from Lemma 17, S_e is c.e. in $\emptyset^{(n-1)}$ and from Theorem 18(iii), this means that S_e is Σ^0_n. Furthermore,

$$\begin{aligned}\mu(S_e) &= \sum_{\sigma\in S_e} 2^{-|\sigma|} \\ &\leq \sum_{\sigma\in S_e} 2^{-2e} \\ &\leq 2^{-2e}2^{e} \\ &\leq 2^{-e},\end{aligned}$$

as at most 2^e σ can enter S_e (since e requires attention via case 1 at most 2^e times). □

It is necessary to show that a real has been constructed that is weakly n-random but not n-random. Fortunately, the sets $\{T_e : e \in \omega\}$ define a unique real that satisfies our requirements.

- *For all e, $T_{e+1} \subseteq T_e$. Also, each T_e is closed, bounded and non-empty and $\lim_e T_e = 0$. Hence it can be concluded that there is a unique real $x \in \cap_e T_e$.*

  Proof. From the construction. Note that for all e,

  $$\begin{aligned}\mu(T_e) &\leq 2^{-|\sigma|} \\ &\leq 2^{-2e},\end{aligned}$$

  so $\lim_e T_e = 0$. □

- *If P_e is a measure zero Π^0_n set, then $x \notin P_e$.*

  Proof. If $t_e > 0$, then $\mu(P_e) > 0$, and R_e is satisfied trivially. However, if $t_e = -1$, then it is possible that P_e is a measure zero set. In this case, consider the final stage s for which e requires attention. By construction, for all $s' > s$, $t_e[s'] = t_e[s] = -1$. $T_e[s]$ was defined so that $T_e[s] \subseteq \overline{P_e}$, i.e. $T_e[s] \cap P_e = \emptyset$. Now for all $s' > s$, $T_e[s'] = T_e[s]$ and furthermore for all $d > e$, $T_d \subseteq T_e$. Since $x \in \cap_e T_e$, this means that $x \notin P_e$. □

- $x \in \cap_e S_e$.

  Proof. The construction ensures that there is a string $\sigma \in S_e$ such that $x \in [\sigma]$. □

The above observations prove that x is weakly n-random, but not n-random, as required.

From these Theorems we see that the n-random and weakly n-random classes form an infinite hierarchy of randomness classes nested inside one another.

Note that it is possible to enumerate the Σ_n^0 sets, and so one can define a Universal n-test. Thus for all n, the proof used to show that the Martin-Löf random reals have measure one can be used to show that the class n-**RAND** of n-random reals has measure one. Furthermore, the nesting properties of the n-random and weakly n-random classes ensures that for all n, n-**W-RAND** has measure one.

15.2 Arithmetic randomness

Yet another type of randomness can be defined:

Definition 50. *A real x is* arithmetically random *if it is n-random for all n.*

Obviously a similar definition could be postulated for weakly arithmetically random reals. However, because of the nesting properties of n-random and weakly n-random classes, the two classes are equivalent.

Theorem 92. *Let* **A-RAND** *denote the class of arithmetically random reals.* **A-RAND** *has measure one.*

Proof. Define **A-NULL** to be the set of all reals that are not arithmetically random.

$$\textbf{A-RAND} = \bigcap_n n\text{-}\textbf{RAND},$$

and so

$$\mathbf{A\text{-}NULL} = \bigcup_n n\text{-}\mathbf{NULL}.$$

Hence the countable additivity property of Lebesgue measure ensures that

$$\mu(\mathbf{A\text{-}NULL}) \leq \sum_{n\in\omega} \mu(n\text{-}\mathbf{NULL}).$$

As discussed above, $\mu(n\text{-}\mathbf{NULL}) = 0$ for all n. This means that for all $\epsilon > 0$, $\mu(n\text{-}\mathbf{NULL}) < \epsilon 2^{-(n+1)}$. Thus for all $\epsilon > 0$,

$$\begin{aligned}\mu(\mathbf{A\text{-}NULL}) &\leq \sum_{n\in\omega} \mu(n\text{-}\mathbf{NULL}) \\ &< \sum_{n\in\omega} \epsilon 2^{-(n+1)} \\ &< \epsilon.\end{aligned}$$

Thus $\mu(\mathbf{A\text{-}NULL}) = 0$ as required. $\square$

Note that

$$\mathbf{A\text{-}RAND} \subset n\text{-}\mathbf{RAND}$$

for all n, as the theorems above ensure that

$$n + 1\text{-}\mathbf{RAND} \subset n\text{-}\mathbf{RAND}.$$

15.3 The difference hierarchy

The arithmetic hierarchy is not the only hierarchy to have the c.e. sets as the lowest rung on the ladder. The difference hierarchy (also known as Ershov's hierarchy) also extends the definition of computably enumerable sets, but in a different way. The classes of the difference hierarchy are defined as follows:

Definition 51. *A set $A \subseteq 2^{<\omega}$ is n-c.e. if there is a recursive function*

$$\mathcal{A} : 2^{<\omega} \times \omega \to \{0, 1\}$$

such that

$$A(x) = \lim_s \mathcal{A}(x, s)$$

and

$$|\{\mathcal{A}(x, s+1) \neq \mathcal{A}(x, s)\}| \leq n.$$

This means is that although a given string may enter A, and then perhaps leave it, and then perhaps enter it again, the number of times it may do so is bounded by n; that is, it can enter and leave A at most n times. For a 1-c.e. set A, once a string x enters A it may never leave it again, so A is a c.e. set. Similarly, strings may go into a 2-c.e. set A and then may come out again, but then they may never re-enter A again. It is obvious that any n-c.e. set must be an m-c.e. set for any $m \geq n$.

We can define an n-c.e. test set in the same way as a Martin-Löf test or an n-test:

Definition 52. *A D_n test is a computable collection of n-c.e. sets $\{U_k : k \in \omega\}$, such that*

$$(\forall k)\mu(U_k) \leq 2^{-k}.$$

A real x is D_n random if and only if for any D_n test $\{U_k : k \in \omega\}$,

$$x \notin \cap_k U_k.$$

However, unlike the arithmetic hierarchy, the difference hierarchy doesn't generate any new classes of random reals.

Theorem 93. *The following are equivalent:*

1. *x is 2-random.*

2. *x is D_2-random.*

3. *x is D_n-random, for $n \geq 2$.*

Proof. $(\neg 2 \Rightarrow \neg 3)$ Suppose that x is not D_2 random; that is, there is a D_2 test

$$\{U_k : k \in \omega\}$$

such that $x \in \cap_k U_k$. For any $n \geq 2$, $\{U_k : k \in \omega\}$ is also a D_n test. Hence x is not D_n-random.

$(\neg 3 \Rightarrow \neg 1)$ Suppose that x is not D_n random; that is, there is an D_n test

$$\{U_k : k \in \omega\}$$

such that $x \in \cap_k U_k$. For each k, and for any string $\sigma \in 2^{<\omega}$, note that

$$\sigma \in U_k \iff (\exists s)(\forall t > s)(\sigma \in U_{k,t}).$$

Thus U_k is a Σ^0_2 test, and so x is not 2-random.

$(\neg 1 \Rightarrow \neg 2)$ Suppose that x is not 2-random. Then there is a 2-test such that $(\forall k) x \in U_k$. Because all the U_k are Σ^0_2 inclusion in U_k is determined by

$$\sigma \in U_k \iff (\exists s)(\forall t > s)(\sigma \in U_{k,s}).$$

We build a D_2 test $\{V_k : k \in \omega\}$ as follows: Suppose that at stage t, $\sigma \downarrow U_{k,t}$. Assume without loss of generality that $|\sigma| \leq t$. (If this is not the case, the computation may be slowed down until it is true). Define

$$S_{\sigma,t} = \{\gamma : \sigma \sqsubseteq \gamma, |\gamma| = t\},$$

and let

$$V_{k,t} = V_{k,t-1} \cup S_{\sigma,t}.$$

If at a later stage t_1, $\sigma \uparrow U_{k,t_1}$, then define

$$V_{k,t_1} = V_{k,t_1-1} - S_{\sigma,t}.$$

If at stage s a string $\gamma \uparrow V_{k,s}$, then $|\gamma| < s$. By construction nothing of length less than s enters V_k after s. This means that γ can never enter V_k again, and hence that V_k is D_2. Furthermore, for all k

$$\begin{aligned} \mu(V_k) &= \mu(U_k) \\ &\leq 2^{-k}, \end{aligned}$$

and so $\{V_k : k \in \omega\}$ is a D_2 test. Since

$$(\forall k)(\exists n)(x \restriction n \in U_k),$$

it follows that

$$(\forall k)(\exists m)(x \restriction m \in V_k),$$

and so x is not D_2 random, as required. $\square$

Chapter 16

s-randomness

16.1 Hausdorff dimension

The study of measure as a way of specifying the size of sets began with work by Borel in 1895. In 1904 the Lebesgue measure was introduced ([29]). As discussed in Chapter 5, Lebesgue measure defines the size of a set via the size of the intervals covering it, and is a very important type of measure. In 1914 however, a more general construction of measure was produced by Carathéodory [6] that included Lebesgue measure as a special case. In any n-dimensional Euclidean space, Carathéodory's generalized version defined the s-dimensional measure of a set A as

$$\mu^s(A) = \inf\{\sum_i |I_i|^s : A \subset \bigcup_i I_i\}$$

for any s, where I_i is an interval in the space. Clearly, Lebesgue measure is the $s = 1$ case of this. A 1918 paper by Felix Hausdorff, [22], extended the definition of s-dimensional measure by allowing s to range over *non-integer* values.

Definition 53. *Let M be a metric space and $F \subseteq M$. Let $\mathfrak{C}$ be a class of open sets. $\mathfrak{C}$ is an* open cover *of F if*

$$(\forall x \in F)(\exists C \in \mathfrak{C} x \in C.$$

$\mathfrak{C}$ *is a* δ-cover *of* F *if for all* $C \in \mathfrak{C}$, $\mu(C) \leq \delta$. *For any measurable set* F, *the Hausdorff t-measure relative to* δ *is:*

$$\mathcal{H}^t_\delta(F) \equiv \inf_{\substack{\delta\text{-covers} \\ \text{of } F}} \sum_{C \in \mathfrak{C}} (\mu(C))^t.$$

The (absolute) Hausdorff t-measure is defined via:

$$\mathcal{H}^t(F) \equiv \lim_{\delta \to 0} \mathcal{H}^t_\delta(F).$$

Note that as δ decreases, there are fewer δ-covers to use and so the infimum of

$$\inf_{\substack{\delta\text{-covers} \\ \text{of} F}} \sum_{C \in \mathfrak{C}} (\mu(C))^t$$

cannot decrease; that is

$$\epsilon \geq \delta \Rightarrow \mathcal{H}^t_\epsilon(F) \geq \mathcal{H}^t_\delta.$$

Theorem 94. *Let* $0 < s < t$. *Then*

- *If* $\mathcal{H}^s(F) < \infty$ *then* $\mathcal{H}^t(F) = 0$.
- *If* $\mathcal{H}^t(F) > 0$ *then* $\mathcal{H}^s(F) = \infty$.

Proof. • If $\mathcal{H}^s(F) < \infty$, then for any t such that $s < t$ (so $t = s + c$ for some $c > 0$)

$$\begin{aligned}
H^t(F) &= \lim_{\delta \to 0} \mathcal{H}^t_\delta(F) \\
&= \lim_{\delta \to 0} \mathcal{H}^{s+c}_\delta(F) \\
&= \lim_{\delta \to 0} \inf_{\substack{\delta\text{-covers} \\ \text{of } F}} \sum_{C \in \mathfrak{C}} (\mu(C))^{s+c} \\
&\leq \lim_{\delta \to 0} \inf_{\substack{\delta\text{-covers} \\ \text{of } F}} \sum_{C \in \mathfrak{C}} (\mu(C))^{s} \delta^c \\
&\leq \lim_{\delta \to 0} \delta^c \mathcal{H}^s_\delta(F) \\
&= 0,
\end{aligned}$$

as $c > 0$ and $\mathcal{H}^s(F) < \infty$.

- If $\mathcal{H}^t(F) > 0$, then for any $s < t$, (i.e. $s = t - c$ where $c > 0$)

$$\begin{aligned}
\mathcal{H}^s(F) &= \lim_{\delta\to 0} \mathcal{H}^s_\delta(F) \\
&= \mathcal{H}^{t-c}_\delta(F) \\
&= \lim_{\delta\to 0} \inf_{\substack{\delta\text{-covers}\\ \text{of } F}} \sum_{C\in\mathfrak{C}} (\mu(C))^{t-c} \\
&\geq \lim_{\delta\to 0} \inf_{\substack{\delta\text{-covers}\\ \text{of } F}} \sum_{C\in\mathfrak{C}} (\mu(C))^{t}\delta^{-c} \\
&\geq \lim_{\delta\to 0} \delta^{-c}\mathcal{H}^t_\delta(F) \\
&= \infty,
\end{aligned}$$

as $c > 0$ and $\mathcal{H}^t(F) > 0$.

□

This theorem implies that if $\mathcal{H}^s$ drops below ∞, then for any $t > s$, $\mathcal{H}^t$ must be zero. Furthermore, if $\mathcal{H}^s(F)$ is non-zero, then $\mathcal{H}^t(F)$ is infinite for $t < s$. Thus there can be at most one s such that $\mathcal{H}^s(F)$ is neither zero nor infinite. If it exists, this s is known as the *Hausdorff dimension* of F. More generally, the Hausdorff dimension is thought of as the "cut-off point" between infinitely large and zero Hausdorff s-measure.

Definition 54. *The Hausdorff dimension of a set F is*

$$\dim_H(F) = \inf\{t \in \mathbb{R}^+ : \mathcal{H}^t(F) = 0\},$$

or equivalently

$$\dim_H(F) = \sup\{t \in \mathbb{R}^+ : \mathcal{H}^t(F) = \infty\}.$$

It follows from this that if $\mathcal{H}^t(F) = 0$ for all t, then $\dim_H(F) = 0$, and if $\mathcal{H}^t(F) = \infty$ for all t, then $\dim_H(F) = \infty$.

In this work we are primarily interested in the real numbers. The following theorem is useful for determining the Hausdorff dimension of many sets in $\mathbb{R}$. In particular, we can use it to prove that $\dim_H(\mathbb{R}) = 1$, so the Hausdorff dimension agrees with the usual topological dimension of the real line.

Theorem 95. *In $\mathbb{R}$, for any measurable set F, $\mathcal{H}^1(F) = \mu(F)$.*

Proof. Let $\{I_i : i \in \omega\}$ be any set of intervals covering F. Then

$$\mu(F) = \inf_{\substack{\text{covers}\\ \text{of } F}} \sum_{i \in \omega} |I_i|.$$

For any interval cover of F, each interval I_i is of the form $[a_i, b_i)$. For any $\delta > 0$ there are points $a_i = x_0 < x_1 < \ldots < x_n = b_i$ such that $|x_j - x_{j-1}| < \delta$ for all j, and $|I_i| = \sum_{j \leq n}(x_j - x_{j-1}) = b_i - a_i$. The subintervals $[x_{j-1}, x_j)$ form a δ-cover of F, and

$$\sum_{i \in \omega} \sum_{j \leq n} [x_{j-1}, x_j) = \sum_{i \in \omega} |I_i|.$$

Thus the value of $\mu(F)$ is going to remain unchanged even if we restrict our covers to δ-covers, as every set that is not a δ-cover has a corresponding δ-cover, for which the sum of the interval diameters is identical. Since $\mu(F)$ is the infimum over all possible covers of F, there cannot be any δ-covers that give a lower value than $\mu(F)$, for any $\delta > 0$. Thus for any $\delta > 0$,

$$\begin{aligned} \mu(F) &= \inf_{\substack{\delta\text{-covers}\\ \text{of } F}} \sum_{i \in \omega} |I_i| \\ &= \mathcal{H}^1_\delta(F). \end{aligned}$$

This is true for any $\delta > 0$, thus

$$\begin{aligned} \mathcal{H}^1(F) &= \lim_{\delta \to 0} \mathcal{H}^1_\delta \\ &= \lim_{\delta \to 0} \mu(F) \\ &= \mu(F). \end{aligned}$$

□

Corollary 96. $\dim_H(\mathbb{R}) = 1$.

Proof. Theorem 95 ensures that

$$\mathcal{H}^1(\mathbb{R}) = \mu(\mathbb{R}) = \infty.$$

Thus $\dim_H(\mathbb{R}) \geq 1$. Furthermore, let $s > 1$ be arbitrary, and note that $\mathbb{R} = \cup_{n\in\mathbb{Z}}[n, n+1]$. For all $n \in \mathbb{Z}$,

$$\mathcal{H}^1([n, n+1]) = \mu([n, n+1]) = 1.$$

Thus $\dim_H([n, n+1]) = 1$, by the definition of $\dim_H$, $\mathcal{H}^s([n, n+1]) = 0$. By the definition of Hausdorff measure, we note that

$$\begin{aligned}
\mathcal{H}^s(\mathbb{R}) &= \mathcal{H}^s(\cup_{n\in\mathbb{Z}}[n, n+1]) \\
&= \sum_{n\in\mathbb{Z}} \mathcal{H}^s([n, n+1]) \\
&= \sum_{n\in\mathbb{Z}} 0 \\
&= 0,
\end{aligned}$$

so $\dim_H(\mathbb{R}) \leq 1$, as required. □

In this work the interval $[0, 1)$ is of particular interest to us. In this case, we assume our open covers consist of sets of extensions of strings. Then a k-cover $\mathfrak{C}$ contains only strings of length k or longer, so in this space

$$\mathcal{H}^t_k(F) = \inf_{\substack{k\text{-covers} \\ \text{of } F}} \sum_{\sigma\in\mathfrak{C}} 2^{-t|\sigma|}$$

and

$$\mathcal{H}^t(F) = \lim_{k\to\infty} \mathcal{H}^t_k(F).$$

Corollary 97. *Let $A \subseteq [0, 1)$ be measurable, and $\mu(A) \neq 0$. Then $\dim_H(A) = 1$.*

Proof. Since $A \subseteq [0, 1)$ we know that $\mu(A)$ is not infinite. In fact,

$$\mu(A) \leq 1.$$

As well as this, if $\mu(A) \neq 0$, it follows that $\mathcal{H}^1(A) \notin \{0, \infty\}$. There is not more that one s such that $\mathcal{H}^s(A)$ is neither zero nor infinite and if such an s exists it must be the Hausdorff dimension of A. Thus $\dim_H(A) = 1$. This is true for any non-measure-zero sets $A \subseteq [0, 1)$, including $[0, 1)$ itself. □

Hence the only subsets of $[0,1)$ that don't have Hausdorff dimension 1 are the measure zero sets. If $\mu(A)=0$, the Hausdorff dimension of A may contain some information about A.

Proposition 98. *For any sets A and B, if $A \subseteq B$ then $\dim(A) \leq \dim(B)$.*

Proof. Let $A \subseteq B$. Take any $s > \dim_H(B)$, so $\mathcal{H}^s(B) = 0$. Now if $\mathfrak{C}$ is a δ-cover of B, then it is clearly also a δ-cover of A. Thus for all δ,

$$\mathcal{H}^s_\delta(A) \leq \mathcal{H}^s_\delta(B),$$

and so

$$\mathcal{H}^s(A) \leq \mathcal{H}^s(B),$$

so $\mathcal{H}^s(A) = 0$. Since this is true for all $s > \dim_H(B)$,

$$(\forall s > \dim_H(B))\mathcal{H}^s(A) = 0.$$

Thus

$$\begin{aligned} \dim_H(A) &= \inf\{s : \mathcal{H}^s(A) = 0\} \\ &\leq \dim_H(B). \end{aligned}$$

□

If $A \subseteq [0,1)$, it follows that

$$\begin{aligned} \dim_H(A) &\leq \dim_H([0,1)) \\ &\leq 1. \end{aligned}$$

If $\mu(A) \neq 0$ then the Hausdorff dimension of A is as large as it can possibly be. On the other hand, measure zero sets with different dimensions exist. For example, countable sets have dimension zero. Countable sets are thus extremely small, even relative to other measure zero sets. The Cantor ternary set (the canonical example of an uncountable measure zero set) has dimension $\frac{\log 2}{\log 3}$, and other (measure zero) sets of varying dimension can be found.

16.2 s-randomness

Since Hausdorff dimension differentiates between the sizes of measure zero sets, and non-random numbers are classified as those that are in a measure zero set (of a certain effectiveness), Hausdorff dimension can be used to qualify the level of non-randomness of a certain set. For example, measure zero sets with Hausdorff dimension 0 represent reals that are exceedingly non-random. Comparatively speaking, those measure zero sets with larger Hausdorff dimension are "more" random-like that those where $\dim_H = 0$.

To put these ideas in terms of the standard definitions for randomness, note that measure zero sets can be represented by martingales, that is, given any set A, $\mu(A) = 0$ if and only if $A = S[d]$, where

$$S[d] = \{x : \limsup_n d(x \upharpoonright n) = \infty\}.$$

Theorem 51 is just an effectivization of this fact. Hausdorff dimension can be characterized in a similar way if we broaden the notion of martingales in the following manner:

Definition 55. *An s-gale is a function $d : 2^{<\omega} \to \mathbb{R}^+$such that for all strings σ,*

$$d(\sigma 0) + d(\sigma 1) = 2^s d(\sigma).$$

Let $\mathcal{D}_s$ represent the class of all s-gales. The following theorem shows how the behaviour of an s-gale on a set is precisely determined by the Hausdorff dimension of that set

Theorem 99 (Lutz, [34]). *Let $X \subseteq 2^\omega$, and set*

$$\mathcal{S}(X) = \{s \in \mathbb{R}^+ : (\exists d \in \mathcal{D}_s) X \subseteq S[d]\}.$$

Then

$$\dim_H(X) = \inf \mathcal{S}(X).$$

Since $\dim_H(X) = \inf\{s : \mathcal{H}^s(X) = 0\}$ this means that $\mathcal{H}^s(X) = 0$ if and only if $s \in \mathcal{S}(X)$.

The proof of this theorem requires the following Lemma. This lemma is a modification of Lemma 117 and can be proved in the same way.

Lemma 100. *Let d be an s-gale, and $X \subseteq 2^\omega$ a prefix-free set. Then*

$$\sum_{x \in X} 2^{-s|x|} d(x) \leq d(\lambda).$$

We now turn to the proof of Theorem 99.

Proof. This result is trivial if $X = \emptyset$, so we will assume that X is non-empty. Assume that $\mathcal{H}^s(X) = 0$; that is,

$$\lim_{k \to \infty} \mathcal{H}^s_k(X) = 0.$$

$\mathcal{H}^s_k(X)$ can only increase as k increases, and for all k, $\mathcal{H}^s_k(X) \geq 0$. This implies that for any k, $\mathcal{H}^s_k(X) = 0$ and in particular $\mathcal{H}^s_0(X) = 0$. For any $r \in \omega$ there is a prefix-free set of strings A_r (with no restriction on the length of strings in A_r) such that A_r covers X and

$$\sum_{\sigma \in A_r} 2^{-s|\sigma|} \leq 2^{-r}.$$

We define:

$$d_r(\tau) = \begin{cases} 2^{(s-1)(|\tau|-|\sigma|)} & \text{if } \sigma \sqsubseteq \tau \text{ and } \sigma \in A_r \\ \sum_{\tau\sigma \in A_r} 2^{-s|\sigma|} & \text{otherwise.} \end{cases}$$

It is clear that for any r, $d_r(\tau)$ is an s-gale and

$$\begin{aligned} d_r(\lambda) &= \sum_{\sigma \in A_r} 2^{-s|\sigma|} \\ &\leq 2^{-r}, \end{aligned}$$

by assumption on A_r. However $d_r(\lambda)$ is non-zero as $X \neq \emptyset$. Finally, for any $\sigma \in A_r$, $d_r(\sigma) = 1$. Set

$$d(\tau) = \sum_{r=0}^{\infty} 2^r d_{2r}(\tau)$$

This is clearly an s-gale, and $d(\lambda)$ is bounded (by 2) but is non-zero.

Take any $x \in X$. For any $q \in \omega$, there is a $\sigma \in A_{2q}$ such that $\sigma \sqsubseteq x$, because A_{2q} covers X. Then

$$\begin{aligned} d(\sigma) &= \sum_{r=0}^{\infty} 2^r d_{2r}(\sigma) \\ &\geq 2^q d_{2q}(\sigma) \\ &\geq 2^q. \end{aligned}$$

As this is true for arbitrary q, the s-gale d succeeds on x; thus $s \in \mathcal{S}(X)$.

Conversely, take any $s \in \mathcal{S}(X)$, so there is an s-gale d that succeeds on all $x \in X$. Note that because we are assuming that $X \neq \emptyset$, $d(\lambda) > 0$! Define for any $k, r \in \omega$

$$a_k = 1 + \max\{d(\sigma) : |\sigma| \leq k\}$$

and

$$A_{k,r} = \{\sigma : d(\sigma) \geq 2^r a_k \text{ and } \tau \sqsubset \sigma \Rightarrow d(\tau) < 2^r a_k\}.$$

By construction $A_{k,r}$ is prefix-free and contains no strings shorter than length k. Since d succeeds on all $x \in X$,

$$(\forall k, r)(\exists j) d(x \upharpoonright j) \geq 2^r a_k,$$

and so $A_{k,r}$ is a cover of X. In fact, because it contains no strings with length shorter than k, it is a k-cover of X. Furthermore, $A_{k,r}$ is a prefix-free set, so Lemma 100 ensures that:

$$\begin{aligned} \sum_{\sigma \in A_{k,r}} 2^{-s|\sigma|} d(\sigma) &\leq d(\lambda) \\ \sum_{\sigma \in A_{k,r}} 2^{-s|\sigma|} 2^r a_k &\leq d(\lambda) \\ \sum_{\sigma \in A_{k,r}} 2^{-s|\sigma|} &\leq 2^{-r} \frac{d(\lambda)}{a_k} \\ &\leq 2^{-r}, \end{aligned}$$

as it can be seen that $a_k > d(\lambda)$. From this we can conclude that for arbitrary r,

$$\mathcal{H}^s_k(X) \leq 2^{-r},$$

and so

$$\mathcal{H}^s(X) = 0,$$

as required. □

The original definition of a random number in terms of martingales can be augmented in the obvious way:

Definition 56. *Let $\mathcal{D}_{s,c}$ be the class of all weakly computable s-gales. A real x is s-random if and only if $(\forall t > s)(\forall d \in \mathcal{D}_{t,c})(x \notin S[d])$.*

Clearly, Martin-Löf randomness is simply s-randomness where $s = 1$.

The theory of s-gales can be expressed in a different, but equivalent way, by using martingales but requiring that they succeed quickly (i.e. requiring that they are basically s-gales).

Recall that a martingale d effectively succeeds on a real x if there is a computable, unbounded, non-decreasing function h such that

$$\limsup_{n\to\infty} d(x \upharpoonright n) - h(n) \geq 0.$$

Schnorr introduced the concept of the *order* of a martingale. This is a more specific version of effective success, because it takes note of *what* computable martingale is the lower bound for d on x. Obviously some martingales will succeed faster than others on a real x, and measuring the order of a martingale on a real reflects that.

We define the concept of a success set as follows:

Definition 57 (Schnorr, [40]). *Let d be a martingale. We define the success set of d relative to a computable, unbounded, non-decreasing function $f(n)$ by*

$$S_{f(n)}[d] = \{x : \limsup_{n\to\infty} d(x \upharpoonright n) - f(n) \geq 0\}.$$

Relative success sets provides an equivalent characterization of s-randomness. Let D_c be the class of all weakly computable martingales, and define

$$\mathcal{S}_c(X) = \{s \in \mathbb{Q}^+ : (\exists d \in \mathcal{D}_{s,c})(\forall x \in X) X \subseteq S[d]\}.$$

Theorem 101. *x is s-random $\iff (\forall t > s)(\exists d \in D_c)(x \notin S_{2^{(1-t)n}}[d])\}$.*

Proof. Define $f(\sigma) = 2^{(1-s)|\sigma|}d(\sigma)$. If d is an s-gale, f is a martingale because

$$\begin{aligned} f(\sigma 0) + f(\sigma 1) &= 2^{(1-s)|\sigma 0|}d(\sigma 0) + 2^{(1-s)|\sigma 1|}d(\sigma 1) \\ &= 2^{(1-s)(|\sigma|+1)}\big[d(\sigma 0) + d(\sigma 1)\big] \\ &= 2^{(1-s)|\sigma|}2^{1-s}\big[2^s d(\sigma)\big] \\ &= 2f(\sigma), \end{aligned}$$

and if f is a martingale then d is an s-gale because

$$\begin{aligned} d(\sigma 0) + d(\sigma 1) &= \frac{f(\sigma 0) + f(\sigma 1)}{2^{(1-s)(|\sigma|+1)}} \\ &= \frac{2f(\sigma)}{2^{(1-s)|\sigma|}2^{(1-s)}} \\ &= \frac{2d(\sigma)}{2^{(1-s)}} \\ &= 2^s d(\sigma). \end{aligned}$$

Suppose that d is an s-gale that succeeds on x; that is,

$$(\forall c \in \omega)(\exists n) d(x \upharpoonright n) \geq c.$$

Because there are infinitely many $c \in \omega$ it follows that

$$(\exists^\infty n) d(x \upharpoonright n) \geq 1.$$

We define a martingale $f(\sigma) = 2^{(1-s)|\sigma|}d(\sigma)$ as discussed above. Thus

$$\begin{aligned} (\exists^\infty n) d(x \upharpoonright n) 2^{(1-s)n} &\geq 2^{(1-s)n} \\ (\exists^\infty n) f(x \upharpoonright n) &\geq 2^{(1-s)n}, \end{aligned}$$

so $x \in S_{2^{(1-s)n}}[f]$. This implies that for all $t > s$, $x \in S_{2^{(1-t)n}}$.

Conversely, suppose that $f(x \upharpoonright n) \geq 2^{(1-s)n}$ for infinitely many n. For any $t > s$, define a t-gale as above via

$$d(\sigma) = \frac{f(\sigma)}{2^{(1-t)|\sigma|}}.$$

Then it follows that

$$\begin{aligned}(\forall m)(\exists n \geq m)d(x \upharpoonright n) &\geq \frac{2^{(1-s)n}}{2^{(1-t)n}} \\ &\geq 2^{(t-s)n}.\end{aligned}$$

Because $t - s > 0$ it follows that for any

$$\begin{aligned}(\forall m)\left(\exists j \geq \frac{m}{(t-s)}\right)d(x \upharpoonright j) &\geq 2^{(t-s)j} \\ &\geq 2^{\frac{(t-s)j}{(t-s)}} \\ &\geq 2^{m},\end{aligned}$$

and so for any $t > s$ there is a t-gale that succeeds on x. □

We observe that if there is some weakly computable d such that

$$x \in S_{2^{(1-s)n}}[d],$$

then for the same d, and for any $t > s$,

$$x \in S_{2^{(1-t)n}}[d],$$

as this is a weaker statement. Thus

$$s\textbf{-NULL} \subseteq t\textbf{-NULL},$$

and so

$$t\textbf{-RAND} \subseteq s\textbf{-RAND}.$$

It seems plausible that

$$t\textbf{-RAND} \neq s\textbf{-RAND},$$

and we will show this to be true, as there is an s-random real that is not t-random, for any s and t such that $t > s$. To do this, we define the *constructive dimension* of a set X:

Definition 58 (Lutz, [32]). *Let $X \subseteq 2^\omega$. Define*

$$\mathcal{S}_c(X) = \{s \in \mathbb{R}^+ : (\exists d \in \mathcal{D}_{s,c}) X \subseteq S[d]\}.$$

The constructive dimension of X is defined via

$$\dim_c(X) = \inf \mathcal{S}_c(X).$$

From this, we can define the (effective) dimension of a single (infinite) string x, written $\dim(x)$, as the constructive dimension of the singleton set containing x.

Proposition 102. $\dim(\Omega) = 1$.

Proof. Since Ω is random, there is no $d \in \mathcal{D}_c$ such that $\{\Omega\} \subseteq S[d]$. Thus $\dim_c(\{\Omega\}) \geq 1$. Conversely,

$$\begin{aligned} \dim_c(\{\Omega\}) &\leq \dim_H(\{\Omega\}) \\ &\leq 1. \end{aligned}$$

Thus $\dim(\Omega) = \dim_c(\{\Omega\}) = 1$, as required. $\square$

From Ω we can define an s-random real for any s. For example, if we define

$$\Omega^{\frac{1}{2}} = \Omega[1]0\Omega[2]0\Omega[3]0\ldots,$$

then $\Omega^{\frac{1}{2}}$ is half as compressible as Ω, and so in terms of our understanding of randomness as incompressibility, $\Omega^{\frac{1}{2}}$ is "half" as random. It turns out this intuition is correct, and can be formalized in terms of s-randomness. Formally, we define a "dilution" function $g_s : 2^\omega \to 2^\omega$ for any rational $s = \frac{s_1}{s_2}$ via

$$g_s(x) = x[1]\ldots x[s_1]0^{s_2-s_1}x[s_1+1]\ldots x[2s_1]0^{s_2-s_1}\ldots$$

We also define a function $g : 2^{<\omega} \to 2^{<\omega}$ such that $\lim_{n\to\infty} g(x \upharpoonright n) = g_s(x)$. If $|\sigma| = ks_1$ for some $k = 1, 2\ldots$, then

$$g(\sigma) = \sigma[1]\ldots\sigma[s_1]0^{s_2-s_1}\sigma[s_1+1]\ldots\sigma[2s_1]0^{s_2-s_1}\ldots\sigma[n]0^{s_2-s_1}.$$

If $|\sigma| < ks_1$, then

$$g(\sigma) = \sigma[1] \ldots \sigma[s_1]0^{s_2-s_1}\sigma[s_1+1] \ldots \sigma[2s_1]0^{s_2-s_1} \ldots \sigma[n].$$

Lemma 103. *For all $\sigma \in 2^{<\omega}$,*

$$|g(\sigma)| \leq \frac{|\sigma|}{s}.$$

Proof. This is clear from the construction. $\square$

Theorem 104 (Lutz, [32]). *Let $s \in \mathbb{Q}^+$. Then*

$$\dim(g_s(\Omega)) = s.$$

Proof. We consider $t > s$. Thus there is some $\epsilon > 0$ such that $t = s + \epsilon$. As $s \in \mathbb{Q}^+$, there exist $s_1, s_2 \in \omega$ such that $s = \frac{s_1}{s_2}$. We define a t-gale d as follows:

- Let $d(\lambda) = 1$.
- For all σ such that $ks_2 \leq |\sigma| < ks_2 + s_1$ for any $k = 0, 1, 2, \ldots$, let

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

- For all σ such that $ks_2 + s_1 \leq |\sigma| < ks_2$ for any $k = 0, 1, 2, \ldots$, let

$$\begin{aligned} d(\sigma 1) &= 0 \\ d(\sigma 0) &= 2^t d(\sigma). \end{aligned}$$

By construction, d is a weakly computable t-gale. (Actually, d is a computable martingale).

Consider $d(g(\Omega \upharpoonright ks_1))$ for any $k = 1, 2, \ldots$. Recall that

$$g(\Omega \upharpoonright ks_1) = \Omega[1] \ldots \Omega[s_1]0^{s_2-s_1} \ldots \Omega[(k-1)s_1+1] \ldots \Omega[ks_1]0^{s_2-s_1}.$$

Thus

$$\begin{aligned} d(g(\Omega \upharpoonright ks_1)) &= \left[\left(\frac{2^t}{2}\right)^{s_1}(2^t)^{s_2-s_1}\right]^k d(\lambda) \\ &= \frac{2^{kts_1}2^{kt(s_2-s_1)}}{2^{ks_1}} \\ &= \frac{2^{kts_2}}{2^{ks_1}} \\ &= \frac{2^{k(s+\epsilon)s_2}}{2^{ks_1}} \\ &= \frac{2^{kss_2+k\epsilon s_2}}{2^{ks_1}} \\ &= \frac{2^{ks_1+k\epsilon s_2}}{2^{ks_1}} \\ &= 2^{k\epsilon s_2}. \end{aligned}$$

Thus

$$\begin{aligned} \limsup_{n\to\infty} d(g_s(\Omega) \upharpoonright n) &\geq \limsup_{n\to\infty} d(g(\Omega \upharpoonright n)) \\ &\geq \lim_{k\to\infty} d(g(\Omega \upharpoonright ks_1)) \\ &\geq \lim_{k\to\infty} 2^{k\epsilon s_2} \\ &\geq \infty, \end{aligned}$$

so $\dim(g_s(\Omega)) \leq s$, as required.

Conversely, suppose that there is some t-gale d that succeeds on $g_s(\Omega)$, where $t \leq s$. (We write $t = s - \epsilon$ for some $\epsilon > 0$.) We will prove that there is a supermartingale f that succeeds on Ω. As Ω is random, this is clearly a contradiction.

For all $\sigma \in 2^{<\omega}$, define f via

$$f(\sigma) = 2^{|\sigma|-t|g(\sigma)|}d(g(\sigma)).$$

Then $f(\lambda) = d(\lambda)$ and

$$\begin{aligned} f(\sigma 0) + f(\sigma 1) &= 2^{|\sigma 0|-t|g(\sigma 0)|}d(g(\sigma 0)) + 2^{|\sigma 1|-t|g(\sigma 1)|}d(g(\sigma 1)) \\ &= 2.2^{|\sigma|}2^{-t|g(\sigma 0)|}\left[d(g(\sigma 0)) + d(g(\sigma 1))\right], \end{aligned}$$

as $|\tau_1| = |\tau_2| \Rightarrow |g(\tau_1)| = |g(\tau_2)|$. Because d is a t-gale,

$$\sum_{\substack{\tau:\tau \sqsupseteq \sigma \\ |\tau|=|g(\sigma 0)|}} d(\tau) = 2^{t(|g(\sigma 0)|-|g(\sigma)|)} d(g(\sigma)),$$

so

$$d(g(\sigma 0)) + d(g(\sigma 1)) \leq 2^{t(|g(\sigma 0)|-|g(\sigma)|)} d(g(\sigma)).$$

Thus it follows that

$$\begin{aligned} f(\sigma 0) + f(\sigma 1) &\leq 2.2^{|\sigma|} 2^{-t|g(\sigma 0)|} 2^{t(|g(\sigma 0)|-|g(\sigma)|)} d(g(\sigma)) \\ &\leq 2.2^{|\sigma|} 2^{-t|g(\sigma)|} d(g(\sigma)) \\ &\leq 2f(\sigma), \end{aligned}$$

so f is a supermartingale, as required. Consider

$$\limsup_{n\to\infty} f(\Omega \upharpoonright n) = \limsup_{n\to\infty} 2^{n-t|g(\Omega\upharpoonright n)|} d(g(\Omega \upharpoonright n)).$$

Lemma 103 implies that $|g(\Omega \upharpoonright n)| \leq \frac{n}{s}$, and so

$$2^{-t|g(\Omega\upharpoonright n)|} \geq 2^{-\frac{tn}{s}}.$$

Recall that $t = s - \epsilon$, and it follows that

$$\begin{aligned} \limsup_{n\to\infty} f(\Omega \upharpoonright n) &= \limsup_{n\to\infty} 2^n 2^{-\frac{(s-\epsilon)n}{s}} d(g(\Omega \upharpoonright n)) \\ &= \limsup_{n\to\infty} 2^n 2^{-n} 2^{\frac{\epsilon n}{s}} d(g(\Omega \upharpoonright n)) \\ &= \limsup_{n\to\infty} 2^{\frac{\epsilon n}{s}} d(g(\Omega \upharpoonright n)). \end{aligned}$$

Since $\epsilon > 0$, this implies that f succeeds on Ω. Since Ω is random, no such f can exist. Thus there can be no t-gale d that succeeds on $g_s(\Omega)$. Hence we may conclude that $\dim(g_s(\Omega)) \geq s$, as required. □

Corollary 105. *For any $s, t \in \mathbb{Q}^+$ such that $t > s$, there is an*

$$x \in s\text{-}\mathbf{RAND} - t\text{-}\mathbf{RAND}.$$

Proof. Choose $x \in 2^\omega$ such that $\dim(x) = t$. (The existence of such an x is assured by Theorem 104 above). $\dim(x) = t$ implies that t is the least value for which there is a t-gale that succeeds on x. In particular, this implies that $x \in t$-**NULL**, and since $t > s$ there is no s-gale d which succeeds on x; that is, x is s-random. $\square$

Thus we have a dense (nested) hierarchy of s-random classes. For simplicity, the work previously only considers those s-gales where s is a rational number. In fact, the results hold for any computable (indeed, Δ^0_2-computable) real, as shown in [32].

We note in passing that clearly Martin-Löf randomness is equivalent to s-randomness where $s = 1$. Furthermore, if $s > 1$, any s-gale is a supermartingale and by Theorem 36 the s-randomness class is equivalent to the Martin-Löf randoms. This parallels the fact that the Hausdorff dimension of a measure zero (i.e. random) set cannot exceed 1. Finally, we note that the characterization of s-randomness in terms of effective success only makes sense when $0 \leq s < 1$.

There is an equivalent test set definition of s-randomness.

Theorem 106 (Calude, Staiger, Terwijn). *Let Γ be a set of strings. The s-measure of Γ, $\mu_s(\Gamma)$, is defined as:*

$$\mu_s(\Gamma) = \sum_{\gamma \in \Gamma} 2^{-s|\gamma|}.$$

An s-test is a computable collection of computably enumerable sets of strings $\{U_k : k \in \omega\}$ such that for every prefix-free subset $\widehat{U}_k$ of U_k, $\mu_s(\widehat{U}_k) \leq 2^{-k}$. A real x is s-random if and only if for all s-tests, $x \notin \cap_k U_k$.

Proof. Suppose that x is not s-random, so there is an s-gale d such that $\limsup_n d(x \upharpoonright n) = \infty$. Assume without loss of generality that $d(\lambda) \leq 1$, and define an s-test $\{U_k : k \in \omega\}$ by $U_k = \{\sigma : d(\sigma) \geq 2^k\}$. As for Theorem 51, U_k is a c.e. set and for all k, $x \in U_k$. To show that U_k is an s-test, take any prefix-free set $C \subseteq U_k$.

$$\begin{aligned}\sum_{\sigma \in C} 2^{-s|\sigma|} &= 2^{-k} \sum_{\sigma \in C} 2^{-s|\sigma|} 2^k \\ &\leq 2^{-k} \sum_{\sigma \in C} 2^{-s|\sigma|} d(\sigma) \\ &\leq 2^{-k},\end{aligned}$$

from Lemma 100.

Conversely, suppose that there is an s-test such that $x \in \cap_k U_k$. To prove that there is an s-gale d that succeeds on x, assume that for all k, $U_k = \{\sigma_{k,m} : m \in \omega\}$. (Note that unlike Theorem 51 and elsewhere, we do not require that this set of strings be prefix-free). For all $k \in \omega$ let $\mathcal{C}_k$ to be the class of all prefix-free subsets of U_k and define for all $\sigma \in 2^{<\omega}$,

$$d_k(\sigma) = 2^{s|\sigma|} \max_{C \in \mathcal{C}_k} \Big\{ \sum_{\tau \in C \cap [\sigma]} 2^{-s|\tau|} \Big\}.$$

d_k is weakly computable since it can only increase as we enumerate more elements of U_k.

Now

$$\begin{aligned}d_k(\sigma) &= 2^{s|\sigma|} \max_{C \in \mathcal{C}_k} \Big\{ \sum_{\tau \in C \cap [\sigma]} 2^{-s|\tau|} \Big\} \\ &= 2^{s|\sigma|} \max_{C \in \mathcal{C}_k} \Big\{ \sum_{\tau \in C \cap [\sigma 0]} 2^{-s|\tau|} + \sum_{\tau \in C \cap [\sigma 1]} 2^{-s|\tau|} \Big\} \\ &\leq 2^{s|\sigma|} \Big[\max_{C \in \mathcal{C}_k} \Big\{ \sum_{\tau \in C \cap [\sigma 0]} 2^{-s|\tau|} \Big\} + \max_{C \in \mathcal{C}_k} \Big\{ \sum_{\tau \in C \cap [\sigma 1]} 2^{-s|\tau|} \Big\} \Big] \\ &\leq 2^{-s} \big[d_k(\sigma 0) + d_k(\sigma 1) \big],\end{aligned}$$

while if C_0 is the prefix-free set that maximises $d_k(\sigma 0)$ and C_1 is the prefix-free set that maximises $d_k(\sigma 1)$, then we can define $C = (C_0 \cap [\sigma 0]) \cup (C_1 \cap$

$[\sigma 1]$) and

$$\begin{aligned} d_k(\sigma) &\geq 2^{s|\sigma|} \sum_{\tau \in C \cap [\sigma]} 2^{-s|\tau|} \\ &\geq 2^{s|\sigma|} \Big(\sum_{\tau \in C_0 \cap [\sigma 0]} 2^{-s|\tau|} + \sum_{\tau \in C_1 \cap [\sigma 1]} 2^{-s|\tau|} \Big) \\ &\geq 2^{-s} \big[d_k(\sigma 0) + d_k(\sigma 1) \big]. \end{aligned}$$

Thus for each k,

$$2^s d_k(\sigma) = d_k(\sigma 0) + d_k(\sigma 1)$$

and so d_k is an s-gale.

Set $d(\sigma) = \sum_k 2^k d_{2k}(\sigma)$. Since d_{2k} is a weakly computable s-gale for all k, so is d. Furthermore,

$$\begin{aligned} d(\lambda) &= \sum_k 2^k d_{2k}(\lambda) \\ &= \sum_k 2^k \max_{C \in \mathcal{C}_{2k}} \Big\{ \sum_{\tau \in C} 2^{-s|\tau|} \Big\} \\ &\leq \sum_k 2^k 2^{-2k} \\ &\leq \sum_k 2^{-k}, \end{aligned}$$

so $d(\lambda)$ is bounded (by 2). We know that for all k, $x \in U_k$ i.e.

$$(\forall k)(\exists n) x \upharpoonright n \in U_{2k}.$$

Thus

$$\begin{aligned} d(x \upharpoonright n) &\geq 2^k d_{2k}(x \upharpoonright n) \\ &\geq 2^k. \end{aligned}$$

Since this is true for any k, d is an s-gale that succeeds on x. □

There is also a machine definition of s-randomness.

Definition 59. *Let $f : 2^{<\omega} \to 2^{<\omega}$ be a function, and let $Q \subseteq \mathrm{ra}(f)$. Then $Q^* \subseteq \mathrm{dom} f$ is defined by:*

- *If $y \in Q$ then $(\exists x \in Q^*)\big[f(x) = y\big]$.*
- *If there exists $z \in \mathrm{dom} f$, $z \neq x$ such that $f(z) = y$ then $|z| \geq |x|$.*
- *If f is computed by the Turing Machine M then if there exists $z \in \mathrm{dom} f$ such that $|z| = |x|$ and $f(z) = y$ but $z \neq x$, then there is a stage s such that $x \in \mathrm{dom}(M_s)$ and $z \notin \mathrm{dom}(M_s)$. (Assume for simplicity that $|\mathrm{dom}(M_{s+1}) - \mathrm{dom}(M_s)| \leq 1$).*

Theorem 107. *A machine M is s-measurable if and only if for all prefix-free subsets $Q \subseteq \mathrm{ra}(M)$, $\mu_s(Q^*) \leq 1$. A real x is s-random if and only if for all s-measurable machines M,*

$$(\exists c)(\forall n) H_M(x \upharpoonright n) \geq n - c.$$

To prove this theorem we require the following Lemma. For the proof of this Lemma, see the Appendix.

Lemma 108. *If we have an s-test $\{U_k : k \in \omega\}$ then there is an s-measurable machine M that maps strings of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$ where for each k,*

$$U_k = \{\sigma_{k,i} : i \in \omega\}.$$

As for the effective Kraft-Chaitin algorithm we can define an s-measurable machine, given an (effective) list of requirements $< \sigma_k, n_k >$ as described in Lemma 108. We now turn to the proof of Theorem 107.

Proof. Suppose that there is an s-measurable machine M such that

$$(\forall c)(\exists n) H_M(x \upharpoonright n) < n - c.$$

For all k define a set U_k via

$$U_k = \{\sigma : H_M(\sigma) < |\sigma| - \frac{k}{s}\}.$$

As in Theorem 51, these sets are c.e. and x is contained in U_k for all k. It remains to be shown that $\{U_k : k \in \omega\}$ is an s-test. Take any prefix-free set $C \subseteq U_k$. Note that by definition of U_k, $C \subseteq \mathrm{ra}(M)$ so C^* is well-defined, and furthermore, for any $\sigma \in C$, $|\sigma^*| = H(\sigma)$.

$$\begin{aligned}
\mu_s(C) &= \sum_{\sigma \in C} 2^{-s|\sigma|} \\
&\leq \sum_{\sigma \in C} 2^{s(-H(\sigma) - \frac{k}{s})} \\
&\leq \sum_{\sigma \in C} 2^{-k} 2^{-sH(\sigma)} \\
&\leq 2^{-k} \sum_{\sigma \in C^*} 2^{-s|\sigma^*|} \\
&\leq 2^{-k} \mu_s(C^*) \\
&\leq 2^{-k},
\end{aligned}$$

because C is prefix-free and M is s-measurable.

Conversely, suppose that x is not s-random, so there is an s-test

$$\{U_k : k \in \omega\}$$

such that $x \in \bigcap_k U_k$. Lemma 108 ensures that is it possible to define a machine M that maps strings of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$ where for each k, $U_k = \{\sigma_{k,i} : i \in \omega\}$. This ensures that

$$(\forall k)(\exists j) H_M(x \upharpoonright j) < j - k$$

as required. □

As for Martin-Löf randomness, we have a universal s-test and a universal s-gale.

Theorem 109. *For every $s \in [0,1]$, there is a universal s-test. That is, there is an s-test $\{U_k : k \in \omega\}$ such that x is s-random if and only if $(\forall k) x \in U_k$.*

Proof. As for Theorem 31, with the obvious alterations. □

Theorem 110. *For every $s \in [0,1]$, there is a universal weakly computable s-gale.*

Proof. As for Theorem 35. □

16.3 Another approach

Here we consider another approach to a similar concept s-randomness that does not have the theoretical grounding in measure theory that the above work has, but is perhaps more intuitively appealing. This considers the problem from the machine characterization, rather than the martingale one.

Suppose σ is a string that had a property such as that every 10-th bit was 1. Then according to every type of randomnes we have discussed so far, this would not be random. For example, it would be straightforward to define a martingale that succeeded on it. However, if the 9 bits in between these 1's were totally chaotic, we would like to be able to say that this string displays chaoticness, to a certain extent. Weak s-randomnes is a way of doing this. It considers the initial-segment complexity of such a string. 9 out of each 10 bits of a reasonably long initial segment would be completely chaotic, and so the shortest description would be to write out these random bits in full, with a coding in to put 1 in at every 10th place. This would take a description of length $0.9n+c$ where n is the length of the string, and c is a subprogram of constant length that advises the computer to put 1 in at every 10th bit. This suggests the following definition:

Definition 60 (Tadaki, [47]). *Let $s \geq 0$ be a real number. A real x is* weakly s-random *if*

$$(\exists c)(\forall n)H(x \upharpoonright n) \geq sn - c.$$

There is an equivalent test set definition.

Theorem 111 (Tadaki, [47]). *Define a weak s-test to be a computable collection of c.e. sets $\{U_n : n \in \omega\}$ such that $\mu_s(U_n) \leq 2^{-n}$ for all n. A real x is weakly s-random if for all weak s-tests, $x \notin \cap_n U_n$.*

Proof. As for Theorem 51. Suppose that x is not weakly s-random, so that for all constants k there is an n such that $H(x \upharpoonright n) < sn - k$. Define

$$U_k = \{\sigma : H(\sigma) < s|\sigma| - k\}.$$

Clearly,

$$\begin{aligned}\mu_s(U_k) &= \sum_{\sigma\in U_k} 2^{-s|\sigma|}\\ &\leq \sum_{\sigma\in U_k} 2^{-H(\sigma)-k}\\ &\leq 2^{-k}\sum_{\sigma\in U_k} 2^{-H(\sigma)}\\ &\leq 2^{-k},\end{aligned}$$

since the domain of the machine is prefix free. Also, U_k is computably enumerable, so $\{U_k : k \in \omega\}$ is a weak s-test. Furthermore,

$$(\forall k)(\exists n)H(x \upharpoonright n) < sn - k,$$

so

$$(\forall k)(x \in U_k),$$

and x is not weakly s-random, as required.

Conversely, suppose that x is there is a weak s-test such that for all k, $x \in U_k$. For each $\sigma_{2k+2,i} \in U_{2k+2}$, request that M maps a string of length $s|\sigma_{2k+2,i}| - (k+1)$. This is well-defined because $s|\sigma_{2k+2,i}| \geq 2k+2$. Furthermore,

$$\begin{aligned}\sum_{\sigma\in \mathrm{dom} M} 2^{-|\sigma|} &= \sum_{k\in\omega}\sum_{i\in\omega} 2^{-s|\sigma_{2k+2,i}|}2^{(k+1)}\\ &= \sum_{k\in\omega} 2^{(k+1)}\mu_s(U_{2k+2})\\ &\leq \sum_{k\in\omega} 2^{(k+1)}2^{-(2k+2)}\\ &\leq \sum_{k\in\omega} 2^{-k-1}\\ &\leq 1.\end{aligned}$$

Thus M defined here is a prefix-free machine.

$$\begin{aligned}\text{Now, } (\forall k)(x \in U_k) &\Rightarrow (\forall k)(\exists i)(x \in [\sigma_{2k+2,i}])\\ &\Rightarrow (\forall k)(\exists i,j)(x \upharpoonright j = \sigma_{2k+2,i})\\ &\Rightarrow (\forall k)(\exists i,j)(H(x \upharpoonright j) = sj - k - 1)\\ &\Rightarrow (\forall k)(\exists j)(H(x \upharpoonright j) < sj - k),\end{aligned}$$

and so x is not weakly s-random, as required. □

There is a universal weak s-test.

Theorem 112. *For any $s \in \mathbb{Q}$ there is a weak s-test $\{U_k : k \in \omega\}$ such that*

$$x \textit{ is weakly } s\textit{-random} \iff x \notin \cap_{k\in\omega} U_k.$$

Proof. Let $\{U_k : k \in \omega\}$ be the weak s-test defined Theorem 111 above. If $x \in \cap_k U_k$ then clearly x is not weakly s-random. If x is not weakly s-random then by the construction of U_k, $x \in \cap_k U_k$. Thus it follows that

$$x \text{ is not weakly } s\text{-random} \iff x \in \cap_{k\in\omega} U_k.$$

The theorem is just the negation of this statement. □

The test set and machine definitions of weak randomness follow naturally from the regular test set and machine definitions of Martin-Löf randomness. However, if we attempt to suggest a martingale definition, the equivalent definition fails. Because the definition of a weak s-test is virtually identical to that of a Martin-Löf test (with just an alteration in the type of measure used), the kind of s-gale that can be generated from a weak s-test will be similar to the weakly computable martingale generated from a Martin-Löf test. However, this s-gale will not be strong enough to generate a corresponding s-test from it. The usual method of defining a test set from a martingale is

$$U_k = \{\sigma : d(\sigma) \geq 2^k\}.$$

In the standard Martin-Löf definition, it is always possible to effectively enumerate a prefix-free set of strings representing U_k. For example, suppose that $d_s(\sigma 0) \geq 2^k$ at a stage s. In that case we put $\sigma 0$ into $U_{k,s}$. Then at stage $t > s$, if $d_t(\sigma) \geq 2^k$ we put $\sigma 1$ into $U_{k,t}$. It is necessary to do this, because as we are working with weakly computable martingales it is not always possible at any given stage to know whether or not $d(\sigma) > 2^k$. It seems natural to replace σ by $\sigma 0$ and $\sigma 1$, because obviously $\mu(\sigma) = \mu(\sigma 0) + \mu(\sigma 1)$, and $[\sigma] = [\sigma 0] \cup [\sigma 1]$. However, if we are building an s-test it is may not be possible to cover $[\sigma]$ with $[\sigma 0]$ and $[\sigma 1]$, because

$$\mu_s(\sigma) = 2^{-s|\sigma|}$$

whilst

$$\mu_s(\sigma 0) + \mu_s(\sigma 1) = 2.2^{-s(|\sigma|+1)}$$

If $s < 1$ then $\mu_s(\sigma 0) + \mu_s(\sigma 1) > \mu_s(\sigma)$. In the worst case, it could happen that our s-gale d succeeds on a real x, but it will appear to be random in the test set characterization, because the test set that is intended to capture it has too large an s-measure and so is rejected. This difficulty in generating a martingale characterization makes weak s-randomness less acceptable as a definition of randomness, and preference must be given to the s-randomness characterizations discussed in Section 16.2.

Theorem 113. *Let s-**W-RAND** be the set of weakly s-random reals. Then for any $0 < s < 1$, s-**RAND** $\subseteq$ s-**W-RAND**.*

Proof. We prove that if a real is not weakly s-random, then it is not s-random. Suppose that we have x such that

$$(\forall c)(\exists n) H_M(x \upharpoonright n) < sn - c,$$

where M is the usual Universal Prefix-free Machine. Define a machine N as follows: if $M(\sigma)\downarrow$, pick a string $f(\sigma)$ of length $\lfloor \frac{|\sigma|+1}{s} \rfloor$ and set $N(f(\sigma)) = M(\sigma)$.

Note that since $s < 1$,

$$\frac{m+1}{s} > m+1,$$

and so

$$\lfloor \frac{m+1}{s} \rfloor \geq m+1.$$

Also,

$$\begin{aligned} \frac{m+1}{s} &= \frac{m}{s} + \frac{1}{s} \\ &> \frac{(m-1)+1}{s} + 1. \end{aligned}$$

Thus

$$\lfloor \frac{m+1}{s} \rfloor \neq \lfloor \frac{(m-1)+1}{s} \rfloor.$$

Thus for all $\sigma \in \mathrm{dom}(M)$, $|f(\sigma)| > |\sigma|$. Furthermore, if $\sigma, \tau \in \mathrm{dom}(M)$ and $|\sigma| \neq |\tau|$, then $|f(\sigma)| \neq |f(\tau)|$. Thus we won't run out of strings of the correct length to use in $\mathrm{dom}(N)$.

Now, $|f(\sigma)| \leq \frac{|\sigma|+1}{s}$ but $|f(\sigma)| > \frac{|\sigma|+1}{s} - 1$. Thus

$$\begin{aligned} \sum_{\tau \in \mathrm{dom}(N)} 2^{-s|\tau|} &< \sum_{\sigma \in \mathrm{dom}(M)} 2^{-s(\frac{|\sigma|+1}{s} - 1)} \\ &< \sum_{\sigma \in \mathrm{dom}(M)} 2^{-|\sigma|-1+s} \\ &< \frac{2^s}{2} \sum_{\sigma \in \mathrm{dom}(M)} 2^{-|\sigma|} \\ &< \frac{2^s}{2} \\ &< 1, \end{aligned}$$

as M is a prefix-free machine, and $s < 1$ so $2^s < 2$. Because the s-measure of the domain is less than 1, it follows that for any subset A of $\mathrm{ra}(N)$, $\mu_S(A^*) < 1$ as well. Thus N is an s-measurable machine.

Since for all c there is an n such that $H_M(x \upharpoonright n) < sn - c$ if follows that

$$\begin{aligned} H_N(x \upharpoonright n) &< \lfloor \frac{sn - c + 1}{s} \rfloor \\ &< \frac{sn - c + 1}{s} \\ &< n - \frac{c - 1}{s} \end{aligned}$$

and so x is not s-random. □

It is possible to generalize Chaitin's halting probability, Ω, in the case of s-randomness. Ω^s is defined as

$$\Omega^s \equiv \sum_{U(\sigma)\downarrow} 2^{-\frac{|\sigma|}{s}}.$$

Theorem 114 (Tadaki, [47]). *If* $0 < s \leq 1$ *,* Ω^s *is weakly* s*-random.*

Proof. The proof of this fact is analogous to Theorem 55. □

For weak s-randomness, if we set $s = 1$, the test set definition and machine definition are equivalent to Martin-Löf randomness and Chaitin randomness respectively. Furthermore, $\Omega^1 = \Omega$.

Chapter 17

Conclusion

Although much of the work being undertaken relating to randomness is of great technical interest, the question must be asked: does it provide insight into the nature of random sequences at all? For example, Church randomness is rejected as a valid notion of randomness due to the failure of some Church random sequences to stand up to some well-known tests of randomness (precisely, the law of the iterated logarithm). Thus, although the class of Church random sequences may be worthy of study in its own right, it is unacceptable as a true measure of randomness.

It is only fair that we subject the other types of randomness to similar scrutiny, and thus we can immediately eliminate Kurtz randomness, as there are Kurtz random sequences that don't even satisfy the law of large numbers (see [28], [23]).

Schnorr randomness fares somewhat better, as all Schnorr random sequences satisfy both the law of large numbers, and the law of the iterated logarithm (see [40]). This shows, among other things, that there are Church random sequences that are not Schnorr random. However, it was proved by Wang [54] that Schnorr randomness is not a proper strengthening of Church randomness, i.e. there are there are Schnorr random sequences that are not Church random. Specifically, Wang constructed a

Schnorr random real x and a computable place selection rule ϕ such that

$$\phi(x) = 1111\ldots$$

Since Church randomness seems to be a *necessary* (though clearly not a sufficient) requirement for randomness, this implies that Schnorr randomness is not a legitimate definition of randomness. As an interesting aside, the random real used in this proof is precisely the real x constructed in Theorem 82 in Chapter 14. In that proof x was shown to be Schnorr random but not *computably* random. Now we see that x is, in fact, not Church random either.

This new result actually subsumes that first one as it was shown (by Schnorr) that all computably random reals are Church random. This suggests that computable randomness is a more successful definition than either Schnorr or Kurtz randomness. The trouble with computable randomness is something of an aesthetic one, but we believe that it is of significance. The inelegance of the test set definition of computable randomness and the lack of a machine version is worrying. The intuitive notions behind these characterizations are compelling, and computable randomness doesn't seem to capture this. We note that this problem may be solved through further scholarship, but at present this makes computable randomness unattractive as an "official" notion of randomness.

We have covered Schnorr's criticism of Martin-Löf randomness in Chapter 9. Schnorr's arguments are strong, and it is probable that the unassailable position Martin-Löf randomness seems to have at the top of the popularity scale is due at least in part to the fact that it came first, but unfortunately the simplicity of the characterizations of Martin-Löf randomness - martingale, test set, *and* machine - render it superior to computable randomness, at least for the time being.

Of course, none of the characterizations should be summarily rejected, as all of them are technically interesting and defensible on some grounds or other, and each have their place as manifestations of randomness in various areas of mathematics. The different rôles each type of randomness can play is commented on in [1]. We also must observe that this work is by no means an exhaustive survey of all possible definitions of algorithmic randomness, and that other randomness definitions are possible. Unfortunately space does not permit us to do justice to them here.

Appendix A

Technical Results

This appendix contains some useful lemmas. Most are mentioned in the main text but some are included here because the appendix itself requires them.

Proof of Lemma 2

Recall Lemma 2. *For every total computable function f there is a corresponding "fixed point" n such that $\phi_n = \phi_{f(n)}$, where ϕ_n is the n-th Turing machine in the standard enumeration.*

Proof. Define a computable function ϕ via

$$\phi(u, z) = \begin{cases} \phi_{\phi_u(u)}(z) & \text{if } \phi_u(u) \downarrow \\ \uparrow & \text{otherwise.} \end{cases}$$

By the *s-m-n* theorem (Lemma 1) there is a total computable function d such that

$$\phi_{d(u)} = \phi(u, \_).$$

Because the composition of two computable functions is a computable function, there is a computable function ϕ_e in the standard enumeration such that $\phi_e(\_) = f(d(\_))$. Since f and d are total, ϕ_e is total and in particular, $\phi_e(e) \downarrow$. This means that $\phi_{d(e)} = \phi_{\phi_e(e)}$ from the definition of d, and $n = d(e)$ is a fixed point for f as

$$\begin{aligned}\phi_n &= \phi_{d(e)} \\ &= \phi_{\phi_e(e)} \\ &= \phi_{f(d(e))} \\ &= \phi_{f(n)},\end{aligned}$$

as required. $\square$

Proof of Lemma 21

Recall Lemma 21. *Let $\{X_i : i \in \omega\}$ be defined as in Proposition 20. Then for any $\epsilon \in \mathbb{Q}^+$ there is a $k \in \omega$ such that*

$$\sum_{i \geq k} \mu(X_i) < \epsilon.$$

Proof. Suppose that this is not the case; that is, for some rational $\epsilon > 0$, for any $k \in \omega$

$$\sum_{i \geq k} \mu(X_i) \geq \epsilon.$$

Pick the first j_1 such that

$$\sum_{i < j_1} \mu(X_i) \geq \epsilon.$$

By assumption

$$\sum_{i \geq 0} \mu(X_i) \geq \epsilon$$

, so such a stage j_1 will exist. Furthermore,

$$\sum_{i \geq j_1} \mu(X_i) \geq \epsilon,$$

so pick the first $j_2 > j_1$ such that

$$\sum_{j_1 \leq i < j_2} \mu(X_i) \geq \epsilon.$$

Thus

$$\sum_{i < j_2} \mu(X_i) \geq 2\epsilon.$$

This process can be repeated indefinitely. For any $M \in \omega$ there is an $n \in \omega$ such that $n\epsilon \geq M$ and so

$$\sum_{i < j_n} \mu(X_i) \geq M.$$

Thus

$$\sum_{i} \mu(X_i) = \infty.$$

This contradicts the definition of $\{X_i : i \in \omega\}$, and so the Lemma is proved.

□

Proof of Lemma 34

Lemma 115. *Let f be a martingale. Then for all $x \in 2^{<\omega}$,*

$$\sum_{|x|=n} f(x) = 2^n f(\lambda),$$

where λ is the empty string.

Proof. This is proved via induction on n.

Base case: Let $n = 0$. Then

$$\begin{aligned} \sum_{|x|=0} f(x) &= f(\lambda) \\ &= 2^0 f(\lambda), \end{aligned}$$

as required.

Induction step: The induction hypothesis is that

$$\sum_{|x|=k} f(x) = 2^k f(\lambda).$$

We wish to prove it holds for $n = k + 1$.

$$\begin{aligned}
\sum_{|x|=k+1} f(x) &= \sum_{|x|=k} f(x0) + f(x1) \\
&= \sum_{|x|=k} 2f(x) \\
&= 2 \sum_{|x|=k} f(x) \\
&= 2^{k+1} f(\lambda),
\end{aligned}$$

as required. ☐

Lemma 116. *Let f be a martingale. Then for all $x \in 2^{<\omega}$, $f(x) \leq 2^{|x|} f(\lambda)$.*

Proof. Follows from Lemma 115. ☐

Lemma 117. *Let $X \subseteq 2^{<\omega}$ be a prefix-free set. Then*

$$f(\lambda) \geq \sum_{x \in X} 2^{-|x|} f(x).$$

Proof. Let X be a prefix-free set and f be a martingale. Recall that, from Lemma 115 that if one considers *all* strings of length n, the martingale property ensures that

$$\sum_{|x|=n} f(x) = 2^n f(\lambda).$$

However, we are interested only in the sum of strings in X:

$$\sum_{\substack{x \in X \\ |x|=n}} f(x).$$

Because X is prefix-free, what strings of length n can be in X depends on what shorter strings are in X - any string y of length $j < n$ in X prohibits 2^{n-j} strings of length n ever entering X. The amount these extensions of y could add to the above sum is $2^{n-|y|}f(y)$, because f is a martingale.

Thus

$$\begin{aligned}\sum_{\substack{x\in X\\|x|=n}} f(x) &\leq 2^n f(\lambda) - \sum_{\substack{|x|=n\\(\exists y\in X)(y\sqsubset x)}} f(x)\\ &\leq 2^n f(\lambda) - \sum_{\substack{y\in X\\|y|\leq n-1}} 2^{n-|y|} f(y).\end{aligned}$$

By multiplying through by a factor of 2^{-n}, we get:

$$\begin{aligned}\sum_{\substack{x\in X\\|x|=n}} 2^{-n} f(x) &\leq 2^{-n}2^n f(\lambda) - 2^{-n} \sum_{\substack{y\in X\\|y|\leq n-1}} 2^{n-|y|} f(y)\\ &\leq f(\lambda) - \sum_{\substack{y\in X\\|y|\leq n-1}} 2^{-|y|} f(y),\end{aligned}$$

which rearranges to give

$$\sum_{\substack{x\in X\\|x|\leq n}} 2^{-|x|} f(x) \leq f(\lambda).$$

Then

$$\begin{aligned}\sum_{x\in X} 2^{-|x|} f(x) &= \lim_{n\to\infty} \sum_{\substack{x\in X\\|x|\leq n}} 2^{-|x|} f(x)\\ &\leq \lim_{n\to\infty} f(\lambda)\\ &\leq f(\lambda),\end{aligned}$$

as required. □

Recall Lemma 34. *Let f be a martingale. Define the set $F_k = \{x \in 2^{<\omega} : f(x) > k\}$. Define $G_k = \bigcup_{x\in F_k}\{y : y \in [x]\}$, where $[x]$ is the set containing all extensions of x. Then*

$$\mu(G_k) \leq \frac{1}{k} f(\lambda).$$

Proof. There is a prefix-free set H_k such that

$$\cup_{\sigma \in H_k}[\sigma] = \cup_{\sigma \in F_k}[\sigma].$$

Clearly

$$\mu(G_k) = \mu(H_k) = \sum_{x \in H_k} 2^{-|x|}.$$

From this fact and Lemma 117,

$$\begin{aligned} f(\lambda) &\geq \sum_{x \in H_k} 2^{-|x|} f(x) \\ &> k \sum_{x \in H_k} 2^{-|x|} \\ &> k\mu(G_k), \end{aligned}$$

as required. □

Proof of Lemma 38

Recall Lemma 38. *Suppose that f is a computable, unbounded and non-decreasing function and that $\{U_n : n \in \omega\}$ is a Schnorr test. For each n consider each U_n to be a prefix-free collection of strings, $U_n = \{\sigma_{n,i} : i \in \omega\}$. Suppose that each U_n has the property that if*

$$S_n = \sum_{i \in \omega} 2^{-|\sigma_{n,i}|} 2^{f(|\sigma_{n,i}|)}$$

then $S_n \leq 2^{-n}$ and S_n is a computable real. Then there is a computable martingale d such that if $x \in \bigcap_n U_n$ then $d(x \upharpoonright n) \geq 2^{f(n)}$ infinitely often.

Proof. Define

$$d(\sigma) = \sum_n w_n(\sigma)$$

where for each $n \in \omega$, $w_n(\sigma)$ is a weighting function defined via

$$w_n(\sigma) = \begin{cases} 2^{f(|\sigma_{n,i}|)} & \text{if } (\exists i \in \omega)\sigma_{n,i} \sqsubseteq \sigma \\ \sum_{\sigma\tau \in U_n} 2^{-|\tau|} 2^{f(|\sigma\tau|)} & \text{otherwise.} \end{cases}$$

It can be shown that w_n satisfies the martingale property and so d will also satisfy it. Note that

$$\begin{aligned} d(\lambda) &= \sum_n w_n(\lambda) \\ &= \sum_n \sum_{\tau \in U_n} 2^{-|\tau|} 2^{f(|\tau|)} \\ &= \sum_n S_n \\ &\leq \sum_n 2^{-n}, \end{aligned}$$

from the assumption in the statement of the Lemma. Hence $d(\lambda)$ is bounded (by 2) and it is also non-zero (as the intersection of the Schnorr test sets is non-empty).

Note that if $\sigma \in U_n$, then $w_n(\sigma) = 2^{f(|\sigma|)}$ and so $d(\sigma) \geq 2^{f(|\sigma|)}$. We know that for all n, $x \in \bigcap_n U_n$. That is, for all n, there exists a j such that $x \upharpoonright j \in U_n$, and so $d(x \upharpoonright j) \geq 2^{f(j)}$ for infinitely many j.

If $|\sigma| = k$, then because for all n, $\mu(U_n) \leq 2^{-n}$, if $n > k$ there can be no i such that $\sigma_{n,i} \sqsubset \sigma$. Thus for any s (assuming that $s > k$), compute U_n for all $n \leq s + 1$ until a stage where

$$|S_n - S_{n,s}| < \frac{2^{-(s+1)}}{s+2}.$$

(This is possible because S_n is a computable real). If we define $d_s(\sigma)$ as above for $d(\sigma)$ but only considering $U_{n,s}$ for $n \leq s + 1$ instead of U_n for all

n, then

$$\begin{aligned}
d(\sigma) - d_s(\sigma) &\leq \sum_{n} \sum_{\sigma\tau \in U_n - U_{n,s}} 2^{-|\tau|} 2^{f(|\sigma\tau|)} \\
&\leq \sum_{n \leq s+1} \sum_{\sigma\tau \in U_n - U_{n,s}} 2^{-|\tau|} 2^{f(|\sigma\tau|)} + \sum_{n > s+1} \sum_{\sigma\tau \in U_n} 2^{-|\tau|} 2^{f(|\sigma\tau|)} \\
&\leq \sum_{n \leq s+1} |S_n - S_{n,s}| + \sum_{n} S_n \\
&\leq \sum_{n \leq s+1} \frac{2^{-(s+1)}}{s+1} + \sum_{n > s+1} 2^n \\
&\leq (s+2)\frac{2^{-(s+1)}}{s+2} + 2^{-(s+1)} \\
&\leq 2^{-s},
\end{aligned}$$

so d is a computable martingale. □

Proof of Lemma 52

Recall Lemma 52. *Define $U_k = \{\sigma : H(\sigma) \leq |\sigma| - k\}$. (Here H can refer to absolute or to function-dependent prefix-free complexity). Then $\mu(U_k) \leq 2^{-k}$.*

Proof. Let M denote the machine for which H is the prefix-free complexity. (If H is the absolute prefix-free complexity, let M denote the appropriate Universal Prefix-Free machine). Since

$$\mu(U_k) = \sum_{\sigma \in U_k} 2^{-|\sigma|},$$

and $\sigma \in U_k$ implies that $|\sigma| \geq H(\sigma) + k$, it follows that

$$\begin{aligned}
\mu(U_k) &\leq \sum_{\sigma \in U_k} 2^{-H(\sigma)} 2^{-k} \\
&\leq 2^{-k} \sum_{\tau \in \mathrm{dom}(M)} 2^{-|\tau|} \\
&\leq 2^{-k},
\end{aligned}$$

as M is a prefix-free machine. □

Proof of Lemma 53

Recall Lemma 53. *Let U_k be a test set such that $\mu(U_k) \leq 2^{-k}$. U_k can be considered as a collection of extensions of strings,*

$$U_k = \bigcup_{i\in\omega}[\sigma_{k,i}],$$

where $\{\sigma_{k,i} : i \in \omega\}$ is a prefix-free set of strings. Then one can define a prefix free machine M that for all $k, i \in \omega$ maps a string of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$.

Proof. Note that the lengths $|\sigma_{2k+2,i}| - (k+1)$ are well-defined, because for each $k, i \in \omega$, $2^{-|\sigma_{2k+2,i}|} \leq 2^{-(2k+2)}$; thus it follows that $|\sigma_{2k+2,i}| \geq 2k+2$, and so $|\sigma_{2k+2,i}| - (k+1) > 0$. Then,

$$\begin{aligned}
\sum_{\sigma\in\mathrm{dom}M} 2^{-|\sigma|} &= \sum_{k\in\omega}\sum_{i\in\omega} 2^{-|\sigma_{2k+2,i}|}2^{k+1} \\
&= \sum_{k\in\omega} 2^{k+1}\sum_{i\in\omega} 2^{-|\sigma_{2k+2,i}|} \\
&= \sum_{k\in\omega} 2^{k+1}\mu(U_{2k+2}) \\
&\leq \sum_{k\in\omega} 2^{k+1}2^{-2k-2} \\
&\leq \frac{1}{2}\sum_{k\in\omega} 2^{-k} \\
&\leq 1,
\end{aligned}$$

so by using the effective Kraft-Chaitin algorithm it is possible to define such a prefix-free machine. □

Proof of Lemma 59

Recall Lemma 59. *Let x, y be binary strings such that $|x| = |y| = n$ and $|x - y| \leq 2^{k-n}$ where $k \in \omega$. There is a constant c_k (depending on k) such that $H(y) \leq H(x) + c_k$.*

Proof. Note that
$$|x-y| = \sum_{i\le n} 2^{-i}|x[i]-y[i]|.$$
If there is a $i < (n-k)$ such that $x[i] \neq y[i]$, then

$$\begin{aligned} |x-y| &\ge 2^{-i} \\ &> 2^{-(n-k)} \\ &> 2^{k-n}, \end{aligned}$$

so for x and y to satisfy the above condition, they must not differ on any bits before the $(n-k)$-th bit. If $x[n-k] \neq y[n-k]$ then $|x-y| \ge 2^{k-n}$ so x and y may *only* differ on that bit. If x is given there is only one string that satisfies this condition. However, if x and y do not differ on any bits up to the $n-k$-th bit, it follows that

$$\begin{aligned} |x-y| &\le \sum_{i=(n-k+1)}^{n} 2^{-i} \\ &\le 2^{k-n}, \end{aligned}$$

as required. Thus any of the $2^{n-(n-k+1)} = 2^{k-1}$ strings that only differ from x after the $n-k$-th bit (plus the special case noted above), satisfy the conditions of the Lemma. We can list these strings (via the lexicographic ordering, for example) as

$$L(1,x), L(2,x), \ldots L(2^{k-1}+1, x).$$

Now suppose that we have a description p of x, such that $U(p) = x$. We define a prefix-free function ϕ such that $\phi(i\tau)$ computes $L(i, U(\tau))$. $i\tau$ can be coded in a prefix-free way that will only increase its length by $\mathcal{O}(\log i)$, and note that for any $i \le 2^{k-1}+1$,

$$\phi(ip) = L(i, U(p)) = L(i, x).$$

Thus for any $y = L(i, x)$,

$$\begin{aligned} H(y) &\leq |ip| \\ &\leq H(x) + |i| + \mathcal{O}(\log i) + \mathcal{O}(1) \\ &\leq H(x) + \mathcal{O}(k), \end{aligned}$$

as required. □

Although we are assuming here that $k < n$, this Lemma works trivially if this is not the case, as then *all* y of length n satisfy the conditions. Then we can define a function $\phi(i)$ that outputs the i-th string in the lexicographic ordering of the strings of length n, which clearly suffices.

Proof of Lemma 63

Recall Lemma 63. *Let x be a c.e. real, and $Y = \{y_0, y_1, y_2, \ldots\}$ be an infinite prefix-free c.e. set. Let*

$$y = \sum_{n \in \omega} 2^{-|y_n|}.$$

If $y \leq_S x$, then we can construct an infinite prefix-free c.e. set X, a partial recursive function f and a constant $c > 0$ such that

$$x = \sum_{\sigma \in X} 2^{-|\sigma|},$$

$\mathrm{dom}(f) = X$, $\mathrm{ra}(f) = Y$, *and* $(\forall \sigma \in X)(|\sigma| \leq |f(\sigma)| + c)$.

Proof. Since x is a c.e. real let $\{x_i : i \in \omega\}$ be a computable sequence of rationals converging to x. Since $y \leq_S x$ it follows that there exists a monotone increasing computable function g and a constant c such that

$$(\forall i)\big[y - \sum_{n=0}^{g(i)} 2^{-|y_n|} \leq 2^c(x - x_i)\big].$$

If

$$\sum_{n=0}^{g(0)} 2^{-|y_n|-c} \not\geq x_0$$

choose a new constant $c_1 \geq c$ such that

$$\sum_{n=0}^{g(0)} 2^{-|y_n|-c_1} \geq x_0.$$

We construct a set $D \subseteq \omega$ in stages. At stage $s \geq 0$, if

$$x_s \leq \sum_{n=0}^{g(s)} 2^{-|y_n|-c} + \sum_{t<s} \sum_{m \in D[t]} 2^{-m},$$

set $D[s] = \emptyset$. Otherwise, we have

$$x_s - \sum_{n=0}^{g(s)} 2^{-|y_n|-c} - \sum_{t<s} \sum_{m \in D[t]} 2^{-m} = p_s > 0.$$

There is a c.e. set such that $\{p_{1,s}, p_{2,s}, \ldots\}$ such that $\sum_i 2^{-p_{i,s}} = p_s$. Set $D[s] = \{p_{1,s}, p_{2,s}, \ldots\}$. Define $D = \cup_s D[s]$. Since for all s, $D[s]$ is a c.e. set, so D is a c.e. set also, and so we can list the elements of $D = \{p_{1,1}, p_{2,1}, \ldots\}$ in the order in which they were enumerated. We define a sequence of natural numbers, $\{n_i : i \in \omega\}$ such that for all i,

$$\begin{aligned} n_{2i} &= |y_i| + c \\ n_{2i+1} &= m_i, \end{aligned}$$

where m_i is the i-th element of D.

Note that for all s,

$$x_s \leq \sum_{i=0}^{g(s)} 2^{-n_{2i}} + \sum_{t \leq s} \sum_{i \in \omega} 2^{-p_{i,t}}.$$

Thus

$$\begin{aligned}
x &= \lim_s x_s \\
&\leq \lim_s \sum_{i=0}^{g(s)} 2^{-n_{2i}} + \sum_{t\leq s}\sum_{i\in\omega} 2^{-p_{i,t}} \\
&\leq \sum_{i\in\omega} 2^{-n_{2i}} + \sum_{i\in\omega} 2^{n_{2i+1}} \\
&\leq \sum_{i\in\omega} 2^{-n_i}.
\end{aligned}$$

Call s an x-stage if $x_s = \sum_{n=0}^{g(s)} 2^{-|y_n|-c} + \sum_{t\leq s}\sum_{m\in D[t]} 2^{-m}$. The following two cases may arise:

1. There are infinitely many x-stages.

2. There are finitely many x-stages.

Note that

$$\sum_{n=0}^{g(s)} 2^{-|y_n|-c} + \sum_{t\leq s}\sum_{m\in D[t]} 2^{-m}$$

is a non-decreasing sum, so if Case 1 occurs, for all r there is an x-stage $s \geq r$ such that

$$\begin{aligned}
\sum_{n=0}^{g(r)} 2^{-|y_n|-c} + \sum_{t\leq r}\sum_{m\in D[t]} 2^{-m} &\leq x_s \\
&\leq x,
\end{aligned}$$

and so

$$\begin{aligned}
x &\geq \lim_{r\to\infty} \sum_{n=0}^{g(r)} 2^{-|y_n|-c} + \sum_{t\leq r}\sum_{m\in D[t]} 2^{-m} \\
&\geq \sum_{i\in\omega} 2^{-n_i}.
\end{aligned}$$

If Case 2 occurs, then go to the last stage s_0 for which

$$x_{s_0} = \sum_{i=0}^{g(s_0)} 2^{-n_{2i}} + \sum_{t \leq s_0} \sum_{i \in \omega} 2^{-p_{i,t}}.$$

Note that this is true for stage 0 if for no other stage. Then $D = \cup_{t \leq s_0} D[t]$, by construction. From above, we know that

$$\begin{aligned} 2^c(x - x_{s_0}) &\geq \sum_{n=0}^{\infty} 2^{-|y_n|} - \sum_{n=0}^{g(s_0)} 2^{-|y_n|} \\ &\geq \sum_{n=g(s_0)+1}^{\infty} 2^{-|y_n|}, \end{aligned}$$

so

$$\begin{aligned} x &\geq \sum_{n=g(s_0)+1}^{\infty} 2^{-|y_n|-c} + x_{s_0} \\ &\geq \sum_{n=g(s_0)+1}^{\infty} 2^{-|y_n|-c} + \sum_{n=0}^{g(s_0)} 2^{-|y_n|-c} + \sum_{m \in D} 2^{-m} \\ &\geq \sum_{n=0}^{\infty} 2^{-|y_n|-c} + \sum_{m \in D} 2^{-m} \\ &\geq \sum_{i \in \omega} 2^{-n_i}. \end{aligned}$$

In either case, $x = \sum_{i \in \omega} 2^{-n_i}$. Among other things, this means that

$$\sum_{i \in \omega} 2^{-n_i} \leq 1$$

(because $x \in [0, 1)$) and so by the effective Kraft-Chaitin algorithm there is a prefix-free set $X = \{a_i : i \in \omega\}$ such that for all i, $|a_i| = n_i$, and $\sum_{i \in \omega} 2^{-|a_i|} = x$. X is infinite because Y is infinite. Finally, let us define a function $h : X \to Y$ such that for all i,

$$\begin{aligned} h(a_{2i}) &= y_i \\ h(a_{2i+1}) &= y_j, \end{aligned}$$

where y_j is some element of Y such that $|y_j| \geq |a_{2i+1}|$. The existence of such a y_j is ensured because Y is infinite and computably enumerable. Note that for all i,

$$\begin{aligned}
|a_{2i}| &= n_i \\
&= |y_i| + c \\
&= |h(a_{2i})| + c \\
|a_{2i+1}| &\leq |h(a_{2i+1})| \\
&\leq |h(a_{2i+1})| + c,
\end{aligned}$$

so all requirements are satisfied. □

Proof of Lemma 66

Recall Lemma 66. *For any d, $n > H(n) + d$ for almost all n.*

Proof. Note that for any d,

$$\begin{aligned}
\sum_{n \geq d+1} 2^{-(n-d)} &= 2^d \sum_{n \geq d+1} 2^{-n} \\
&= 1.
\end{aligned}$$

Thus by the effective Kraft-Chaitin algorithm, one can define a prefix-free machine mapping strings of length $n - d$ to n, for all $n \geq d + 1$. Thus the prefix-free complexity is constrained; in fact, for all $n \geq d + 1$,

$$H(n) \leq n - d,$$

as required. □

Proof of Lemma 78

Recall Lemma 78. *Let F be a computable martingale, and $\sigma \in 2^{<\omega}$ a finite string. There is a computable real y extending σ such that for all $n \geq |\sigma|$, $F(y \upharpoonright n) \leq F(\sigma)$.*

Proof. This is proved via induction on n.

Base case: Suppose $|\sigma| = m$. Define $y \upharpoonright m = \sigma$. Then y extends σ, and $F(y \upharpoonright m) \leq F(\sigma)$.

Induction step: The induction hypothesis is that for k where $k > m$, $y \upharpoonright k$ extends σ and $F(y \upharpoonright k) \leq F(\sigma)$. Because of the martingale property, either

$$F((y \upharpoonright k)0) \leq F(\sigma)$$

or

$$F((y \upharpoonright k)1) \leq F(\sigma).$$

Define

$$y \upharpoonright (k+1) = \begin{cases} (y \upharpoonright k)0 & \text{if } F((y \upharpoonright k)0) \leq F((y \upharpoonright k)1) \\ (y \upharpoonright k)1 & \text{otherwise} \end{cases}$$

Note that

$$\begin{aligned} F(y \upharpoonright (k+1)) &\leq F(y \upharpoonright k) \\ &\leq F(\sigma) \end{aligned}$$

from the induction hypothesis. y is computable because F is a computable martingale □

Proof of Lemma 79

Recall Lemma 79. *Let $F_0, F_1, \ldots, F_m$ be a sequence of computable martingales, $c_0, c_1, \ldots, c_m$ a sequence of constants, and $\sigma \in 2^{<\omega}$ a finite string. Then there is a computable real y extending σ such that for all $n \geq |\sigma|$,*

$$\sum_{i=0}^{m} c_i F_i(y \upharpoonright n) \leq \sum_{i=0}^{m} c_i F_i(\sigma).$$

Proof. The proof is an analogue to that of Lemma 78. Simply at each stage choose the next bit (0 or 1) that ensures the sum will not exceed the original value. Because $F_0, F_1, \ldots, F_m$ are martingales this will always be possible. □

Proof of Lemma 80

Recall Lemma 80. *If a real is computable then it is not Martin-Löf random.*

Proof. Recall that a real x is computable if there is a computable sequence

$$q_0, q_1, \ldots$$

and a computable function f such that

$$|x - q_{f(n)}| < 2^{-n}.$$

We can define a Martin-Löf test via

$$U_k = (q_{f(k+1)} - 2^{-(k+1)}, q_{f(k+1)} + 2^{-(k+1)}),$$

such that for all k, $\mu(U_k) \leq 2^{-k}$ and $x \in \cap_k U_k$, so x is not Martin-Löf random. (In fact, this shows that x cannot even be Kurtz random as the sets U_k are finite and computable). □

Proof of Lemma 81

Recall Lemma 81. *For all i, let f_i be a function such that for all $x \in \mathrm{dom}(f_i)$,*

$$f_i(x) \in \mathbb{R}^+ \cup \{0\}.$$

Then

$$\limsup_n \sum_{i=0}^{\infty} f_i(n) < \infty \Rightarrow (\forall i) \limsup_n f_i(n) < \infty.$$

Proof. Proof by contrapositive. Suppose that there is some i such that $\limsup_n f_i(n) \to \infty$, that is,

$$(\forall k)(\exists m_k)(f_i(m_k) > k).$$

Then it follows that

$$(\forall k)(\exists m_k)(\sum_{i=0}^{\infty} f_i(m_k) > k),$$

so $\limsup_n \sum_{i=0}^{\infty} f_i(n) \to \infty$, as required. □

Proof of Lemma 108

Recall Lemma 108. *If we have an s-test $\{U_k : k \in \omega\}$ then there is an s-measurable machine M that maps strings of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$ where for each k,*

$$U_k = \{\sigma_{k,i} : i \in \omega\}.$$

Proof. Note firstly that for every k and i, $\sigma_{2k+2,i}$ is a prefix-free subset of U_{2k+2} all by itself, and thus we know that

$$2^{-s|\sigma_{2k+2,i}|} \leq 2^{-(2k+2)}.$$

Thus

$$s|\sigma_{2k+2,i}| \geq (2k+2)$$

and so $|\sigma_{2k+2,i}| \geq 2k+2$ as $s \leq 1$. From this we may conclude that

$$\begin{aligned} |\sigma_{2k+2,i}| - (k+1) &\geq k+1 \\ &\geq 0. \end{aligned}$$

Thus requiring the machine to map a string of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$ does not result in an absurdity for any $k, i \in \omega$. Secondly, we must check is that for all t, there are no more than 2^t strings of length t required for the domain of M. Because of the definition of M,

$$(\forall k)\big(k > t-1 \implies |\sigma_{2k+2,i}| - (k+1) > t\big)$$

and so we only need to concern ourselves with the sets U_{2k+2} where $k \leq t-1$.

In these cases, the strings which will have a requirement of length t will have length $|\sigma_{2k+2,i}| - (k+1) = t$; that is those strings in U_{2k+2} that have length $t+k+1$. Define $\#(S,t)$ to be the number of strings $\tau \in S$ such that

$|\tau| = t$ for any set S. Since the set of strings in U_{2k+2} that all have length $t+k+1$ must be a prefix-free set, it follows that

$$\begin{aligned}
\sum_{\substack{\sigma \in U_{2k+2} \\ |\sigma| = t+k+1}} 2^{-s(t+k+1)} &\leq 2^{-(2k+2)} \\
\#(U_{2k+2}, t+k+1)2^{-s(t+k+1)} &\leq 2^{-(2k+2)} \\
\#(U_{2k+2}, t+k+1) &\leq 2^{s(t+k+1)-(2k+2)} \\
&\leq 2^{(t+k+1)-(2k+2)} \\
&\leq 2^{t-k-1},
\end{aligned}$$

as $s \leq 1$. Then

$$\begin{aligned}
\#(\mathrm{dom}(M), t) &= \sum_{k<t} \#(U_{2k+2}, t+k+1) \\
&\leq \sum_{k<t} 2^{t-k-1} \\
&\leq 2^{t-1} \sum_{k<t} 2^{-k} \\
&\leq 2^{t}(1 - 2^{-(t-1)}) \\
&\leq 2^{t} - 2^{-1} \\
&< 2^{t}.
\end{aligned}$$

Thus there can never be more requirements needing a string of length t than there are strings of length t to be allocated.

It remains to show that M is an s-measurable machine. Take any prefix-free subset Q of $\mathrm{ra}(M)$. Now

$$Q \subseteq \cup_{k \in \omega} U_{2k+2},$$

so it is possible to define $Q_k = U_{2k+2} \cap Q$ (a prefix-free subset of U_{2k+2}) and to consider its pre-image $Q_k^* \subseteq \mathrm{dom}(M)$, as described in Definition 59.

$$\begin{aligned}
\mu_s(Q_k^*) &= \sum_{\sigma \in Q_k} 2^{-s|\sigma|} 2^{s(k+1)} \\
&= 2^{s(k+1)} \mu_s(Q_k) \\
&\leq 2^{s(k+1)} 2^{-2k-2} \\
&\leq 2^{-k-1},
\end{aligned}$$

because Q_k is a prefix-free subset of U_{2k+2}, and $s \leq 1$. Thus it follows that

$$\begin{aligned}
\mu_s(Q^*) &= \mu_s(\bigcup_{k \in \omega} Q_k^*) \\
&\leq \sum_{k \in \omega} \mu_s(Q_k^*) \\
&\leq \sum_{k \in \omega} 2^{-k-1} \\
&\leq 1.
\end{aligned}$$

Hence M is s-measurable. □

Bibliography

[1] AMBOS-SPIES, K., AND KUČERA, A. Randomness in computability theory. In *Computability Theory: Current Trends and Open Problems*, P. C. et. al., Ed., vol. 257. American Mathematical Society, 2000, pp. 1–14.

[2] BENNETT, C. Logical depth and physical complexity. In *The Universal Turing Machine. A Half-Century Survey*, R. Herken, Ed. Oxford University Press, Oxford, 1988, pp. 227–258.

[3] CALUDE, C., AND COLES, R. Program-size complexity of initial segments and domination reducibility. In *Jewels are Forever*, J. Karhumäki, H. Maurer, G. Paŭn, and G. Rozenberg, Eds. Springer-Verlag, Berlin, 1999, pp. 225–237.

[4] CALUDE, C., HERTLING, P., KHOUSSAINOV, B., AND WANG, Y. Recursively enumerable reals and Chaitin Omega numbers. *Theor. Comp. Sci.* (2001), 125–149.

[5] CALUDE, C., AND NIES, A. Chaitin Ω numbers and strong reducibilities. *Journal of Universal Computer Science 3*, 11 (1997), 1162–1166.

[6] CARATHEÓDORY, C. Über das lineare Maßvon Punktmengen - eine Verallgemeinerung des Längenbegriffs. *Nachrichten der K. Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-physikalische Klasse* (1914), 404–426.

[7] CHAITIN, G. On the length of programs for computing finite binary sequences. *Journal of the Association for Computing Machinery 13* (1966), 547–569.

[8] CHAITIN, G. On the length of programs for computing finite binary sequences: statistical considerations. *Journal of the Association for Computing Machinery 16* (1969), 145–159.

[9] CHAITIN, G. A theory of program size formal identical to information theory. *Journal of the Association for Computing Machinery 22* (1975), 329–340.

[10] CHAITIN, G. Algorithmic information theory. *IBM Journal of Research and Development* (1977), 350–359.

[11] CHURCH, A. On the concept of a random sequence. *Bull. AMS* (1940).

[12] DOWNEY, R., AND GRIFFITHS, E. Schnorr randomness (extended abstract). In *CCA 2002 Computability and Complexity in Analysis* (Amsterdam, 2002), V. Brattka, M. Schröder, and K. Weihrauch, Eds., vol. 66 of *Electronic Notes in Theoretical Computer Science*, Elsevier. 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002.

[13] DOWNEY, R., GRIFFITHS, E., AND LAFORTE, G. On computable randomness. in preparation.

[14] DOWNEY, R., GRIFFITHS, E., AND REID, S. Kurtz randomness. Submitted, 2003.

[15] DOWNEY, R., AND HIRSCHFELDT, D. Algorithmic randomness and complexity. Springer-Verlag Monographs in Computer Science, in preparation.

[16] DOWNEY, R., HIRSCHFELDT, D., AND LAFORTE, G. Randomness and reducibility (extended abstract). In *Mathematical Foundations of*

Computer Science, J. Sgall, A. Pultr, and P. Kolman, Eds., no. 2136 in Lecture Notes in Computer Scinece. Springer, 2001, pp. 316–327. To appear in *Journal of Computing and System Sciences*.

[17] EDGAR, G. *Measure, topology and fractal geometry.* Springer-Verlag, 1990.

[18] EDGAR, G. *Integral, probability and fractal measures*. Springer, 1998.

[19] FALCONER, K. *Fractal geometry: mathematical foundations and applications*. Wiley, Chichester, New York, 1990.

[20] FORTNOW, L. My computational complexity web log. http://fortnow.com/lance/complog/archive/2002_09_29_archive.html (accessed September 2003).

[21] FRIEDBERG, R., AND ROGERS, H. Reducibility and completeness for sets of integers. *Z. Math. Logik Grundlag. Math. 5* (1959), 117–125.

[22] HAUSDORFF, F. Dimension und äußeres maß. *Mathematische Annalen 79* (1918), 157–79.

[23] KAUTZ, S. *Degrees of Random Sets*. PhD thesis, Cornell University, 1991.

[24] KOLMOGOROV, A. Three approaches to the quantitative definition of information. *Problems of Information and Transmission 1*, 1 (1965), 1–7.

[25] KRAFT, L. A device for quantizing, grouping, and coding amplitude modulated pulses. MSc. thesis, Dept. of Electrical Engineering, MIT, Cambridge, Mass., 1949.

[26] KUČERA, A. Measure, Π^0_1-classes and complete extensions of PA. In *Recursion Theory Week*, H. Ebbinghaus, G. Müller, and G. Sacks, Eds., no. 1141 in Lecture Notes in Mathematics. Springer-Verlag, Berlin, 1985, pp. 245–259.

[27] KUČERA, A., AND SLAMAN, T. Randomness and recursive enumerability. *SIAM Journal on Computing 31*, 1 (2001), 199–211.

[28] KURTZ, S. *Randomness and Genericity in the Degrees of Unsolvability*. PhD thesis, University of Illinois at Urbana-Champaign, 1981.

[29] LEBESGUE, H. *Leçons sur l'intégration et la Recherché des Fonctions Primitives*. Gauthier-Villars, 1904.

[30] LEVIN, L. On the notion of a random sequence. *Soviet Mathematics Doklady 14* (1973), 1413–1416.

[31] LI, M., AND VITANYI, P. *An Introduction to Kolmogorov Complexity and its Applications*. 2nd ed., Springer-Verlag, New York, 1997.

[32] LUTZ, J. The dimensions of individual strings and sequences. Information and Computation, to appear.

[33] LUTZ, J. Almost everywhere high nonuniform complexity. *J. Comp. System Sci. 44* (1992), 220–258.

[34] LUTZ, J. Dimension in complexity classes. In *IEEE Conference on Computational Complexity* (2000), pp. 158–169.

[35] MARTIN-LÖF, P. The definition of random sequences. *Information and Control 9* (1966), 602–619.

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

[37] ROGERS, C. *Hausdorff measures*, 2 ed. Cambridge University Press, New York, 1998.

[38] ROGERS, H. *Theory of recursive functions and effective computability*. McGraw-Hill series in higher mathematics. McGraw-Hill, New York, 1967.

[39] SCHNORR, C. A unified approach to the definition of random sequences. *Mathematical System Theory 5* (1971), 246–258.

[40] SCHNORR, C. *Zufälligkeit und Wahrscheinlichkeit*, vol. 218 of *Lecture Notes in Math.* Springer, Berlin, 1971.

[41] SOARE, R. Recursion theorem and Dedekind cuts. *Transactions of the American Mathematical Society 140* (1969), 271–294.

[42] SOARE, R. *Recursively Enumerable Sets and Degrees*. Perspectives in Mathematical Logic. Springer-Verlag New York, Inc., New York, NY, 1987.

[43] SOLOMONOFF, R. A preliminary report on a general theory of inductive inference. Tech. Rep. ZTB-138, Zator Company, Cambridge, Mass., November 1960.

[44] SOLOMONOFF, R. A formal theory of inductive inference, part 1 and part 2. *Information and Control 7* (1964), 1–22,224–254.

[45] SOLOVAY, R. Draft of a paper (or series of papers) on Chaitin's work. Unpublished manuscript. IBM Thomas J Watson Research Center, Yorktown Heights, New York, May 1975.

[46] SPIEGEL, M. *Schaum's outline of theorem and problems of advanced Calculus*. Schaum's outline series. McGraw-Hill, New York, 1974. Previous ed: New York: Schaum, 1963.

[47] TADAKI, K. A generalization of Chaitin's halting probability Ω and halting self-similar sets. *Hokkaido Mathematical Journal 31*, 1 (February 2002), 219–253.

[48] TERWIJN, S. Complexity and randomness. research report, University of Auckland, Auckland, New Zealand, March 2003. Notes for a course. Can be downloaded from http://www.logic.at/people/terwijn/publications.html.

[49] TURING, A. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society, Series 2 42* (1936), 230–265.

[50] VAN LAMBALGEN, M. *Random Sequences*. PhD thesis, University of Amsterdam, 1987.

[51] VILLE, J. *Etude critique de la notion de collectif*. Fauthiers-Villars, 1939.

[52] VON MISES, R. Grundlagen der Wahrscheinlichkeitsrechnung. *Mathemat. Zeitsch. 5* (1919), 52–99.

[53] WALD, A. Die widerspruchsfreiheit des kollektivbegriffs der wahrscheinlichkeitsrechnung. *Ergebnisse eines mathematischen Kolloquiums 8* (1937), 38–72.

[54] WANG, Y. *Randomness and Complexity*. PhD thesis, University of Heidelberg, 1996.

[55] WANG, Y. A separation of two randomness concepts. *Information Processing Letters 69*, 3 (1999), 115–118.