Algorithmic Randomness and Complexity

Rod Downey
School of Mathematical and Computing Sciences
Victoria University
PO Box 600
Wellington
New Zealand

Denis R. Hirschfeldt
Department of Mathematics
University of Chicago
Chicago IL 60637
U.S.A.

July 28, 2007

Contents

1 Preface

Though we did not know it at the time, this book's genesis began with the arrival of Cris Calude in New Zealand. Cris has always had an intense interest in algorithmic information theory. The event that led to much of the recent research presented here was a seemingly innocuous question articulated by Cris. This question goes back to Solovay's legendary manuscript [284], and Downey learnt of it during a visit Richard Coles made to Victoria University in early 2000. Coles was then a Postdoctoral Fellow with Calude at Auckland University. In effect, the question was whether the Solovay degrees of left-computably enumerable reals are dense.

At the time, neither of us knew much about Kolmogorov complexity, but we had a distinct interest in it after Lance Fortnow's illuminating lectures [110] at Kaikoura[1] in January 2000. After thinking about Calude's question for a while, and eventually solving it together with André Nies [79], we began to realize that there was a huge and remarkably fascinating area of research, whose potential was largely untapped, lying at the intersection of computability theory and the theory of algorithmic randomness.

[1]Kaikoura was the setting for a wonderful meeting on computational complexity. There is a set of lecture notes [76] resulting from this meeting, aimed at graduate students. More information and some notes can also be found through the web site `http://www.mcs.vuw.ac.nz`. Kaikoura is on the east coast of the South Island of New Zealand, and is famous for its beauty and for tourist activities such as whale watching and dolphin, seal, and shark swimming. The name "Kaikoura" is a Maori word meaning "eat crayfish", which is a fine piece of advice.

We also found that, while there is a truly classic text about *general* Kolmogorov Complexity, namely Li and Vitanyi [185], most of the questions we were interested in either were open, were exercises in Li and Vitanyi with difficulty ratings of about 40-something (out of 50), or necessitated an archeological dig into the depths of a literature with few standards in notation[2] and terminology, littered with relentless re-discovery of theorems and a significant amount of unpublished material. Particularly noteworthy amongst the unpublished material was the aforementioned set of notes by Solovay [284], which contained absolutely fundamental results about Kolmogorov complexity in general, and about initial segment complexities of reals in particular. As our interests broadened, we also became aware of seminal results from Stuart Kurtz' PhD Dissertation [165], which, like Solovay's results, seemed unlikely to ever be published in a journal. Meanwhile, a large number of other authors started to make great strides in our understanding of this area of research.

Thus, we decided to try to organize the material on the algorithmic randomness and computability theory into a coherent book. We were especially thankful for Solovay's permission to present, for the first time, the details from his unpublished notes.[3] We were encouraged by the support of Springer-Verlag in this enterprise.

Naturally, this project has conformed to Hofstadter's Law: Things will take longer than one thinks they will, even if one takes into account Hofstadter's Law. Some of the reason for this delay is that gifted researchers such as Evan Griffiths, John Hitchcock, Jack Lutz, Joe Miller, Wolfgang Merkle, Elvira Mayordomo, An. A. Muchnik, Andre Nies, Jan Reimann, Frank Stephan, Sebastiaan Terwijn, Liang Yu, and others continued to relentlessly prove theorems that made it necessary to re-write large sections of the book.

This is *not* a basic text on Kolmogorov complexity. We concentrate on the Kolmogorov complexity of reals (i.e., infinite sequences) and only cover as much as we need on the complexity of finite strings. There is quite a lot of background material in computability theory needed for some of the more sophisticated proofs we present. For this reason we do give a full but, by necessity, rapid refresher course in basic "advanced" computability theory. This material should not be read from beginning to end. Rather, the reader should dip into Chapter 5 as the need arises. For a fuller introduction, we refer the reader to the classic texts of Rogers [253], Soare [280], and Odifreddi [233, 234].

---

[2]We hope to help standardize notation. In particular, we have fixed upon the notation for Kolmogorov complexity used by Li and Vitanyi: C for plain Kolmogorov complexity and K for the prefix-free Kolmogorov complexity.

[3]Of course, Li and Vitanyi [185] used Solovay's notes extensively, mostly in the exercises and for quoting results.

In the introduction, we will try to motivate the material to follow, but we will mostly avoid historical comments. The history of the evolution of Kolmogorov complexity and related topics can make certain people rather agitated, and we feel neither competent nor masochistic enough to enter the fray. What seems clear is that, at some stage, time was ripe for the evolution of the ideas needed for Kolmogorov complexity. There is no doubt that many of the basic ideas were implicit in Solomonoff [282], and that many of the fundamental results are due to Kolmogorov [151]. The measure theoretical approach was pioneered by Martin-Löf [198]. Many fundamental results were established by Levin [177, 178, 332] and by Schnorr [265, 266], particularly those using the measure of domains to avoid the problems of plain complexity in addressing the initial segment complexity of reals. It is but a short step from there to prefix-free complexity (and discrete semimeasures), first articulated by by Levin [178] and Chaitin [43, 44]. Schnorr's penetrating ideas, only some of which are available in their original form in English (see [264, 265]), are behind much modern work in computational complexity, such as Lutz' approach in [190, 192, 193], which is based on martingales and orders, though as has often been the case in this area, Lutz developed his material without being too aware of Schnorr's work. Lutz was apparently the first to explicitly connect orders and Hausdorff dimension. From yet another perspective, martingales, or rather supermartingales, are essentially the same as continuous semimeasures, and again we see the penetrating insight of Levin [332].

We are particularly pleased to present the results of Kurtz and Solovay, and the seminal material of Antonin Kučera in this book. Kučera was a real pioneer in connecting computability and randomness, and we believe that it is only recently that the community has really appreciated his deep intuition.

2
Acknowledgements

The biggest acknowledgement must be to the Marsden Fund of New Zealand. There is absolutely no question that this work would not have happened without their generous support.

Richard Coles, Denis Hirschfeldt, Geoff LaForte, and Joe Miller have all been Downey's Marsden Postdoctoral fellows. Nies and Downey also have received direct Marsden support for the period, and the first interest in the area was stimulated by the Kaikoura 2000 talk of Lance Fortnow the conference being almost completely supported by the Marsden Foundation via the New Zealand Mathematical Research Institute. This Institute is a virtual one, and was the brainchild of Vaughan Jones. After receiving the Fields Medal, among other things, Vaughan has devoted his substantial influence to bettering New Zealand Mathematics. The visionary NZMRI was founded with this worthy goal in mind. The NZMRI was conceived to run annually a workshop at picturesque location devoted to specific areas of mathematics. These would involve lecture series by overseas experts aimed at the graduate student, and fully funded for New Zealand attendees. The NZMRI is chaired by Vaughan, has as its other directors Downey, and the uniformly excellent Marston Conder, David Gauld, Gaven Martin from Auckland University.

Liang Yu and Downey were also supported by the New Zealand New Zealand Institute for Mathematics and its Applications, a recent CoRE (Centre of Research Excellence), which grew from the NZMRI.

As a postdoctoral fellow, Guohua Wu was supported by the New Zealand Foundation for Research Science and Technology, having previously been supported by the Marsden Foundation as Downey's PhD student. Also

Stephanie Reid was similarly supported by the Marsden Foundation for her MSc thesis.

At other times, Downey has been supported by the NSF whilst visiting Notre Dame, and the University of Chicago. Hirschfeldt also acknowledges the generous support of the NSF.

We wish to thank our colleagues for corrections and the generous sharing of their results and ideas. These include Eric Allender, Klaus Ambos-Spies, Azat Arslanov, Veronica Becher, Cris Calude, Richard Coles, Peter Cholak, Chi Tat Chong, Barbara Csima, Ding Decheng, Jean-Paul Delahaye, Lance Fortnow, Santiago Fugiera, Evan Griffiths, John Hitchcock, Carl Jockusch, Bjorn Kjos-Hanssen, Antonin Kučera, Geoff LaForte, Jack Lutz, Wolfgang Merkle, Joe Miller, Antonio Montalbán, Andre Nies, Jan Reimann, Alex Raichev, Alexander Shen, Richard Shore, Ted Slaman, Frank Stephan, Paul Vitanyi, Guohua Wu, Yang Yue, Yu Liang, and Xizhong Zheng.

Thanks indeed to Springer-Verlag and particularly to ??? for allowing us to run overtime.

Downey dedicates this book to his wife Kristin, and Hirschfeldt to ???

3
Introduction

How random is a real? Given two reals, which is more random? How should we even try to quantify these questions, and how do various choices of measurement relate? Once we have reasonable measuring devices, and, using these devices, we divide the reals into equivalence classes of the same "degree of randomness" what do the resulting structures look like? Once we measure the level of randomness how does the level of randomness relate to classical measures of complexity Turing degrees of unsolvability? Should it be the case that high levels of randomness mean high levels of complexity in terms of computational power, or low levels of complexity? Conversely should the structures of computability such as the degrees and the computably enumerable sets have anything to say about randomness for reals?

These were the kinds of questions motivating the research which is represented in this book. While some fundamental questions remain open, nevertheless we now have a reasonable insight into many of these questions, and the resulting body of work is both beautiful and has a number of rather deep theorems.

We all see a sequence like $11111111111\ldots$ and another like $101101011101010111100001010100010111\ldots$ and feel that the second (obtained by the first author by a coin toss) is more random than the first. However, in terms of measure theory, each is equally likely. It is a deep and fundamental question to try to understand why some sequence might be "random" and another might be "lawful," and, moreover, how to quantify our intuition into a meaningful mathematical notion.

The roots of the study of algorithmic randomness -questions like these- go back to the work of von Mises at the dawn of the 20th century. In a remarkable paper [317], von Mises argued that a a random real should be what we would now call *stochastic* in the sense that it ought to obey all "reasonable" statistical tests. For instance, the number of zeroes in the sequence $\leqslant n$ asymptotically be the same as the number of ones. Actually he expressed his ideas about *Kollektivs* in terms of "acceptable selection rules." He observed that any countable collection could be beaten, but at the time it was unclear what types of selection rules should be admissible. There seemed to him no canonical choice.

Later, with the development of computability/recursive function theory, it seemed that the notion of *algorithmic* randomness is intimately tied to the notion of a computable function. This was argued by Church [52] and others. Church made the connection with the theory of computability by suggesting that one should take all *computable* stochastic properties. A blow to this program was made by Ville [312], who pointed out that no matter what admissible rules were chosen there would be reals which were random relative to the chosen rules, *yet* had properties that would demonstrate that the real was not random.

In a sweeping generalization, Martin-Löf [198] noted that selection rules, and similar stochastic approaches are special kinds of measure zero sets, and the this approach culminated with Martin-Löf's definition of randomness as reals that avoided certain effectively presented measure 0 sets.

Exactly which choice of "effectively presented measure 0 sets" are appropriate now becomes the issue and manipulating the acceptable kinds of measure 0 sets allows one to calibrate randomness in a natural way.

In this book we will consider a number of such variations, n-randomness, arithmetical randomness, s-randomness (associated with Hausdorff dimension), Schnorr randomness, Kurtz randomness and the like.

There are subtle and deep questions about the measure-theoretical approach and its relationship with the original stochastic approach. We will discuss these relationships later in the book.

The evolution and clarification of many these notions is carefully discussed in the PhD Thesis of Michael van Lambalgen [314].

What we call the measure-theoretical approach is only one of the three basic approaches to algorithmic randomness. They are in terms of *unpredictability*, *typicalness*, and *incompressibility.* Strangely, the last was the first to be adequately addressed. Kolmogorov [151], gave the first basic results on what we now call Kolmogorov complexity, though these results were foreshadowed by Ray Solomonoff [282]. Roughly, the idea is that a string should be random only if it cannot be compressed by some program.

This leads to the now "standard" notion of Kolmogorov complexity of a *string*, σ, namely the length of shortest program τ such that $U(\tau) = \sigma$. Then a string is random iff its Kolmogorov complexity is the same size as its length. Kolmogorov proved basic results such as this is well-defined notion since, up to a constant, the choice of universal machine does not matter.

However, when we turn to reals, this compressibility paradigm has a few problems cause by the fact that τ gives, essentially $|\tau|+\log|\tau|$ many bits of information, and Martin-Löf showed that this fact can be exploited to show that is we guess "a *real* is random iff all its segments are incompressible" then there are *no* random reals. What is needed is a Kolmogorov complexity that avoids this extra $\log n$ many bits of information. This was invented first by Levin [177, 178] and Schnorr [265, 266] developed by Gács [114] and also discovered slightly later by Chaitin [42], Schnorr used "discrete" monotone machines (process complexity), Levin using either monotone or prefix-free Turing machines, and Chaitin used prefix-free machines.

Using these ideas, it is possible to show that Martin-Löf's original definition is the same as one based on compressibility. The same is mostly true for other notions of randomness based on measure theory: there are corresponding machine definitions.

The first few chapters of this book look at such considerations. The first one is a brief run down of the basic measure theory we will use. The second is a very quick recap of the important results and techniques of classical computability we will use in the book. The reader is warned that the novice might find some of this chapter relatively tough going, and is really meant as more of *reference* for the rest of the book, and might even only be *scanned* on a first reading.

The next few chapters develop the basics of Kolmogorov complexity and the measure-theoretical approach, and their connections. We include proofs of the basic results such as the counting theorems, Symmetry of Information and the Coding Theorem. We will also include some fundamental results on counting various finite sets associated with C and K.

Next, in Chapter 7, we present for the first time in a published form, the details of Solovay's remarkable results relating prefix-free Kolmogorov complexity $K(\sigma)$ and $C(\sigma)$, plain complexity. For instance, it is proven that

$$K(\sigma) = C(\sigma) + C^{(2)}(\sigma) + \mathcal{O}(C^{(3)}(\sigma)),$$

and this cannot be improved to have a $\mathcal{O}(C^{(4)}(\sigma))$ term. These results are presented here with Solovay's permission. We use these results to obtain a lovely result of An. A. Muchnik that for all c there exist strings σ and τ with $C(\sigma) > C(\tau) + c$, yet $K(\tau) > K(\sigma) + c$, plus its improvement by Joe Miller, who showed, for example, that it is additionally possible to have $|\sigma| = |\tau|$.

In Chapter 9, we will introduce ideas in the study of algorithmic information theory for *reals*. In this chapter, we will stress the approaches of compression, and of measure, leaving the martingale approach till Chapter 10. We will prove Schnorr's theorem that a real α is Martin-Löf random iff its initial segment complexity is essentially n for segments of length n.

We also include some fascinating new theorems of Miller-Yu-Nies-Stephan-Terwijn classifying the possible initial segment complexities of random reals, and relating 2-randomness to plain complexity. (A real is 2-random iff its initial segment infinitely often hits n with its plain complexity.) Here we also introduce the basic notions of higher level randomness such as n-randomness, and prove basic lemmas. We also look at randomness relative to other measures, and prove Demuth's result relating randomness and truth table reducibility.

One of the basic structures of computability theory is the collection of computably enumerable sets. By analogy, one of the basic structures of the study of effective randomness is the *left computable* or, *computably enumerable reals*, which are the limits of effective increasing sequences of rationals, and are the measures of *domains* or prefix-free machines. A classic example is Chaitin's Ω which is the measure of the domain of a universal prefix free machine. We will devote one chapter to basic properties of this class of reals and will continue to study them as a core class for randomness considerations. Later we will include a proof that almost all reals are halting probabilities relative to some oracle.

The next couple of chapters are devoted to the theory of the calibration of reals using initial segment measures or relative randomness. For instance, we will study Solovay reducibility where $\alpha \leqslant_S \beta$ means that there is a constant C and a partial computable function $g : \mathbb{Q} \mapsto \mathbb{Q}$, such that for all rationals $q < \beta$ $g(q)\downarrow$ and $C(\beta - q) > \alpha - g(q)$. We prove such gems as the Kučera-Slaman Theorem that a computably enumerable real is complete under $\leqslant_S$ iff it is random. Many other measures are discussed and the degree structures analyzed.

In Chapter 10 we look at the martingale approach, and other calibrations of randomness, this time based around variations of the measure approach. These include Schnorr, Kurtz and computable randomness. We give the Downey-Griffith machine characterization of Schnorr randomness, and see how the concepts relate to Turing degrees.

In Chapter 11, we examine the relationship between randomness and the Turing degrees, as pioneered by Kurtz and Kučera. We prove the remarkable Kučera-Gács Theorem that any set can be coded into a random one, and that all degrees above $\mathbf{0}'$ are random. We will look at the results of Stephan classifying randomness in terms of PA degrees, and Stillwell's Theorem that the "almost all" theory of degrees is decidable. Additionally, we will look at Kutrz's results such as the proof that almost every real is computably enumerable in and above one of lesser degree. Again, Kurtz's marerial is only currently available with access to his PhD Thesis.

This theme is continued into the next chapter, where we study the global structure of the Kolmogorov degrees. It is shown that there are $2^{\aleph_0}$ many K-degrees, and almost all of them are K-incomparible. Most of these results are due to Miller and Yu, though the first such result was due to Solovay [284], who showed that $\Omega^{\emptyset'} \not\leqslant_K \Omega$. We give Solovay's proofs, then turn to the more general Miller-Yu methods. They filter through yet another reducibility called von Lambalgen reducibility, which turns out to to be an excellent tool for this analysis in its own right. For instance, using it, we can easily show that if $A \leqslant_T B$, and both are random, then if B is n-random, so too is A.

In Chapter 18 we look at Ω as an operator on Cantor Space. At some early stage there seemed some hope that it might be degree invariant, and hence providing a counterexample to a longstanding conjecture of Martin that the only degree invariant operators on the degrees were iterates of the jumps. However, it is shown that, amongst other things, there are sets $A =^* B$ with Ω^A and Ω^B relatively random (and hence Turing incomparible). It is shown that almost every set is actually an Ω^X for some set X. Thus whislt it might seem that Ω is a peculiarity of the computably enumerable reals, it is in fact central to the study of randomness. Many other remarkable results can be found here, emphasizing the fact that our understanding of operators that are computably enumerable operators, but *not* computably enumerable *in and above*, is extremely limited.

The next chapter is concerned with yet another calibration of randomness. This time via effectivizations of the classical refinement of measure zero: Hausdorff and other dimensions. We study the effectivizations of Ambos-Spies, Lutz, Staiger and others. There are some extremely attractive results relating Hausdorff dimension to Turing degrees. We also look at box counting and packing dimension, which has relevance later in the Chapter 19. A key result here is that packing dimension again has a characterization in terms of initial segment complexity. The chapter culminates in many ways with Lutz' theory of what are called *termgales* allowing us to assign a dimension to individual strings. This assignation allows for a result relating dimension to Kolmogorov complexity which is a direct analogy of the Coding Theorem.

The next chapter is the longest in the book and has some of the deepest material. It is devoted to the amazing phenomenon of K-*triviality.* Whilst Chaitin had proven that if, for all n, $C(A \upharpoonright n) \leqslant C(n) + \mathcal{O}(1)$, then A is computable, Solovay [284] had shown that there were noncomputable sets B with $K(B \upharpoonright n) \leqslant K(n) + \mathcal{O}(1)$ for all n. These sets are now called the K-trivial sets. They have astonishing properties. For each constant $\mathcal{O}(1)$ there are only finitely many corresponding B's. They are "natural" solutions to Post's problem. This fact was first established by Downey, Hirschfeldt, Nies and Stephan, [81]. Nies (and Hirschfeldt for some results) then proved that they were all superlow, they formed a natural ideal in the Turing degrees, and each K-trivial was below a computable enumerable K-trivial. Nies also showed that this class coincided with other classes such as the (super-) low for K reals.

In chapter 16, we will look at lowness for other randomness notions such as the Schnorr low reals, proving the Terwijn-Zambella [304] Theorem classifying such reals in terms of tracing functions, and Nies' Theorem that there are no noncomputable sets which are low for computable randomness.

The penultimate chapter will be devoted to randomness considerations relative to computably enumerable *sets*. We prove things like Kummer's Theorem that the Turing degree containing c.e. sets of maximal Kolmogorov complexity are exactly the array noncomputable ones. Also we will prove the results of Kummer and An. A. Muchnik about the truth table completeness of the overgraphs: the collections of nonrandom strings, and the Downey-Reimann material about the computable dimensions of computably enumerable sets.

The last chapter looks at a modern re-examination of the original idea of *selection* going back to von Mises. We give material relating stochasticity with randomness, and the recent results on nonmonotonic selection.

The main idea in this book is that we wish to understand how to calibrate randomness for reals, and how that relates to traditional measures of complexity such as relative computational power. Most of the material is from the last few years, where there has been an explosion of wonderful ideas in the area. The material above is our version of what we see as many of the highlights. We apologize in advance to those whose work has been neglected by our ignorance.

Part I

Background

4 Preliminaries

4.1 Notation and Conventions

For functions $f, g, h : \mathbb{N} \to \mathbb{N}$, we write:

(i) $f(n) \leqslant g(n) + O(h(n))$ to mean that there is a constant c such that $f(n) \leqslant g(n) + ch(n)$ for all n,

(ii) $f(n) = g(n) \pm O(h(n))$ to mean that $f(n) \leqslant g(n) + O(h(n))$ and $g(n) \leqslant f(n) + O(h(n))$, and

(iii) $f(n) \leqslant g(n) + o(h(n))$ to mean that $\lim_n \frac{f(n)-g(n)}{h(n)} = 0$.

We say that f and g are *assymptotically equal* if $\lim_n \frac{f(n)}{g(n)} = 1$, and write $f \sim g$ to mean that $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

Note that, in particular, $f(n) = O(h(n))$ means that there is a constant c such that $f(n) \leqslant ch(n)$ for all n. Similarly, $f(n) = o(h(n))$ means that $\lim_n \frac{f(n)}{h(n)} = 0$.

Sometimes, when we write an expression such as $f(n) \leqslant g(n) + O(h(n))$, there is an extra additive constant on the right hand side. Such a constant is of course absorbed into the $O(h(n))$ term, but only if $h(n) > 0$. So we make it a convention that, even if $h(n) = 0$, the $O(h(n))$ term is at least $O(1)$, and hence still absorbs the constant.

We also use this notation for other kinds of functions, such as functions from $2^{<\omega}$ to $\mathbb{N}$, in an analogous way.

When we write $\log n$, we mean the base 2 logarithm of n, rounded up to the nearest integer. By convention, $\log 0 = 0$.

We identify sets with their characteristic functions. For instance, $A(n) = 1$ and $n \in A$ mean the same thing.

For a function f, we write $f^{(k)}$ to mean f composed with itself k many times. Thus, for example $\log^{(2)} n$ means $\log \log n$. By convention, $f^{(0)}$ is the identity function.

For each $n > 1$ we fix an injective and surjective computable function $\mathbb{N}^n \to \mathbb{N}$ and denote its value on the arguments $a_1, \dots, a_n$ by $\langle a_1, \dots, a_n \rangle$.

For $A \in 2^\omega$, let $A \upharpoonright n$ be the string consisting of the first n bits of A.

We will use standard logical notation, including the quantifiers $\exists^\infty$ for "there exist infinitely many" and $\forall^\infty$ for "for all but finitely many".

4.2 Basics from Measure Theory

We will assume that the reader is familiar with the basic concepts of measure theory as given in initial segments of classical texts such as Oxtoby [235], but we review a few important facts in this chapter.

For simplicity of presentation, we study the Cantor space 2^ω of infinite binary sequences. The (sub-)basic open sets in this space are

$$[\sigma] := \{\sigma X : X \in 2^\omega\}$$

for $\sigma \in 2^{<\omega}$. In this topology these sets are clopen, and the clopen sets are finite unions of such basic open sets. Although the real unit interval is not homeomorphic to 2^ω with this topology, it is well known that the two spaces are isomorphic in the measure theoretical sense.

The (Lebesgue) measure of a basic open set $[\sigma]$ is $\mu([\sigma]) := 2^{-|\sigma|}$. A sequence of basic open sets $\{[\sigma] : \sigma \in A\}$ is said to *cover* a set $C \subseteq 2^\omega$ if $C \subseteq \bigcup_{\sigma \in A} [\sigma]$. The *outer measure* of C is

$$\mu^*(C) = \inf\{\textstyle\sum_n 2^{-|\sigma_n|} : \{[\sigma_n] : n \in \mathbb{N}\} \text{ covers } C\}.$$

The *inner measure* of C is $\mu_*(C) = 1 - \mu^*(\overline{C})$, where $\overline{C}$ is the complement of C. If C is measurable, then $\mu^*(C) = \mu_*(C)$, and we refer to this quantity as the *measure* $\mu(C)$ of C. *Null sets* are those that have measure 0. A set C has measure zero iff there is a sequence of open covers of C whose measure goes zero, that is, a collection $\{V_n : n \in \mathbb{N}\}$ of open sets such that

(i) $\mu(V_n) \leqslant 2^{-n}$ and

(ii) $C \subseteq \bigcap_n V_n$.

We will later effectivize this notion when we look at Martin-Löf randomness in Chapter 9. We will also effectivize the following fact, known as the (First) Borel-Cantelli Lemma, when we look at Chaitin-Schnorr randomness and Solovay randomness.

Theorem 4.2.1 (Borel-Cantelli Lemma). *Let $\{C_i : i \in \mathbb{N}\}$ be a sequence of Lebesgue measurable sets such that $\sum_i \mu(C_i) < \infty$. Then*

$$\mu(\{X : \exists^\infty n\,(X \in C_n)\}) = 0.$$

Proof. Let $D = \{X : \exists^\infty n\,(X \in C_n)\}$. Choose a sequence $\{n_i : i \in \mathbb{N}\}$ such that $\sum_{j \geqslant n_i} \mu(C_j) \leqslant 2^{-i}$. For each i, we have $D \subseteq \bigcup_{j \geqslant n_i} C_j$, so D is a null set. □

The following (simplified version of a) classical fact from measure theory, known as the Lebesgue Density Theorem, will be used often in this book. It implies that for every set A of positive measure there are basic open sets within which the measure of A appears to be arbitarily close to 1. As we will see, this is the critical fact allowing for the existence of 0-1 laws in degree theory. Our proof follows the one in Terwijn's dissertation [306].

Definition 4.2.2. A measurable set A has *density d at X* if

$$\lim_n 2^n \mu(A \cap [X \upharpoonright n]) = d.$$

Let $\Xi(A) = \{X : A \text{ has density 1 at } X\}$.

Theorem 4.2.3 (Lebesgue Density Theorem). *If A is measurable then so is $\Xi(A)$. Furthermore, the measure of the symmetric difference of A and $\Xi(A)$ is zero, so $\mu(\Xi(A)) = \mu(A)$.*

Proof. It suffices to show that $A - \Xi(A)$ is a null set, since $\Xi(A) - A \subseteq \overline{A} - \Xi(\overline{A})$ and $\overline{A}$ is measurable. For $\varepsilon \in \mathbb{Q}^+$, let

$$B_\varepsilon = \{X \in A : \liminf_n 2^n \mu(A \cap [X \upharpoonright n]) < 1 - \varepsilon\}.$$

Then $A - \Xi(A) = \bigcup_\varepsilon B_\varepsilon$, so it suffices to show that each B_ε is null.

Suppose for a contradiction that there is an ε such that B_ε is not null, that is, $\mu^*(B_\varepsilon) > 0$. It is easy to see that $\mu^*(B_\varepsilon) = \inf\{\mu(U) : B \subseteq U \wedge U \text{ open}\}$, so there is an open set $U \supseteq B_\varepsilon$ such that $(1-\varepsilon)\mu(U) < \mu^*(B_\varepsilon)$. Let

$$I = \{\sigma : [\sigma] \subseteq U \ \wedge\ \mu(A \cap [\sigma]) < (1-\varepsilon)2^{-|\sigma|}\}.$$

Then the following facts hold.

(i) If $X \in B_\varepsilon$ then I contains $X \upharpoonright n$ for infinitely many n.

(ii) If $\{\sigma_i : i \in \mathbb{N}\}$ is a sequence of elements of I such that $[\sigma_i] \cap [\sigma_j] = \emptyset$ for $i \neq j$, then $\mu^*(B_\varepsilon - \bigcup_i [\sigma_i]) > 0$.

Fact (i) holds because U is open and contains B_ε. Fact (ii) holds because

$$\mu^*(B_\varepsilon \cap \textstyle\bigcup_i [\sigma_i]) \leqslant \mu(A \cap \bigcup_i [\sigma_i]) = \sum_i \mu(A \cap [\sigma_i]) < \sum_i (1-\varepsilon)2^{-|\sigma_i|} \leqslant (1-\varepsilon).$$

Construct a sequence $\{\sigma_i : i \in \mathbb{N}\}$ as follows. Let σ_0 be any element of I. Given σ_i for $i \leqslant n$, let $I_n = \{\sigma \in I : [\sigma] \cap [\sigma_i] = \emptyset \text{ for all } i \leqslant n\}$. By (i) and (ii), I_n is infinite. Let $d_n = \sup\{2^{-|\sigma|} : \sigma \in I_n\}$ and let σ_{n+1} be an element of I_n such that $2^{-|\sigma_{n+1}|} > \frac{d_n}{2}$.

Let $X \in B_\varepsilon - \bigcup_i [\sigma_i]$, which exists by (ii). By (i), there is a $\tau \in I$ with $X \in [\tau]$. If $[\tau]$ were disjoint from every $[\sigma_n]$, and hence contained in every I_n, then we would have $2^{-|\tau|} \leqslant d_n < 2^{-|\sigma_n|+1}$ for every n, contradicting the fact that $\mu(\bigcup_n [\sigma_n]) \leqslant 1$. So there is a least n such that $[\tau] \cap [\sigma_n] \neq \emptyset$. Note that this means that either $\tau \preccurlyeq \sigma_n$ or $\sigma_n \preccurlyeq \tau$. The minimality of n implies that $2^{-|\tau|} \leqslant d_{n-1} < 2^{-|\sigma_n|+1}$, so $|\tau| \geqslant |\sigma_n|$. Thus $\sigma_n \preccurlyeq \tau$, and hence $[\tau] \subseteq [\sigma_n]$. But then $X \in [\sigma_n]$, contradicting the choice of X. □

A *tailset* is a set $A \subseteq 2^\omega$ such that for all $\sigma \in 2^{<\omega}$ and $X \in 2^\omega$, if $\sigma X \in A$ then $\tau X \in A$ for all τ with $|\tau| = |\sigma|$. An important corollary to the Lebesgue Density Theorem is the following theorem of Kolmogorov.

Theorem 4.2.4 (Kolmogorov's 0-1 Law)**.** *If A is a tailset, then either $\mu(A) = 0$ or $\mu(A) = 1$.*

Proof. Again we follow the slick proof in Terwijn's thesis. Suppose that $\mu(A) > 0$. By Theorem 4.2.3, choose $X \in A$ such that A has density 1 at X. Let $\varepsilon > 0$. Choose n sufficiently large so that $2^n \mu(A \cap [X \upharpoonright n]) > 1 - \varepsilon$. Since A is a tailset, we know that $2^n \mu(A \cap [\sigma]) > 1 - \varepsilon$ for all σ of length n. Hence $\mu(A) > 1 - \varepsilon$. Since ε is arbitrary, $\mu(A) = 1$. □

5
Computability Theory

Important Note

In this chapter we will develop a significant amount of computability theory. Much of this technical material will not be needed until much later in the book, and perhaps in only a small section of the book. We have chosen to gather it in one place for ease of reference. However, as a result this chapter is quite uneven in difficulty, and we strongly recommend that the reader use most of it as a reference for later chapters, rather than reading through all of it in detail before proceeding. This is especially so for those readers unfamiliar with more advanced techniques such as priority arguments.

5.1 Computable functions, coding, and the halting problem

At the heart of our understanding of algorithmic randomness is the notion of an algorithm. Thus the tools we use are based on classical computability theory. While we expect the reader to have had at least one course in the rudiments of computability theory, such as a typical course on "theory of computation", the goal of this chapter is to give a reasonably self-contained account of the basics, as well as some of the tools we will need. We do, however, assume familiarity with the technical definition of computability via Turing machines (or some equivalent formalism). For more details the reader is referred to, for example, Salomaa [263], Rogers [253], Soare [280], or Odifreddi [233, 234].

Our initial concern is with functions from A into $\mathbb{N}$ where $A \subseteq \mathbb{N}$, i.e., *partial* functions on $\mathbb{N}$. If $A = \mathbb{N}$ then the function is called *total*. Looking only at $\mathbb{N}$ may seem rather restrictive. For example, later we will be concerned with functions that take the set of finite binary strings or subsets of the rationals as their domains and/or ranges. However, from the point of view of classical computability theory (that is, where *resources* such as time and memory do not matter), our definitions naturally extend to such functions by *coding*; that is, the domains and ranges of such functions can be coded as subsets of $\mathbb{N}$. For example, the rationals $\mathbb{Q}$ can be coded in $\mathbb{N}$ as follows.

Definition 5.1.1. Let $r \in \mathbb{Q} - \{0\}$ and write $r = (-1)^{\delta}\frac{p}{q}$ with $p, q \in \mathbb{N}$ in lowest terms and $\delta = 0$ or 1. Then define the Gödel number of r, denoted by $\#(r)$, as $2^{\delta}3^{p}5^{q}$. Let the Gödel number of 0 be 0.

The function $\#$ is an injection from $\mathbb{Q}$ into $\mathbb{N}$, and given $n \in \mathbb{N}$ we can decide exactly which $r \in \mathbb{Q}$, if any, has $\#(r) = n$. Similarly, if σ is a finite binary string, say $\sigma = a_1a_2\ldots a_n$, then we can define $\#(\sigma) = 2^{a_1+1}3^{a_2+1}\ldots(p_n)^{a_n+1}$, where p_n denotes the n-th prime. There are a myriad other codings possible, of course. For instance, one could code the string σ as the binary number 1σ, so that, for example, the string 01001 would correspond to 101001 in binary. Coding methods such as these are called "effective codings", since they include algorithms for deciding the resulting injections, in the sense discussed above for the Gödel numbering of the rationals.

Convention 5.1.2. Henceforth, unless otherwise indicated, when we discuss computability issues relating to a class of objects, we will always regard these objects as (implicitly) effectively coded in some way.

Part of the philosophy underlying computability theory is the celebrated Church-Turing thesis, which states that *the algorithmic (i.e., intuitively computable) partial functions are exactly those that can be computed by Turing machines on the natural numbers*. Thus, we formally adopt the definition of algorithmic, or computable, functions as being those that are computable by Turing machines, but argue informally, appealing to the intuitive notion of computability as is usual. An excellent discussion of the subtleties of the Church-Turing thesis can be found in Odifreddi [233].

There are certain important basic properties of the algorithmic partial functions that we will use throughout the book, often implicitly.

Property 5.1.3 (Enumeration Theorem – Universal Turing Machine)**.** *There is an algorithmic way of enumerating all the partial computable functions. That is, there is a list $\Phi_0, \Phi_1, \ldots$ of all such functions such that we have an algorithmic procedure for passing from an index i to a*

Turing machine computing Φ_i*, and vice-versa. Using such a list, we can define a partial computable function* $f(x, y)$ *of two variables such that* $f(x, y) = \Phi_x(y)$ *for all* x, y*. Such a function, and any Turing machine that computes it, are called* universal.

To a modern computer scientist, this result is obvious. That is, given a program in some computer language, we can convert it into ASCII code, and treat it as a number. Given such a binary number, we can decode the number and decide whether the number corresponds to the code of a program, and if so execute it. Thus a compiler for the given language yields a universal program.

For any partial computable function f, there are infinitely many ways to compute f. If Φ_y is one such algorithm for computing f, we say that y is an *index* for f.

The point of Property 5.1.3 is that we can pretend that we have all the machines $\Phi_1, \Phi_2, \ldots$ in front of us. For instance, to compute 10 steps in the computation of the 3rd machine on input 20, we can pretend to walk to the 3rd machine, put 20 on the tape and run it for 10 steps (we write the result as $\Phi_3(20)[10]$). Thus we can *computably* simulate the action of computable functions. In many ways, Property 5.1.3 is the platform that makes undecidability proofs work, since it allows us to diagonalize over the class of partial computable functions *without leaving this class.* For instance, we have the following result, where we write $\Phi_x(y)\downarrow$ to mean that Φ_x is defined on y, or equivalently, that the corresponding Turing machine halts on input y, and $\Phi_x(y)\uparrow$ to mean that Φ_x is not defined on y.

Proposition 5.1.4 (Unsolvability of the halting problem)**.** *There is no algorithm that, given* x, y*, decides whether* $\Phi_x(y)\downarrow$*. Indeed, there is no algorithm to decide whether* $\Phi_x(x)\downarrow$*.*

Proof. Suppose such an algorithm exists. Then by Property 5.1.3, it follows that the following function g is (total) computable:

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

Again using Property 5.1.3, there is a y with $g = \Phi_y$. Since g is total, $g(y)\downarrow$, so $\Phi_y(y)\downarrow$, and hence $g(y) = \Phi_y(y) + 1 = g(y) + 1$, which is a contradiction. □

Note that we *can* define a *partial* computable function g via $g(x) = \Phi_x(x) + 1$ and avoid contradiction, as it will follow that, for any index y for g, we have $\Phi_y(y)\uparrow = g(y)\uparrow$. Also, the reason for the use of *partial* computable functions in Property 5.1.3 is clear. The argument above also shows that there is no computable procedure to enumerate all (and only) the total computable functions.

Proposition 5.1.4 can be used to show that many problems are algorithmically unsolvable by "coding" the halting problem into these problems. For example, we have the following result.

Proposition 5.1.5. *There is no algorithm to decide whether the domain of Φ_x is empty.*

To prove this proposition, we need a lemma, known as the *s-m-n* Theorem. We state it for unary functions, but it holds for *n*-ary ones as well. Strictly speaking, the lemma below is the *s*-1-1 theorem. For the full statement and proof of the *s-m-n* Theorem, see [280].

Lemma 5.1.6 (The **s-m-n** Theorem, Kleene)**.** *Let $g(x, y)$ be a partial computable function of two variables. Then there is a computable function $s(x)$ such that, for all x, y,*

$$\Phi_{s(x)}(y) = g(x, y).$$

The proof of Lemma 5.1.6 runs as follows: Given a Turing machine M computing g and a number x, we can build a Turing machine N that on input y simulates the action of writing the pair (x, y) on M's input tape and running M. We can then find an index $s(x)$ for the function computed by N. We turn to the proof of Proposition 5.1.5.

Proof of Proposition 5.1.5. We code the halting problem into the problem of deciding whether $\mathrm{dom}(\Phi_x) = \emptyset$. That is, we show that *if* we could decide whether $\mathrm{dom}(\Phi_x) = \emptyset$ *then* we could solve the halting problem. Define a partial computable function of two variables by

$$g(x, y) = \begin{cases} 1 & \text{if } \Phi_x(x)\downarrow \\ \uparrow & \text{if } \Phi_x(x)\uparrow . \end{cases}$$

Notice that g ignores its second input.

Via the *s-m-n* Theorem, we can consider $g(x, y)$ as a computable collection of partial computable functions. That is, there is a computable $s(x)$ such that, for all x, y,

$$\Phi_{s(x)}(y) = g(x, y).$$

Now

$$\mathrm{dom}(\Phi_{s(x)}) = \begin{cases} \mathbb{N} & \text{if } \Phi_x(x)\downarrow \\ \emptyset & \text{if } \Phi_x(x)\uparrow, \end{cases}$$

so if we could decide for a given x whether $\Phi_{s(x)}$ has empty domain, then we could solve the halting problem. $\square$

5.2 Computable enumerability and Rice's Theorem

We now show that the reasoning used in the proof of Proposition 5.1.5 can be pushed much further. First we wish to regard all problems as coded by subsets of $\mathbb{N}$. For example, the halting problem can be coded by

$$\emptyset' := \{x : \Phi_x(x)\downarrow\}$$

(or if we insist on the two-variable formulation, by $\{\langle x, y\rangle : \Phi_x(y)\downarrow\}$). Next we need some terminology.

Definition 5.2.1. A set $A \subseteq \mathbb{N}$ is called

(i) *computably enumerable* (*c.e.*) if $A = \mathrm{dom}(\Phi_e)$ for some e, and

(ii) *computable* if A and $\overline{A} := \mathbb{N} - A$ are both computably enumerable.

A set is *co-c.e.* if its complement is c.e. Thus a set is computable iff it is both c.e. and co-c.e. Of course, it also makes sense to say that A is computable if its characteristic function χ_A is computable, particularly since, as mentioned in Chapter 4, we identify sets with their characteristic functions. It is straightforward to check that A is computable in the sense of Definition 5.2.1 if and only if χ_A is computable.

We let W_e denote the e-th computably enumerable set, that is, $\mathrm{dom}(\Phi_e)$, and let $W_e[s] = \{x \leqslant s : \Phi_e(x)[s]\downarrow\}$, where $\Phi_e(x)[s]$ is the result of running the Turing machine corresponding to Φ_e for s many steps on input x. We think of $W_e[s]$ as the result of performing s steps in the enumeration of W_e.

The name *computably enumerable* comes from a notion of "effectively countable", via the following characterization, whose proof is a straightforward exercise.

Proposition 5.2.2. *An infinite set A is computably enumerable iff either $A = \emptyset$ or there is a total computable function f from $\mathbb{N}$ onto A. (If A is infinite then f can be chosen to be injective.)*

Thus we can think of an infinite computably enumerable set as an effectively infinite list (but *not necessarily in increasing numerical order*). Note that computable sets correspond to decidable questions, since if A is computable, then either $A \in \{\emptyset, \mathbb{N}\}$ or we can decide whether $x \in A$ as follows. Let f and g be computable functions such that $f(\mathbb{N}) = A$ and $g(\mathbb{N}) = \overline{A}$. Now enumerate $f(0), g(0), f(1), g(1), \ldots$ until x occurs (as it must). If x occurs in the range of f, then $x \in A$; if it occurs in the range of g, then $x \notin A$. An equivalent characterization is to say that A is computable iff its characteristic function is computable.

It is straightforward to show that $\emptyset'$ is computably enumerable. Thus, by Proposition 5.1.4, it is an example of a computably enumerable set that is not computable.

If A is c.e., then it clearly has a *computable approximation*, that is, a uniformly computable family $\{A_s\}_{s\in\mathbb{N}}$ of sets such that $A(n) = \lim_s A_s(n)$ for all n. (For example, $\{W_e[s]\}_{s\in\omega}$ is a computable approximation of W_e.) In Section 5.6, we will give an exact characterization of the sets that have computable approximations. In the particular case of c.e. sets, we can choose the A_s so that $A_0 \subseteq A_1 \subseteq A_2 \subseteq \cdots$. In the constructions we discuss, whenever we are given a c.e. set , we assume we have such an approximation, and think of A_s as the numbers put into A by stage s of the construction.

An *index set* is a set A such that if $x \in A$ and $\Phi_x = \Phi_y$ then $y \in A$. For example, $\{x : \mathrm{dom}(\Phi_x) = \emptyset\}$ is an index set. An index set can be thought of as coding a problem about computable functions (like the emptyness of domain problem) whose answer does not depend on the particular algorithm used to compute a function. Generalizing Proposition 5.1.5, we have the following result, which shows that nontrivial index sets are never computable.

Theorem 5.2.3 (Rice's Theorem)**.** *An index set A is computable (and so the problem it codes is decidable) iff $A = \mathbb{N}$ or $A = \emptyset$.*

Proof. The proof of Rice's Theorem is very similar to that of Proposition 5.1.5.

Let $A \notin \{\emptyset, \mathbb{N}\}$ be an index set. Let e be such that $\mathrm{dom}(\Phi_e) = \emptyset$. We can assume without loss of generality that $e \in \overline{A}$ (the case $e \in A$ being symmetric). Fix $i \in A$. By the s-m-n Theorem, there is a computable $s(x)$ such that, for all $y \in \mathbb{N}$,

$$\Phi_{s(x)}(y) = \begin{cases} \Phi_i(y) & \text{if } \Phi_x(x)\downarrow \\ \uparrow & \text{if } \Phi_x(x)\uparrow . \end{cases}$$

If $\Phi_x(x)\downarrow$ then $\Phi_{s(x)} = \Phi_i$ and so $s(x) \in A$, while if $\Phi_x(x)\uparrow$ then $\Phi_{s(x)} = \Phi_e$ and so $s(x) \notin A$. Thus, if A were computable then $\emptyset'$ would also be computable. $\square$

Of course, many nontrivial decision problems are not coded by index sets, and so can have decidable solutions.

5.3 The Recursion Theorem

Kleene's Recursion Theorem (also known as the Fixed Point Theorem) is a fundamental result in classical computability theory. It allows us to use an index for a computable function or c.e. set that we are building in a construction as part of that very construction. Thus it forms the theoretical underpinning of the common programming practice of having a routine make recursive calls to itself.

Theorem 5.3.1 (Recursion Theorem, Kleene)**.** *Let f be a total computable function. Then there is a number n, called a* fixed point *of f, such that*

$$\Phi_n = \Phi_{f(n)},$$

and hence

$$W_n = W_{f(n)}.$$

Furthermore, such an n can be computed from an index for f.

Proof. First define a total computable function d via the s-m-n Theorem so that

$$\Phi_{d(e)}(k) = \begin{cases} \Phi_{\Phi_e(e)}(k) & \text{if } \Phi_e(e)\downarrow \\ \uparrow & \text{if } \Phi_e(e)\uparrow . \end{cases}$$

Let i be such that

$$\Phi_i = f \circ d$$

and let $n = d(i)$. Notice that Φ_i is total. The following calculation shows that n is a fixed point for f.

$$\Phi_n = \Phi_{d(i)} = \Phi_{\Phi_i(i)} = \Phi_{f \circ d(i)} = \Phi_{f(n)}.$$

The explicit definition of n given above can clearly be carried out computably given an index for f. $\square$

There are many variations on this theme. For example, if $f(x, y)$ is computable, then there is a function $n(y)$ such that, for all y,

$$\Phi_{n(y)} = \Phi_{f(n(y),y)}.$$

This result is called the *Recursion Theorem with Parameters.* In Section 6.5, we will see that there is a version of the Recursion Theorem for functions computed by *prefix-free machines*, a class of machines that will play a key role in this book.

Here is a very simple application of the Recursion Theorem. We show that $\emptyset'$ is not an index set. Let f be a computable function such that $\Phi_{f(n)}(n)\downarrow$ and $\Phi_{f(n)}(m)\uparrow$ for all $m \neq n$. Let n be a fixed point for f, so that $\Phi_n = \Phi_{f(n)}$. Let $m \neq n$ be another index for Φ_n. Then $\Phi_n(n)\downarrow$ and hence $n \in \emptyset'$, but $\Phi_m(m)\uparrow$ and hence $m \notin \emptyset$. So $\emptyset'$ is not an index set. Note that this example also shows that there is a Turing machine that halts only on its own index.

5.4 Reductions

The key concept used in the proof of Rice's Theorem is that of *reduction*, that is, the idea that "if we can do B then this ability also allows us to do A". In other words, *questions about problem A are* reducible *to ones about problem B*. We want to use this idea to define (partial) orderings that calibrate problems according to computational difficulty. The idea is to have $A \leqslant B$ if the ability to solve B allow us to also solve A, meaning that B is "at least as hard as" A. In this section, we introduce several ways to formalize this notion, beginning with the best-known one, Turing reducibility.

5.4.1 Oracle machines and Turing reducibility

An *oracle machine* is a Turing machine with an extra infinite read-only *oracle tape*, which it can access one bit at a time while performing its computation. If there is an oracle machine M that computes the set A when its oracle tape codes the set B, then we say that A is *Turing reducible* to B, or *B-computable*, or *computable in B*, and write $A \leqslant_{\mathrm{T}} B$. [1] Note that, in computing $A(n)$ for any given n, the machine M can make only finitely many queries to the oracle tape; in other words, it can access the value of $B(m)$ for at most finitely many m. This definition can be extended to functions in the obvious way.

For example, let $E = \{x : \mathrm{dom}(\Phi_x) \neq \emptyset\}$. In the proof of Proposition 5.1.5, we showed that $\emptyset' \leqslant_{\mathrm{T}} E$. Indeed the proof of Rice's Theorem demonstrates that, for a nontrivial index set I, we always have $\emptyset' \leqslant_{\mathrm{T}} I$. On the other hand, the unsolvability of the halting problem implies that $\emptyset' \nleqslant_{\mathrm{T}} \emptyset$. (Note that $X \nleqslant_{\mathrm{T}} \emptyset$ iff X is computable.)

We write $A \equiv_{\mathrm{T}} B$ if $A \leqslant_{\mathrm{T}} B$ and $B \leqslant_{\mathrm{T}} A$. We write $A <_{\mathrm{T}} B$ if $A \leqslant_{\mathrm{T}} B$ and $B \nleqslant_{\mathrm{T}} A$. Finally, we write $A \mid_{\mathrm{T}} B$ if $A \nleqslant_{\mathrm{T}} B$ and $B \nleqslant_{\mathrm{T}} A$.

It is not hard to check that Turing reducibility is transitive and reflexive, and thus is a pre-ordering on the subsets of $\mathbb{N}$. The equivalence classes of the form $\deg(A) := \{B : B \equiv_{\mathrm{T}} A\}$ code a notion of equicomputability and are called *(Turing) degrees (of unsolvability)*. We always use boldface letters such as $\mathbf{a}$ for Turing degrees. A Turing degree is *computably enumerable* if it contains a computably enumerable set (which does not imply that all the sets in the degree are c.e.). The Turing degrees inherit a natural ordering from Turing reducibility: $\mathbf{a} \leqslant \mathbf{b}$ iff $A \leqslant_{\mathrm{T}} B$ for some (or equivalently all) $A \in \mathbf{a}$ and $B \in \mathbf{b}$. We will relentlessly mix notation by writing, for example, $A <_{\mathrm{T}} \mathbf{b}$, for a set A and a degree $\mathbf{b}$, to mean that $A <_{\mathrm{T}} B$ for some (or equivalently all) $B \in \mathbf{b}$.

---

[1] We can also put resource bounds on our procedures. For example, if we count steps and ask that computations halt in a polynomial (in the length of the input) number of steps, then we arrive at the polynomial computable functions and the notion of polynomial time reducibility. We will not consider such reducibilities here, but refer the reader to Ambos-Spies and Mayordomo [9].

We let $\mathbf{0}$ denote the degree of the computable sets. Note that each degree is countable and has only countably many predecessors (since there are only countably many oracle machines), so there are continuum many degrees.

For an oracle machine Φ, we write Φ^A for the function computed by Φ with oracle A (i.e., with A coded into its oracle tape). The analog of Proposition 5.1.5 holds for oracle machines. That is, there is an effective enumeration $\Phi_0, \Phi_1, \ldots$ of all oracle machines, and a universal oracle machine Φ such that $\Phi^A(x, y) = \Phi_x^A(y)$ for all x, y and all oracles A.

We had previously defined Φ_e to be the eth partial computable function. However, we have already identified partial computable functions with Turing machines, and we can regard normal Turing machines as oracle machines with empty oracle tape; in fact it is convenient to identify the eth partial computable function Φ_e with $\Phi_e^{\emptyset}$. Thus there is no real conflict between the two notations, and we will not worry about the double meaning of Φ_e.

When a set A has a computable approximation $\{A_s\}_{s \in \mathbb{N}}$, then we write $\Phi_e^A(n)[s]$ to mean the result of running Φ_e with oracle A_s on input n for s many steps.

The *use* of a converging oracle computation $\Phi^A(n)$ is $x+1$ for the largest number x such that the value of $A(x)$ is queried during the computation. (If no such value is queried, then the use of the computation is 0.) We denote this use by $\varphi^A(n)$. In general, when we have an oracle computation represented by an uppercase Greek letter, its use function is represented by the corresponding lowercase Greek letter. Although in a sense trivial, the following principle is quite important.

Lemma 5.4.1 (Use principle)**.** *Let $\Phi^A(n)$ be a converging oracle computation, and let B be a set such that $B \upharpoonright \varphi^A(n) = A \upharpoonright \varphi^A(n)$. Then $\Phi^B(n) = \Phi^A(n)$.*

Proof. The sets A and B give the same answers to all questions asked during the relevant computations, so the results must be the same. $\square$

For a set A, let

$$A' = \{e : \Phi_e^A(e)\downarrow\}.$$

The set A' represents the halting problem *relativized* to A. The general process of extending a definition or result in the non-oracle case to the oracle case is known as *relativization*. For instance, Proposition 5.1.4 (the unsolvability of the halting problem) can be relativized with a completely analogous proof to show that $A \not\leqslant_{\mathrm{T}} A'$ for all A. Two good (but not hard) exercises are to show that $A <_{\mathrm{T}} A'$ and that if $A \equiv_{\mathrm{T}} B$ then $A' \equiv_{\mathrm{T}} B'$.

Another important example of relativization is the concept of a set B being *computably enumerable in* a set A, which means that $B = \mathrm{dom}(\Phi_e^A)$ for some e.

We often refer to A' as the *jump* of A. The *jump operator* is the function $A \mapsto A'$. The *nth jump of* A, written as $A^{(n)}$, is the result of applying the jump operator n times to A. So, for example, $A^{(2)} = A''$ and $A^{(3)} = A'''$. If $\mathbf{a} = \deg(A)$ then we write $\mathbf{a}'$ for $\deg(A')$, and similarly for the nth jump notation. This definition makes sense by the second exercise mentioned in the previous paragraph. Note that we have a hierarchy of degrees $\mathbf{0} < \mathbf{0}' < \mathbf{0}'' < \cdots$.

5.4.2 Strong reducibilities

The reduction used in the proof of Rice's Theorem is of a particularly strong type, since to decide whether $x \in \emptyset'$, we simply compute $s(x)$ and ask whether $s(x) \in A$. Considering this kind of reduction leads to the following definition.

Definition 5.4.2 (m-reducibility)**.** We say that A is *many-one reducible* (*m-reducible*) to B, and write $A \leqslant_{\mathrm{m}} B$, if there is a total computable function f such that for all x,

$$x \in A \text{ iff } f(x) \in B.^2$$

We will ignore the trivial cases of the above definition where $A = \emptyset$ or $B = \mathbb{N}$. If the function f in the definition of m-reduction is injective, then we say that A is *1-reducible* to B, and write $A \leqslant_1 B$. Note that if B is c.e. and $A \leqslant_{\mathrm{m}} B$ then A is also c.e. Also note that $\emptyset'$ is *m-complete*, in the sense that $A \leqslant_{\mathrm{m}} \emptyset'$ for any c.e. set A (because we can easily define a computable function f such that $n \in A$ iff $f(n) \in \emptyset'$).

It is not difficult to construct sets A and B such that $A \leqslant_{\mathrm{T}} B$ but $A \nleqslant_{\mathrm{m}} B$. For example, $\overline{\emptyset'} \leqslant_{\mathrm{T}} \emptyset'$, but $\overline{\emptyset'} \nleqslant_{\mathrm{m}} \emptyset'$, since $\overline{\emptyset'}$ is not c.e. It is also possible to construct such examples in which A and B are both c.e. Thus m-reducibility strictly refines Turing reducibility, and hence we say that m-reducibility is an example of a *strong reducibility.* There are many other strong reducibilities. Their definitions depend on the types of oracle access used in the corresponding reductions. We mention two that will be important in this book.

One of the key aspects of Turing reducibility is that a Turing reduction may be adaptable, in the sense that the number and type of queries made of the oracle depends upon the oracle itself. For instance, imagine a reduction that works as follows: on input x, the oracle is queried as to whether it contains some power of x. That is, the reduction asks whether 1 is in the

[2] In the context of complexity theory, resource bounded versions of m-reducibility are at the basis of virtually all modern NP-completeness proofs. Although Cook's original definition of NP-completeness was in terms of polynomial time *Turing* reducibility, the Karp version in terms of polynomial time m-reducibility is most often used. It is still an open question of structural complexity theory whether there is a set A such that the polynomial time T-degree of A collapses to a single polynomial time m-degree.

oracle, then whether x is in the oracle, then whether x^2 is in the oracle, and so on. If the answer is yes for some x^n, then the reduction checks whether the least such n is even, in which case it outputs 0, or odd, in which case it outputs 1.

Note that there is *no limit* to the number of questions asked of the oracle on a given input. This number depends on the oracle. Indeed, if the oracle happens to contain no power of x, then the computation on input x will not halt at all, and infinitely many questions will be asked.

Many naturally arising reductions do not have this adaptive property. One class of examples gives rise to the notion of *truth-table reducibility*. A *truth table* on the variables $v_1, v_2, \dots$ is a (finite) boolean combination σ of these variables. We write $A \vDash \sigma$ if A satisfies σ with v_n interpreted by n. For example, σ might be $((v_1 \wedge v_2) \vee (v_3 \rightarrow v_4)) \wedge v_5)$, in which case $A \vDash \sigma$ iff $5 \in A$ and either $3 \notin A$ or $4 \in A$ or both $1 \in A$ and $2 \in A$. Let $\sigma_0, \sigma_1, \dots$ be an effective list of all truth tables.

Definition 5.4.3 (Truth-table reducibility)**.** We say that A is *truth-table reducible* to B, and write $A \leqslant_{\mathrm{tt}} B$, if there is a computable function f such that for all x,

$$x \in A \text{ iff } B \vDash \sigma_{f(x)}.$$

Notice that an m-reduction is a simple example of a tt-reduction. The relevant truth table for input n has a single entry $v_{f(n)}$, where f is the function witnessing the given m-reduction.

One way to look at a tt-reduction is that it is one in which the oracle queries to be performed on a given input are predetermined, independently of the oracle, and the computation halts for every oracle. Removing the last restriction yields the notion of *weak truth-table reduction*, which can also be thought of as bounded Turing reduction, in the sense that there is a computable bound, independent of the oracle, on the amount of the oracle to be queried on a given input.

Definition 5.4.4 (Weak truth-table reducibility)**.** We say that A is *weak truth-table reducible* (*wtt-reducible*) to B, and write $A \leqslant_{\mathrm{wtt}} B$, if there are a computable function f and an oracle Turing machine Φ such that $\Phi^B = A$ and for all x, the largest element queried by Φ in the computation of $\Phi^B(x)$ is less than $f(x)$.

The same kinds of notation that we introduced for Turing reducibility also apply here, and for any other reducibility. For example, for any reducibility R, we write $A \equiv_{\mathrm{R}} B$ to mean that $A \leqslant_{\mathrm{R}} B$ and $B \leqslant_{\mathrm{R}} A$, and the R-degree $\deg_{\mathrm{R}}(A)$ of A is the set of all B such that $A \equiv_{\mathrm{R}} B$. The reducibilities we have seen so far calibrate sets into degrees of greater and greater fineness, in the order T, wtt, tt, m, 1.

5.5 The arithmetic hierarchy

We define the notions of Σ^0_n, Π^0_n, and Δ^0_n sets as follows. A set A is Σ^0_n if there is a computable relation $R(x_1, \dots, x_n, y)$ such that $y \in A$ iff

$$\underbrace{\exists x_1 \, \forall x_2 \, \exists x_3 \, \forall x_4 \cdots Q_n x_n}_{n \text{ alternating quantifiers}} R(x_1, \dots, x_n, y).$$

Since the quantifiers alternate, Q_n is $\exists$ if n is odd and $\forall$ if n is even. In this definition, we could have had n alternating quantifier *blocks*, instead of single quantifiers, but we can always collapse two successive existential or universal quantifiers into a single one by using pairing functions, so that would not make a difference.

The definition of A being Π^0_n is the same, except that the leading quantifier is a $\forall$ (but there still are n alternating quantifiers in total). It is easy to see that A is Π^0_n iff $\overline{A}$ is Σ^0_n.

Finally, we say a set is Δ^0_n if it is both Σ^0_n and Π^0_n (or equivalently, if both it and its complement are Σ^0_n). Note that the Δ^0_0, Π^0_0, and Σ^0_0 sets are all exactly the computable sets. The same is true of the Δ^0_1 sets, as shown by Theorem 5.5.1 below.

These notions give rise to Kleene's *arithmetical hierarchy*, which can be pictured as follows.

$$\begin{array}{ccccccc}
 & \Pi^0_1 & & \Pi^0_2 & & & \\
\Delta^0_1 & & \Delta^0_2 & & \Delta^0_3 & & \dots \\
 & \Sigma^0_1 & & \Sigma^0_2 & & &
\end{array}$$

Here lines mean inclusion (rightward along the page) and all inclusions are proper.

As we will see in the next section, there is a strong relationship between the arithmetical hierarchy and enumeration. The following is a simple example at the lowest level of the hierarchy

Theorem 5.5.1 (Kleene). *A set A is computably enumerable iff A is Σ^0_1.*

Proof. Suppose A is c.e. Then $A = \mathrm{dom}(\Phi_e)$ for some e, so $n \in A$ iff $\exists s \, \Phi_e(y)[s]\downarrow$. Thus A is Σ^0_1.

Conversely, if A is Σ^0_1 then for some computable R we have $n \in A$ iff $\exists x \, R(x, n)$. Define a partial computable function g by letting $g(y) = 1$ at stage s iff $s \geqslant n$ and there is an $x < s$ such that $R(x, n)$ holds. Then $n \in A$ iff $n \in \mathrm{dom}(g)$, so A is c.e. $\square$

5.6 The Limit Lemma and Post's Theorem

There is an important generalization of Theorem 5.5.1 due to Post. It ties in the arithmetical hierarchy with the degrees of unsolvability, and gives completeness properties of degrees of the form $\mathbf{0}^{(n)}$, highlighting their importance. In this section we will look at this and related characterizations, beginning with Shoenfield's Limit Lemma.

Saying that a set A is c.e. can be thought of as saying that A has a computable approximation that, for each n, starts out by saying that $n \notin A$, and then changes its mind at most once. More precisely, there is a computable binary function f such that for all n we have $A(n) = \lim_s f(n,s)$, with $f(n,0) = 0$ and $f(n,s+1) \neq f(n,s)$ for at most one s. Generalizing this idea, Shoenfield's Limit Lemma characterizes the sets computable from the halting problem as being those that have computable approximations with *finitely many* mind changes, and hence are "effectively approximable". (In other words, the sets computable from $\emptyset'$ are exactly those that have computable approximations, as defined in Section 5.2.)

Lemma 5.6.1 (Shoenfield's Limit Lemma)**.** *For a set A, we have $A \leqslant_{\mathrm{T}} \emptyset'$ iff there is a computable binary function g such that, for all n,*

1. $\lim_s g(n,s)$ *exists (i.e., $|\{s : g(n,s) \neq g(n,s+1)\}| < \infty$), and*
2. $A(n) = \lim_s g(n,s)$.

Proof. ($\Rightarrow$) Suppose $A = \Phi_e^{\emptyset'}$ for some e. Fix a computable approximation $\{\emptyset'_s\}_{s\in\mathbb{N}}$ of $\emptyset'$. Define g by letting $g(n,s) = 0$ if either $\Phi_e^{\emptyset'}[s]\uparrow$ or $\Phi_e^{\emptyset'}[s]\downarrow\neq 1$; and letting $g(n,s) = 1$ otherwise. Fix n, and let s be a stage such that $\emptyset'_t \upharpoonright \varphi_e^{\emptyset'}(n) = \emptyset' \upharpoonright \varphi_e^{\emptyset'}(n)$ for all $t \geqslant s$. By the use principle (Lemma 5.4.1), $g(n,t) = \Phi_e^{\emptyset'}(n)[t] = \Phi_e^{\emptyset'}(n) = A(n)$ for all $t \geqslant s$. Thus g has the required properties.

($\Leftarrow$) Suppose such a function g exists. Without loss of generality, we can assume that $g(n,0) = 0$ for all n. To show that $A \leqslant_{\mathrm{T}} \emptyset'$, it is enough to build a c.e. set B such that $A \leqslant_{\mathrm{T}} B$, since every c.e. set is computable in $\emptyset'$. We put $\langle n,k\rangle$ into B whenever we find that

$$|\{s : g(n,s) \neq g(n,s+1)\}| \geqslant n.$$

Now define a Turing reduction Γ as follows. Given an oracle X, on input n, search for the least k such that $\langle n,k\rangle \notin X$, and if one is found, then output 0 if k is even and 1 is k is odd. Clearly, $\Gamma^B = A$. □

Remark 5.6.2. Intuitively, the proof of the "only if" direction of the limit lemma boils down to saying that, since $\emptyset'$ can decide whether $\exists s > t\,(g(n,s) \neq g(n,s+1))$ for any n and t, it can also compute $\lim_s g(n,s)$.

Corollary 5.6.3. *For a set A, we have $A \leqslant_{\mathrm{wtt}} \emptyset'$ iff there are a computable binary function g and a computable function h such that, for all n,*

1. $|\{s : g(n,s) \neq g(x,s+1)\}| < h(x)$, *and*
2. $A(n) = \lim_s g(n,s)$.

Proof. Mimic the proof of Theorem 5.6.1, observing that h gives the computable bound on the use. □

As we have seen, we often want to relativize results, definitions, and proofs in computability theory. The limit lemma relativizes to show that $A \leqslant_{\mathrm{T}} B'$ iff there is a B-computable binary function f such that $A(n) = \lim_s f(n,s)$ for all n. Combining this fact with induction, we have the following generalization of the limit lemma.

Corollary 5.6.4 (Limit lemma, strong form)**.** *Let $k \geqslant 1$. For a set A, we have $A \leqslant_{\mathrm{T}} \emptyset^{(k)}$ iff there is a computable function g of $k+1$ variables such that $A(n) = \lim_{s_1} \lim_{s_2} \dots \lim_{s_k} g(n, s_1, s_2, \dots, s_k)$ for all n.*

We now turn to Post's characterization of the levels of the arithmetical hierarchy. Let $\mathcal{C}$ be a class of sets (such as a level of the arithmetical hierarchy). A set A is $\mathcal{C}$ *complete* if $A \in \mathcal{C}$ and $B \leqslant_{\mathrm{T}} A$ for all $B \in \mathcal{C}$. If in fact $B \leqslant_{\mathrm{m}} A$ for all $B \in \mathcal{C}$, then we say that A is $\mathcal{C}$ *m-complete*, and similarly for other strong reducibilities.

Theorem 5.6.5 (Post's Theorem)**.** *Let $n \geqslant 0$.*

(i) *A set B is Σ^0_{n+1} iff B is c.e. in some Σ^0_n set iff B is c.e. in some Π^0_n set.*

(ii) *The set $\emptyset^{(n)}$ is Σ^0_n m-complete.*

(iii) *A set B is Σ^0_{n+1} iff B is c.e. in $\emptyset^{(n)}$.*

(iv) *A set B is Δ^0_{n+1} iff $B \leqslant_{\mathrm{T}} \emptyset^{(n)}$.*

Proof. (i) First note that if B is c.e. in A then B is also c.e. in $\overline{A}$. Thus, being c.e. in a Σ^0_n set is the same as being c.e. in a Π^0_n set, so all we need to show is that B is Σ^0_{n+1} iff B is c.e. in some Π^0_n set.

The "only if" direction has the same proof as the corresponding part of Theorem 5.5.1, except that the computable relation R in that proof is now replaced by a Π^0_n relation R.

For the "if" direction, let B be c.e. in some Π^0_n set A. Then, by Theorem 5.5.1 relativized to A, there is an e such that $n \in B$ iff

$$\exists s\, \exists \sigma \prec A\, (\Phi^\sigma_e(n)[s]\downarrow). \tag{5.1}$$

The property in parentheses is computable, while the property $\sigma \prec A$ is a combination of a Π^0_n statement (asserting that certain elements are in A) and a Σ^0_n statement (asserting that certain elements are not in A), and hence is Δ^0_{n+1}. So (5.1) is a Σ^0_{n+1} statement.

(ii) We have already seen that $\emptyset'$ is Σ^0_1 m-complete. Now suppose that $\emptyset^{(n)}$ is Σ^0_n m-complete. Since $\emptyset^{(n+1)}$ is c.e. in $\emptyset^{(n)}$, it is Σ^0_{n+1}. Let C be

Σ^0_{n+1}. By part (i), C is c.e. in some Σ^0_n set, and hence it is c.e. in $\emptyset^{(n)}$. As in the unrelativized case, it is now easy to define a computable function f such that $n \in C$ iff $f(n) \in \emptyset^{(n+1)}$. (In more detail, let e be such that $C = W_e^{\emptyset^{(n)}}$, and define f so that for all oracles X and all n and x, we have $\Phi^X_{f(n)}(x)\downarrow$ iff $n \in W_e^X$.)

(iii) By (i) and (ii), and the fact that if X is c.e. in Y and $Y \leqslant_{\mathrm{T}} Z$, then X is also c.e. in Z.

(iv) The set B is Δ^0_{n+1} iff B and $\overline{B}$ are both Σ^0_{n+1}, and hence both c.e. in $\emptyset^{(n)}$ by (ii). But a set and its complement are both c.e. in X iff the set is computable in X. Thus B is Δ^0_{n+1} iff $B \leqslant_{\mathrm{T}} \emptyset^{(n)}$. □

There are many "natural" sets, such as certain index sets, that are complete for various levels of the arithmetical hierarchy. The following result gives a few examples.

Theorem 5.6.6. (i) *Fin* $:= \{e : W_e$ *is finite*$\}$ *is* Σ^0_2 *m-complete.*

(ii) *Tot* $:= \{e : \Phi_e$ *is total*$\}$ *and Inf* $:= \{e : W_e$ *is infinite*$\}$ *are both* Π^0_2 *complete.*

(iii) *Cof* $:= \{e : W_e$ *is cofinite*$\}$ *is* Σ^0_3 *complete.*

Proof sketch. None of these are terribly difficult. We do (i) as an example. We know that $\emptyset''$ is Σ^0_2 m-complete by Post's Theorem, so it is enough to give an m-reduction from Fin to $\emptyset''$. It is easy to show, using the *s-m-n* Theorem, that there is a computable function f such that for all s and e, we have $s \in W_{f(e)}$ iff there is a $t \geqslant s$ such that either $\Phi_e^{\emptyset'}(e)[t]\uparrow$ or $\emptyset'[t+1] \upharpoonright \varphi_e^{\emptyset'}(e)[t] \neq \emptyset'[t] \upharpoonright \varphi_e^{\emptyset'}(e)[t]$. Then $f(e) \in \mathrm{Fin}$ iff $\Phi_e^{\emptyset'}(e)\downarrow$ iff $e \in \emptyset''$.

Part (ii) is similar, and (iii) is also similar but more intricate. See Soare [280] for more details. □

5.7 A note on reductions

There are several ways to describe a reduction procedure. Formally, $A \leqslant_{\mathrm{T}} B$ means that there is an e such that $\Phi_e^B = A$. In practice, though, we obviously never actually build an oracle Turing machine to witness the fact that $A \leqslant_{\mathrm{T}} B$, but avail ourselves of the Church-Turing Thesis to informally describe a reduction Γ such that $\Gamma^B = A$. One such description is given in the proof of the $\Leftarrow$ direction of Shoenfield's Limit Lemma (Lemma 5.6.1). This proof gives an example of a *static* definition of a reduction procedure, in that the action of Γ is specified by a rule, rather than being defined during a construction. As an example of a *dynamic* definition of a reduction procedure, we reprove the $\Leftarrow$ direction of the Limit Lemma.

Recall that we are given a computable binary function g such that, for all n,

1. $\lim_s g(n,s)$ exists and

2. $A(n) = \lim_s g(n,s)$.

We wish to show that $A \leqslant_{\mathrm{T}} \emptyset'$, by building a c.e. set B and a reduction $\Gamma^B = A$.

We simultaneously construct B and Γ in stages. We begin with $B_0 = \emptyset$. For each n, we leave the value of $\Gamma^B(n)$ undefined until stage n. At stage n, we let $\Gamma^B(n)[n] = g(n,n)$ with use $\gamma^B(n)[n] = \langle n, 0\rangle + 1$.

Furthermore, at stage $s > 0$ we proceed as follows for each $n < s$. If $g(n,s) = g(n,s-1)$, then we change nothing. That is, we let $\Gamma^B(n)[s] = \Gamma^B(n)[s-1]$ with the same use $\gamma^B(n)[s] = \gamma^B(n)[s-1]$. Otherwise, we enumerate $\gamma(n,s-1) - 1$ into B, which allows us to redefine $\Gamma^B(n)[s] = g(n,s)$, with use $\gamma^B(n)[s]$ defined to be the least $\langle n,k\rangle \notin B$.

It is not hard to check that $\Gamma^B = A$, and that in fact this reduction is basically the same as that in the original proof of the Limit Lemma (if we assume without loss of generality that $g(n,n) = 0$ for all n), at least as far as its action on oracle B goes.

More generally, the rules for a reduction Δ^C from a c.e. set C are as follows, for each input n.

1. Initially $\Delta^C(n)[0]\uparrow$.

2. At some stage s we must define $\Delta^C(n)[s]\downarrow = i$ for some value i, with some use $\delta^C(n)[s]$. By this action, we are promising that $\Delta^C(n) = i$ unless $C \upharpoonright \delta^C(n)[s] \neq C_s \upharpoonright \delta^C(n)[s]$.

3. The convention now is that $\delta^C(n)[t] = \delta^C(n)[s]$ for $t > s$ unless $C_t \upharpoonright \delta^C(n)[s] \neq C_s \upharpoonright \delta^C(n)[t]$. Should we find a stage $t > s$ such that $C_t \upharpoonright \delta^C(n)[s] \neq C_s \upharpoonright \delta^C(n)[t]$, we then again have $\Delta^C(n)[t]\uparrow$.

4. We now again must have a stage $u \geqslant t$ at which we define $\Delta^C(n)[u]\downarrow =$ j for some value j, with some use $\delta^C(n)[u]$. We then return to step 3, with u in place of s.

5. If Δ^C is to be total, we have to ensure that we stay at step 3 permanently from some point on. That is, there must be a stage u at which we define $\Delta^C(n)[u]$ and $\delta^C(n)[u]$, such that $C \upharpoonright \delta^C(n)[u] = C_u \upharpoonright \delta^C(n)[u]$. One way to achieve this is to ensure that, from some point on, whenever we redefine $\delta^C(n)[u]$, we set it to the same value.

In some constructions, C will be given to us, but in others we will build it along with Δ. In this case, when we want to redefine the value of the computation $\Delta^C(n)$ at stage s, we will often be able to do so by putting a number less than $\delta^C(n)[s]$ into C (as we did in the Limit Lemma example above).

Remark 5.7.1. There is a similar method of building a reduction Δ^C when C is not c.e., but merely Δ^0_2. The difference is that now we must promise that if $\Delta^C(n)[s]$ is defined and there is a $t > s$ such that $C_t \upharpoonright \delta^C(n)[s] = C_s \upharpoonright \delta^C(n)[s]$, then $\Delta^C(n)[t] = \Delta^C(n)[s]$ and $\delta^C(n)[t] = \delta^C(n)[s]$.

A more formal view of a reduction is as a partial computable map from strings to strings obeying certain continuity conditions. In this view, a reduction $\Gamma^B = A$ is specified by a partial computable function $f : 2^{<\omega} \to 2^{<\omega}$ such that

1. if $\sigma \prec B$, then $f(\sigma) \prec A$;
2. for all $\sigma \prec \tau$, if both $f(\sigma)\downarrow$ and $f(\tau)\downarrow$, then $f(\sigma) \preccurlyeq f(\tau)$; and
3. for all $\tau \prec A$ there is a $\sigma \prec B$ such that $\tau \prec f(\sigma)$.

The reduction in the proof of the $\Leftarrow$ direction of the Limit Lemma can be viewed in this way by letting $f(\sigma)$ be the longest string τ such that for all $n < |\tau|$, there is a k such that $\sigma(\langle n, k\rangle) = 0$, and $\tau(n) = 0$ iff k is even.

5.8 The finite extension method

In this section, we introduce one of the main techniques used in classical degree theory. We will refine this technique in the next section to what is known as the finite injury method.

The dynamic construction of a reduction from B to A in the Limit Lemma, discussed in the previous section, has much in common with many proofs in classical computability theory. We perform a *construction* where some object is built in *stages*. Typically, we have some overall *goal* that we break down into smaller subgoals that we argue are all met in the limit. In this case, the goal is to construct the reduction $\Gamma^B = A$. Here we break the goal into the subgoals of defining $\Gamma^B(n)$ for each n, and we accomplish these subgoals by using the information supplied by our "opponent", who is feeding us information about the universe, in this case the values $g(n, s)$.

As an archetype for such proofs, think of Cantor's proof that the collection of all infinite binary sequences is uncountable. One can conceive of this proof as follows. Suppose we could list the infinite binary sequences as $\mathcal{S} = \{S_0, S_1, \dots\}$, with $S_e = s_{e,0}s_{e,1}\dots$. It is our goal to construct a binary sequence $U = u_0u_1\dots$ that is not on the list $\mathcal{S}$. This should be thought of as a game against our opponent who must supply us with $\mathcal{S}$. We construct u in stages, at stage t specifying only $u_0 \dots u_t$, the initial segment of U of length $t+1$. Our list of *requirements* is the decomposition of the overall goal into subgoals of the form

$$\mathcal{R}_e : \; U \neq S_e.$$

There is one such requirement for each $e \in \mathbb{N}$. Of course, we know how to satisfy these requirements. At stage e, we simply ensure that $u_e \neq s_{e,e}$ by setting $u_e = 1 - s_{e,e}$. This action ensures that $U \neq S_e$ for all e; in other words, all the requirements are met. This fact contradicts the assumption that $\mathcal{S}$ lists all infinite binary sequences, as U is itself an infinite binary sequence.

Notice that if we define a real to be *computable* if it has a computable binary expansion, then the above proof shows that there is no computable enumeration of all the computable reals (modulo some unimportant technicalities involving nonunique representations of reals).

Clearly, the proof of the unsolvability of the halting problem can also be similarly recast, where this time the e-th requirement asks us to kill the e-th member of some supposed list of all algorithms deciding whether $\Phi_e(e)\downarrow$.

While later results will be more complicated than these easy examples, readers should keep the overall *structure* of the above in mind: *Our constructions will be in finite steps, where one or more objects are constructed stage by stage in finite pieces. These objects will be constructed to satisfy a list of requirements. The strategy we use will be dictated by how our opponent reveals the universe to us. Our overall goal is to satisfy all requirements in the limit.*

To finish this section, we look at a slightly more involved version of this technique. While we know that there are uncountably many Turing degrees, the only ones we have seen so far are the iterates of the halting problem. Rice's Theorem (Theorem 5.2.3) shows that all index sets are of degree $\geqslant \mathbf{0}'$. In 1944, Post [238] observed that all computably enumerable problems known at the time were either computable or of Turing degree $\mathbf{0}'$. He asked the following question.

Question 5.8.1 (Post's Problem)**.** *Does there exist a computably enumerable degree* $\mathbf{a}$ *with* $\mathbf{0} < \mathbf{a} < \mathbf{0}'$*?*

As we will see in the next section, Post's problem was finally given a positive answer by Friedberg [110] and Muchnik [220], using a new and ingenious method called the priority method. This method was an "effectivization" of an earlier method discovered by Kleene and Post [146]. It is called the finite extension method, and was used to prove the following result.

Theorem 5.8.2 (Kleene and Post [146])**.** *There are degrees* $\mathbf{a}$ *and* $\mathbf{b}$*, both below* $\mathbf{0}'$*, such that* $\mathbf{a} \mid \mathbf{b}$*. In other words, there are* $\emptyset'$*-computable sets that are incomparable under Turing reducibility.*[3]

[3]The difference between the Kleene-Post Theorem and the solution to Post's Problem is that the degrees constructed in the proof of Theorem 5.8.2 are not necessarily computably enumerable, but merely Δ^0_2.

Proof. We construct $A = \lim_s A_s$ and $B = \lim_s B_s$ in stages, to meet the following requirements for all $e \in \mathbb{N}$.

$$\mathcal{R}_{2e} : \Phi_e^A \neq B.$$
$$\mathcal{R}_{2e+1} : \Phi_e^B \neq A.$$

Note that if $A \leqslant_{\mathrm{T}} B$ then there must be some procedure Φ_e with $\Phi_e^B = A$. Hence, if we meet all our requirements then $A \nleqslant_{\mathrm{T}} B$, and similarly $B \nleqslant_{\mathrm{T}} A$, so that A and B have incomparable Turing degrees. The fact that $A, B \leqslant_{\mathrm{T}} \emptyset'$ will come from the construction and will be observed at the end.

The argument is by finite extensions, in the sense that at each stage s we specify a finite portion A_s of A and a finite portion B_s of B. These finite portions A_s and B_s will be specified as binary strings. The key invariant that we need to maintain throughout the construction is that $A_s \preccurlyeq A_u$ and $B_s \preccurlyeq B_u$ for all stages $u \geqslant s$. Thus, after stage s we can only *extend* the portions of A and B that we have specified by stage s, which is a hallmark of the finite extension method.

Construction.

Stage 0. Let $A_0 = B_0 = \lambda$ (the empty string).

Stage $2e+1$. (Attend $\mathcal{R}_{2e}$.) We will have specified A_{2e} and B_{2e} at stage $2e$. Pick some number x, called a *witness*, with $x \geqslant |B_{2e}|$, and check whether there is a string σ properly extending A_{2e} such that $\Phi_e^\sigma(x)\downarrow$.

If such a σ exists, then let A_{2e+1} be the length-lexicographically least such σ. Let B_{2e+1} be the string of length $x+1$ extending B_{2e} such that $B_{2e+1}(n) = 0$ if $|B_{2e}| \leqslant n < x$ and $B_{2e+1}(x) = 1 - \Phi_e^\sigma(x)$.

If no such σ exists, then let $A_{2e+1} = A_{2e}0$ and $B_{2e+1} = B_{2e}0$.

Stage $2e+2$. (Attend $\mathcal{R}_{2e+1}$.) Define A_{2e+2} and B_{2e+2} by proceeding in the same way as at stage $2e+1$, but with the roles of A and B reversed.

End of Construction.

Verification. First note that we have $A_0 \prec A_1 \prec \cdots$ and $B_0 \prec B_1 \prec \cdots$, so A and B are well-defined.

We now prove that we meet the requirement $\mathcal{R}_n$ for each n; in fact, we show that we meet $\mathcal{R}_n$ at stage $n+1$. Suppose that $n = 2e$ (the case where n is odd being completely analogous). At stage $n+1$, there are two cases to consider. Let x be as defined at that stage.

If there is a σ properly extending A_n with $\Phi_e^\sigma(x)\downarrow$, then our action is to adopt such a σ as A_{n+1} and define B_{n+1} so that $\Phi_e^{A_{n+1}}(x) \neq B_{n+1}(x)$. Since A extends A_{n+1} and $\Phi_e^{A_{n+1}}(x)\downarrow$, it follows that A and A_{n+1} agree on the use of this computation, and hence $\Phi_e^A(x) = \Phi_e^{A_{n+1}}$. Since B extends B_{n+1}, we also have $B(x) = B_{n+1}(x)$. Thus $\Phi_e^A(x) \neq B(x)$, and $\mathcal{R}_n$ is met.

If there is no σ extending A_n with $\Phi_e^\sigma(x)\downarrow$, then since A is an extension of A_n, it must be the case $\Phi^A(x)\uparrow$, and hence $\mathcal{R}_n$ is again met.

Finally we argue that $A, B \leqslant_{\mathrm{T}} \emptyset'$. Notice that the construction is in fact fully computable except for the decision as to which case we are in at a given stage. There we must decide whether there is a convergent computation of a particular kind. For instance, at stage $2e+1$ we must decide whether the following holds:

$$\exists\tau\,\exists s\,[\tau \succ A_{2e} \wedge \Phi_e^{\tau}(x)[s]\downarrow]. \tag{5.2}$$

This is a Σ_1^0 question, uniformly in x, and hence can be decided by $\emptyset'$.[4] $\square$

The reasoning at the end of the above proof is quite common: we often make use of the fact that $\emptyset'$ can answer any Δ_2^0 question, and hence any Σ_1^0 or Π_1^0 question.

A key ingredient of the proof of Theorem 5.8.2 is the use principle (Lemma 5.4.1). In constructions of this sort, where we build objects to defeat certain oracle computations, a typical requirement will say something like "the reduction Γ is not a witness to $A \leqslant_{\mathrm{T}} B$." If we have a converging computation $\Gamma^B(n)[s] \neq A(n)[s]$ and we "preserve the use" of this computation by not changing B after stage s on the use $\gamma^B(n)[s]$ (and similarly preserve $A(n)$), then we will preserve this disagreement. But this use is only a finite portion of B, so we still have all the numbers bigger than it to meet other requirements. In the finite extension method, this use preservation is automatic, since once we define $B(x)$ we never redefine it, but in other constructions we will introduce below, this may not be the case, because we may have occasion to redefine certain values of B. In that case, to ensure that $\Gamma^B \neq A$, we will have to structure the construction so that, if Γ^B is total, then there are n and s such that $\Gamma^B(n)[s] \neq A(n)[s]$ and, from stage s on, we preserve both $A(n)$ and $B \upharpoonright \gamma^B(n)[s]$.

5.9 Post's Problem and the finite injury method

A more subtle generalization of the finite extension method is the *priority method*. We begin by looking at the simplest incarnation of this elegant technique, the *finite injury priority method*. This method is somewhat like the finite extension method, but with backtracking.

The idea behind this method is the following. Suppose we must again satisfy requirements $\mathcal{R}_0, \mathcal{R}_1, \ldots$, but this time we are constrained to some sort of effective construction, so we are not allowed questions of a noncomputable oracle during the construction. As an illustration, let us reconsider Post's Problem (Question 5.8.1). Post's Problem asks us to find a c.e. degree strictly between $\mathbf{0}$ and $\mathbf{0}'$. It is clearly enough to construct c.e. sets A and B with incomparable Turing degrees. The Kleene-Post method does allow us to construct sets with incomparable degrees below $\mathbf{0}'$, but these

[4] More precisely, we use the *s-m-n* Theorem to construct a computable ternary function f such that for all e, σ, x, and z, we have $\Phi_{f(e,\sigma,x)}(z)\downarrow$ iff (5.2) holds. Then (5.2) holds iff $f(e,\sigma,x) \in \emptyset'$.

sets are *not* computably enumerable, because in that construction we satisfy the relevant requirements *in order*, using an $\emptyset'$ oracle question at each stage. To make A and B c.e., we must have a computable construction where elements go into the sets A and B but never leave them. The key idea, discovered independently by Friedberg [110] and Muchnik [220], is to pursue multiple strategies for each requirement, in the following sense.

In the proof of the Kleene-Post Theorem, it appears that, in satisfying the requirement $\mathcal{R}_{2e}$, we need to know whether or not there is a σ extending A_{2e} such that $\Phi_e^\sigma(x)\downarrow$, where x is our chosen witness. Now our idea is to first *guess* that no such σ exists, which means that nothing will be done for $\mathcal{R}_{2e}$, save keeping x out of B, unless at some point we find an appropriate σ, at which point we will make A extend σ and put x into B if necessary, as in the Kleene-Post construction.

The only problem is that putting x into B may well upset the action of other requirements of the form $\mathcal{R}_{2i+1}$, because such a requirement might need B to extend some string τ (for the same reason that $\mathcal{R}_{2e}$ needs A to extend σ, which may no longer be possible. If we nevertheless put x into B, we say that we have *injured* $\mathcal{R}_{2i+1}$. Of course, $\mathcal{R}_{2i+1}$ can now choose a new witness and start over from scratch, but perhaps another requirement may injure it again later. So we need to somehow ensure that, for each requirement, there is a stage after which it is never injured.

To make sure that this we put a *priority ordering* on our requirements, by stating that $\mathcal{R}_j$ has stronger priority than $\mathcal{R}_i$ if $j < i$, and allow $\mathcal{R}_j$ to injure $\mathcal{R}_i$ only if $\mathcal{R}_j$ has stronger priority than $\mathcal{R}_i$. Thus $\mathcal{R}_0$ is never injured. The requirement $\mathcal{R}_1$ may be injured by the action of $\mathcal{R}_0$. However, once this happens $\mathcal{R}_0$ will never act again, so if it does happen, then we *initialize* $\mathcal{R}_1$, meaning that we restart its action with a new witness, chosen to be larger than any number previously seen in the construction, and hence larger than any number $\mathcal{R}_0$ cares about. This new incarnation of $\mathcal{R}_1$ is guaranteed never to be injured. It should now be clear that, by induction, each requirement will eventually reach a point, following a finite number of initializations, after which it will never be injured and hence will succeed in reaching its goal.

The reader may think of this kind of constructions as a game between a team of industrialists (each trying to erect a factory) and a team of environmentalists (each trying to build a park). In the end we want the world to be happy. In other words, we want all desired factories and parks to be built. Members of the two teams have their own places in the pecking order. For instance, industrialist 6 has higher priority than all environmentalists except the first six, and therefore can build anywhere except on parks built by the first six environmentalists. For example, industrialist 6 may choose to build on land already demarcated by environmentalist 10, who would then need to find another place to build a park. Of course, even if this happens, a higher ranked environmentalist, such as number 3, for instance, could later lay claim to that same land, forcing industrialist 6 to find another place to build a factory. Whether the highest ranked industrialist has priority over the highest ranked environmentalist or vice-versa is irrelevant to the construction, so we leave that detail to each reader's political leanings.

In general, in a finite injury priority argument, we have a list of requirements in some priority ordering. There are several different ways to meet each individual requirement. Exactly which way will be possible to implement depends upon information that is not initially available to us but is "revealed" to us during the construction. The problem is that actions by one requirement can injure others. We must arrange things so that only requirements of higher priority can injure ones of lower priority, and we can always restart the ones of lower priority once they are injured. In a finite injury argument, any requirement *requires attention* only finitely often, and we argue by induction that each requirement eventually gets an environment wherein it can be met. As we will later see, there are much more complex infinite injury arguments where one requirement might injure another infinitely often, but the key there is that the injury is somehow controlled so that it is still the case that each requirement eventually gets an environment wherein it can be met. Of course, imposing this *coherence criterion* on our constructions means that each requirement must ensure that its action does not prevent weaker requirements from finding appropriate environments. (A principle known as Harrington's "golden rule".) For a more thorough account of these beautiful techniques and their uses in modern computability theory, see Soare [280].

We now turn to the formal description of the solution of Post's Problem by Friedberg and Muchnik, which was the first use of the priority method. In Chapter 15, we will return to Post's Problem and explore its connections to Kolmogorov complexity.

Theorem 5.9.1 (Friedberg [110], Muchnik [220]). *There exist computably enumerable sets A and B such that A and B have incomparable Turing degrees.*

Proof. We build $A = \bigcup_s A_s$ and $B = \bigcup_s B_s$ in stages to satisfy the same requirements as in the proof of the Kleene-Post Theorem. That is, we make A and B c.e. while meeting the following requirements for all $e \in \mathbb{N}$.

$$\mathcal{R}_{2e} : \Phi_e^A \neq B.$$
$$\mathcal{R}_{2e+1} : \Phi_e^B \neq A.$$

The strategy for a single requirement. We begin by looking at the strategy for a single requirement $\mathcal{R}_{2e}$. We first pick a witness x to *follow* $\mathcal{R}_{2e}$. This number is targeted for B, and, of course, we initially keep x out of B. We then wait for a stage s such that $\Phi_e^A(x)[s]\downarrow = 0$. If such a stage does not occur, then either $\Phi_e^A(x)\uparrow$ or $\Phi_e^A(x)\downarrow \neq 0$. In either case, since we keep x out of B, we have $\Phi_e^A(x) \neq 0 = B(x)$, and hence $\mathcal{R}_{2e}$ is satisfied.

If a stage s as above occurs, then we put x into B and *protect* A_s. That is, we try to ensure that any number entering A from now on is greater than any number seen in the construction thus far, and hence in particular greater than $\varphi_e^A(x)[s]$. If we succeed then, by the use principle, $\Phi_e^A(x) = \Phi_e^A(x)[s] = 0 \neq B(x)$, and hence again $\mathcal{R}_{2e}$ is satisfied.

Note that when we take this action, we might injure a requirement $\mathcal{R}_{2i+1}$ that is trying to preserve the use of a computation $\Phi_i^B(x')$, since x may be below this use. As explained above, the priority mechanism will ensure that this can happen only if $2i + 1 > 2e$.

We now proceed with the full construction. We will denote by A_s and B_s the sets of elements enumerated into A and B, respectively, by the end of stage s.

Construction.
Stage 0. Let $A_0 = B_0 = \emptyset$.

Stage $s + 1$. Say that $\mathcal{R}_j$ *requires attention* at this stage if one of the following holds.

(i) $\mathcal{R}_j$ currently has no follower.

(ii) $\mathcal{R}_j$ has a follower x and either
  (a) $j = 2e$ and $\Phi_e^A(x)[s]\downarrow= 0 = B_s(x)$ or
  (b) $j = 2e + 1$ and $\Phi_e^B(x)[s]\downarrow= 0 = A_s(x)$.

Find the least j with $\mathcal{R}_j$ requiring attention. (If there is none, then proceed to the next stage.) We suppose that $j = 2e$, the odd case being symmetric. If $\mathcal{R}_{2e}$ has no follower, then let x be a *fresh large* number (that is, one larger than all numbers seen in the construction so far) and appoint x as $\mathcal{R}_{2e}$'s follower.

If $\mathcal{R}_{2e}$ has a follower x, then it must be the case that $\Phi_e^A(x)[s]\downarrow= 0 = B_s(x)$. In this case, enumerate x into B and *initialize* all r_k with $k > 2e$ by canceling all their followers.

In either case, we say that $\mathcal{R}_{2e}$ *receives attention* at stage s.
End of Construction.

Verification. We prove by induction that, for each j,

(i) $\mathcal{R}_j$ receives attention only finitely often, and

(ii) $\mathcal{R}_j$ is met.

Suppose that (i) holds for all $k < j$. Suppose that $j = 2e$ for some e, the odd case being symmetric. Let s be the least stage such that for all $k < j$, the requirement $\mathcal{R}_k$ does not require attention after stage s. By the minimality of s, some requirement $\mathcal{R}_k$ with $k < j$ received attention at stage s, and hence $\mathcal{R}_j$ does not have a follower at the beginning of stage $s+1$. Thus, $\mathcal{R}_j$ requires attention at stage $s+1$, and is appointed a follower x. Since $\mathcal{R}_j$ cannot have its follower canceled unless some $\mathcal{R}_k$ with $k < j$ receives attention, x is $\mathcal{R}_j$'s permanent follower.

It is clear by the way followers are chosen that x is never any other requirement's follower, so x will not enter B unless $\mathcal{R}_j$ acts to put it into B. So if $\mathcal{R}_j$ never requires attention after stage $s+1$, then $x \notin B$, and we never have $\Phi_e^A(x)[t]\downarrow= 0$ for $t > s$, which implies that either $\Phi_e^A(x)\uparrow$ or $\Phi_e^A(x)\downarrow\neq 0$. In either case, $\mathcal{R}_j$ is met.

On the other hand, if $\mathcal{R}_j$ requires attention at a stage $t+1 > s+1$, then $x \in B$, and $\Phi_e^A(x)[t]\downarrow= 0$. The only requirements that put numbers into A after stage $t+1$ are ones weaker than $\mathcal{R}_j$ (i.e., requirements $\mathcal{R}_k$ for $k > j$). Each such strategy is initialized at stage $t+1$, which means that, when it is later appointed a follower, that follower will be bigger than $\varphi_e^A(x)[t]$. Thus no number less than $\varphi_e^A(x)[t]$ will ever enter A after stage $t+1$, which implies, by the use principle, that $\Phi^A(x)\downarrow= \Phi_e^A(x)[t] = 0 \neq B(x)$. Thus, in this case also, $\mathcal{R}_j$ is met. Since $x \in B_{t+2}$ and x is $\mathcal{R}_j$'s permanent follower, $\mathcal{R}_j$ never requires attention after stage $t+1$.

This concludes the induction and hence the proof of the Friedberg-Muchnik Theorem. □

The above proof is an example of the simplest kind of finite injury argument, what is called a *bounded injury* construction. That is, we can put a computable bound *in advance* on the number of times that a given requirement $\mathcal{R}_j$ will be injured. In this case, the bound is $2^j - 1$.

We give another example of this kind of construction, connected with the important concept of lowness.

The halting problem, and hence the jump operator, play a fundamental role in much of computability theory. We know that if $A \leqslant_{\mathrm{T}} B$ then $A' \leqslant_{\mathrm{T}} B'$, and that $A <_{\mathrm{T}} A'$. It is natural to ask what else can be said about the jump operator. The next theorem proves that the jump operator is not injective. Indeed injectivity fails in the first place it can, in the sense that there are sets that the jump operator cannot distinguish from $\emptyset$.

Definition 5.9.2. A set A is *low$_n$* if $A^{(n)} \equiv_{\mathrm{T}} \emptyset^{(n)}$. In particular, low$_1$ sets are called *low*.

A set A is *high$_n$* if $A^{(n)} \equiv_{\mathrm{T}} \emptyset^{(n+1)}$. In particular, high$_1$ sets are called high.

Theorem 5.9.3 (Friedberg)**.** *There is a noncomputable c.e. low set.*

Proof. We construct our set A in stages. To make A noncomputable we need to meet the requirements

$$\mathcal{P}_e : \ \overline{A} \neq W_e.$$

To make A low we meet the requirements

$$\mathcal{N}_e : \ \forall n\,[(\exists^\infty s\, \Phi_e^A(n)[s]\downarrow) \implies \Phi^A(n)\downarrow].$$

To see that such requirements suffice, suppose they are met and define the computable binary function g by letting $g(x,s) = 1$ if $\Phi_e^A(x)[s]\downarrow$ and

$g(x, s) = 0$ otherwise. Then $g(e) := \lim_s g(e, s)$ is well-defined, and by the limit lemma, $A' = \{e : g(e) = 1\} \leqslant_{\mathrm{T}} \emptyset'$.

The strategy for $\mathcal{P}_e$ is simple. We pick a fresh large follower x, and keep it out of A. If x enters W_e, then we put x into A. We meet $\mathcal{N}_e$ by an equally simple conservation strategy. If we see $\Phi_e^A(n)[s]\downarrow$ then we simply try to ensure that $A \upharpoonright \varphi_e^A(n)[s] = A_s \upharpoonright \varphi_e^A(n)[s]$ by initializing all weaker priority requirements, which forces them to choose fresh large numbers as followers. These numbers will be to big to injure the $\Phi_e^A(n)[s]$ computation after stage s. The priority method sorts the actions of the various strategies out. Since $\mathcal{P}_e$ picks a fresh large follower each time it is initialized, it cannot injure any $\mathcal{N}_j$ for $j < e$. It is easy to see that any $\mathcal{N}_e$ can be injured at most e many times, and that each $\mathcal{P}_e$ is met, since it is initialized at most 2^e many times. □

Actually, the above proof above constructs a noncomputable c.e. set that is *superlow*, where a set A is superlow if $A' \equiv_{\mathrm{tt}} \emptyset'$.

5.10 Finite injury arguments of unbounded type

5.10.1 Sacks' Splitting Theorem

There are priority arguments in which the number of injuries to each requirement, while finite, is not bounded by any computable function. One example is the following proof of Sacks' Splitting Theorem [259]. We write $A = A_0 \sqcup A_1$ to mean that $A = A_0 \cup A_1$ and $A_0 \cap A_1 = \emptyset$.

Theorem 5.10.1 (Sacks [259])**.** *Let A be a noncomputable c.e. set. Then there exist Turing incomparable c.e. sets A_0 and A_1 such that $A = A_0 \sqcup A_1$.*

Proof. We build $A_i = \bigcup A_{i,s}$ in stages by a priority argument to meet the following requirements for all $e \in \mathbb{N}$ and $i = 0, 1$.

$$\mathcal{R}_{e,i} : \ \Phi_e^{A_i} \neq A.$$

These requirements suffice because, if $A_{1-i} \leqslant_{\mathrm{T}} A_i$, then $A \leqslant_{\mathrm{T}} A_i$.

Without loss of generality, we will assume that we are given an enumeration of A so that exactly one number enters A at each stage. At each stage we must put this number $x \in A_{s+1} - A_s$ into exactly one of A_0 or A_1, to ensure that $A = A_0 \sqcup A_1$.

To meet $\mathcal{R}_{e,i}$, we define the *length of agreement function*

$$l^i(e, s) := \max\{n : \forall k < n\, (\Phi_e^{A_i}(k)[s] = A(k)[s])\},$$

and the *maximum length of agreement function*

$$m^i(e, s) := \max\{l^i(e, t) : t \leqslant s\},$$

which can be thought of as a high water mark for the length of agreements seen so far. Associated with this maximum length of agreement function is a use function

$$u(e, i, s) := \max\{\varphi_e^{A_i}(l^i(e, t) - 1)[t] : t \leqslant s\},$$

using the convention that use functions are monotone increasing where defined.

The main idea of the proof is perhaps initially counterintutive. Let us consider a single requirement $\mathcal{R}_{e,i}$ in isolation. At each stage s, although we want $\Phi_e^{A_i} \neq A$, instead of trying to destroy the agreement between $\Phi_e^{A_i}(k)[s]$ and $A(k)[s]$ represented by $l^i(e, s)$, we try to *preserve* it. (A method sometimes called *Sacks' method of preserving agreements*.) The way we do this preservation is to put numbers entering $A \upharpoonright u(e, i, s)$ after stage s into A_{1-i} and *not* into A_i. By the use principle, since this action freezes the A_i side of the computations involved in the definition of $l^i(e, s)$, it ensures that for all $k < l^i(e, s)$, we have $\Phi_e^{A_i}(k) = \Phi_e^{A_i}(k)[s]$.

Now suppose that $\liminf_s l^i(e, s) = \infty$, so for each k we can find infinitely many stages s at which $k < l^i(e, s)$. For each such stage, $\Phi_e^{A_i}(k) = \Phi_e^{A_i}(k)[s] = A(k)[s]$. Thus $A(k) = A(k)[s]$ for any such s. So we can compute $A(k)$ simply by finding such an s, which contradicts the noncomputability of A. Thus $\liminf_s l^i(e, s) < \infty$, which clearly implies that $\mathcal{R}_{e,i}$ is met.

In the full construction, of course, we have competing requirements, which we sort out by using priorities. That is, we establish a priority list of our requirements (for instance, saying that $\mathcal{R}_{e,i}$ is stronger than $\mathcal{R}_{e',i'}$ iff $\langle e, i\rangle < \langle e', i'\rangle$). At stage s, for the single element x_s entering A at stage s, we find the strongest priority $\mathcal{R}_{e,i}$ with $\langle e, i\rangle < s$ such that $x_s < u(e, i, s)$ and put x_s into A_{1-i}. We say that $\mathcal{R}_{e,i}$ *acts* at stage s. (If there is no such requirement, then we put x_s into A_0.)

To verify that this construction works, we argue by induction that each requirement eventually stops acting and is met. Suppose that all requirements stronger than $\mathcal{R}_{e,i}$ eventually stop acting, say by a stage $s > \langle e, i\rangle$. At any stage $t > s$, if $x_t < u(e, i, t)$, then x_t is put into A_{1-i}. The same argument as in the one requirement case now shows that if $\liminf_s l^i(e, s) = \infty$ then A is computable, so $\liminf_s l^i(e, s) < \infty$, which implies that $\mathcal{R}_{e,i}$ eventually stops acting and is met. □

In the above construction, injury to a requirement $\mathcal{R}_{e,i}$ happens whenever $x_s < u(e, i, s)$ but x_s is nonetheless put into A_i, at the behest of a stronger priority requirement. How often $\mathcal{R}_{e,i}$ is injured depends on the lengths of agreement attached to stronger priority requirements, and thus cannot be computably bounded.

Note that, for any noncomputable c.e. set C, we can easily add requirements of the form $\Phi_e^{A_i} \neq C$ to the above construction, satisfying them in the same way that we did for the $\mathcal{R}_{e,i}$. Thus, as shown by Sacks [259], in addition to making $A_0 \mid_{\mathrm{T}} A_1$, we can also ensure that $A_i \not\geqslant_{\mathrm{T}} C$ for $i = 0, 1$.

Note also that the computable enumerability of A is not crucial in the above argument. Indeed, a similar argument works for any set A that has a computable approximation, that is, any Δ_2^0 set A. Such an argument shows that if A and C are noncomputable Δ_2^0 sets, then there exist Turing incomparable Δ_2^0 sets A_0 and A_1 such that $A = A_0 \sqcup A_1$ and $A_i \not\geqslant_{\mathrm{T}} C$ for $i = 0, 1$. Readers not familiar with priority arguments involving Δ_2^0 sets are advised to check that the details of the proof still work in this case.

5.10.2 The Pseudo-Jump Theorem

Another basic construction which also uses the finite injury method was discovered by Jockusch and Shore [**?**]. It involves what are called *pseudo-jump operators.*

Definition 5.10.2 (Jockusch and Shore [**?**])**.** For an index e, we define the pseudo-jump operator $V_e^A = A \oplus W_e^A$, for all oracles A. We say that V_e is *nontrivial* iff for all oracles A, $A <_T V_A$.

One example of such a nontrivial pseudo-jump operator is the jump operator itself. The name pseudo-jump mainly comes from the following attractive theorem.

Theorem 5.10.3 (Jockusch and Shore [**?**])**.** *For any nontrivial psuedo-jump operator V, there is a noncomputable c.e. set A with $V^A \equiv_T \emptyset'$.*

Proof. We build $A = \cup_s A_s$ in stages, and work with approximations $V^A[s]$ where $V^A = A \oplus W^A$. We shall build a reduction $\Gamma^{V^A}(2x) = \emptyset'(x)$ with use $\gamma(2x, s)$. The rules here must be the usual reduction ones. That is

(1) If $q \leqslant y$ then $\gamma(q, s) \leqslant \gamma(y, s)$.

(ii) If x enters $\emptyset'$ after stage s, (so that at stage s we have $\Gamma^{V^A}(2x) = 0[s]$) then $V^A \upharpoonright \gamma(2x, s)[s] \neq V^A \upharpoonright gamma(2x, s)$.

(iii) If $\gamma(q, s+1) \neq \gamma(q, s)$ then some $z \leqslant \gamma(q, s)$ must enter $A_{s+1} - A_s$ or $W^A[s+1] \neq W^A[s]$.

The reader should be aware that it is (iii) above which potentially causes some grief in the below. That is, since W^A is not a computably enumerable set, it *could* be that there are stages $s_0 < s_1 < s_2$ with $W^A \upharpoonright \gamma(q, s_0)[s_0] = W^A \upharpoonright \gamma(q, s_0)[s_2]$, *yet* $W^A \upharpoonright \gamma(q, s_0)[s_0] \neq W^A \upharpoonright \gamma(q, s_0)[s_1]$. A number z could enter $W^A[s_0+1] - W^A[s_0]$ below $\gamma(q, s_0)$, say, with some W^A axiom saying that "$z \in W^A$ iff $A \upharpoonright p = A_{s_0+1} \upharpoonright p$". (Here the convention is that $p < s_0$.) At stage s_1 it might be that $A_{s_0+1} \upharpoonright p \neq A_{s_1} \upharpoonright p$, and hence z might leave $W^A[s_1]$. Thus, $\gamma(q, s)$ is potentially not monotone in the second variable. *However, the thing that save us from this is the following. We are the ones building A. Thus if we move $\gamma(q, s_0+1) > s_0$ at stage s_0 (as we do below), then because we will be preserving A to protect this, the* *only numbers to enter $A - A_{s_0+1}$ below s_0 will be numbers below $\gamma(q, s_0)$ and hence we can permantly move $\gamma(q, s_0+1) > s_0$.* We must meet the requirements

$$\mathrm{R}_e \overline{W_e} \neq A.$$

$\mathrm{N}_n^A : (\exists^\infty s)(n \in V^A[s]) \Longrightarrow n \in V^A$,
$\mathrm{P}_n^A : n \in \emptyset'$ if and only if $\gamma^A(n) \in A$,
The requirements and the construction The R_e requirements are met by a standard Feiedberg argument. We will wait till $\gamma(2j+1, s) \in W_{e,s}$ for some $j \geqslant e$, which is unrestrained in that $\gamma(2j+1, s) > \max\{r(k, s) : k \leqslant e\}$, and put $\gamma(2e+1, s) \in A_{s+1} - A_s$.

If x enters $\emptyset'[s]$ then we put $\gamma(2x, s)$ into $A_{s+1} - A_s$.

It is the negative N_n^A requirements which cause the problems. They will generate the $r(n, s)$ restraints in question. They have two actions. First if n occurs in $V^A[s]$ then they restrain this fact with a restraint $r(n, s) = s$, for stages $t \geqslant s+1$ while $n \in V^A[t]$. This restraint will be obeyed by all R_e for $e \leqslant n$. Second, if $n \in V^A[s]$, and $V^A[s+1] \neq V^A[s']$ for all $s' \leqslant s$, then N_n^A will move $\gamma(n, s) \geqslant s$. *End of Construction*

Notice that if this is caused by an $A-$change, then the movement makes $\gamma(n, t) \geqslant s$ for all $t \geqslant s$. If the movement is caused by a W^A-change, then perhaps we might potentially have to move it back, as discussed above. But, by the restraints, the only numbers to enter A at stage t, say, after s below $\gamma(n, s+1)$ will be of the form $\gamma(q, t)$ and hence will by necessity have $q \leqslant n$, and hence $\gamma(q, t) = \gamma(q, s) < \gamma(n, s)$. It is thus easy to show that $\lim_s \gamma(n, s)$ exists, and all requirements are met. □

Jockusch and Shore [**?**, **?**] iused this methodology to establish a number of results. One application is a finite injury proof that there is a high c.e. incomplete Turing degree, a result first proven using the infinite injurt method, as we see in Section **??**. This is proven by taking the e from the original FRiedberg theorem that there is a set W_e noncomputable and of low (even superlow) Turing degree, and relativizing to have an operator V_e such that $Y <_T W_e^Y$ for all Y, V_e^Y is (super-)low over Y, and use Theorem 5.10.3 to construct a c.e. A with $V_e^A \equiv_T \emptyset'$. Then $\emptyset'$ is (super-)low over A and hence A is (super-)high. We will later use this technique when we look at Martin-Löf lowness. We refer the reader to [**?**, **?**] Downey and Shore [**?**], and Coles, Downey, Jockusch and LaForte [**?**] for more on the general theory of pseudo-jump operators.

5.11 The infinite injury method

In this section, we introduce the infinite injury method. We begin by discussing the concept of priority trees, then give a few examples of constructions involving such trees. See Soare [280] for further examples.

5.11.1 Priority trees and guessing

The Friedberg-Muchnik construction used to prove Theorem 5.9.1 can be viewed another way. Instead of having a single strategy for each requirement, which is restarted every time a stronger priority requirement acts, we can attach multiple versions of each requirement to a *priority tree*, in this case the full binary tree $2^{<\omega}$.

Recall that the relevant requirements are the following.

$$\mathcal{R}_{2e} : \Phi_e^A \neq B.$$
$$\mathcal{R}_{2e+1} : \Phi_e^B \neq A.$$

Attached to each node σ is a "version" R_σ of $\mathcal{R}_{|\sigma|}$. We call such a version a *strategy* for $\mathcal{R}_{|\sigma|}$, and refer to σ itself as a *guess*, for reasons that will soon become clear. Thus there are 2^e separate strategies for $\mathcal{R}_e$.

Each strategy R_σ has two *outcomes*, 0 and 1. The outcome 1 indicates that R_σ appoints a follower but never enumerates this follower into its target set. The outcome 0 indicates that R_σ actually enumerates its follower into its target set. At a stage s, we have a guess as to which outcome is correct, depending on whether or not R_σ has enumerated its follower into its target set. If this guess is ever 0, then we immediately know that 0 is indeed the correct outcome, so we regard the 0 outcome as being stronger than the 1 outcome. We order our guesses lexicographically, and hence if σ is to the left of τ, then we think of σ as being stronger than τ.

Figure 5.1 shows the above setup. For comparison, it also shows the setup for the Minimal Pair Theorem, which will be discussed in the next section.

Here is how this mechanism is used. Consider $\mathcal{R}_1$. We have two strategies for this requirement. One, which in this paragraph we write as $R_{\langle 0\rangle}$ rather than R_0 to avoid confusion with $\mathcal{R}_0$, is below the 0 outcome of R_λ and the other, $R_{\langle 1\rangle}$, is below the 1 outcome of R_λ. We think of these versions as guessing that R_λ's outcome is 0 and 1, respectively. The strategy $R_{\langle 1\rangle}$ believes that R_λ will appoint a follower, but will never enumerate it into its target set. Thus its action is simply to act immediately after its guess appears correct, that is, once R_λ appoint its follower. At this point, $R_{\langle 1\rangle}$ can appoint its own follower, and proceed to act as $\mathcal{R}_1$ would have acted in the original proof of the Friedberg-Muchnik Theorem. Its belief may turn out to be false (i.e., R_λ may enumerate its follower into its target set), in which case its action may be negated, but in that case we do not care about $R_{\langle 1\rangle}$. Indeed, in general, we only care about strategies whose guesses

Figure 5.1. The Assignment of Priorities and Outcomes

about the outcomes of strategies above them in the tree are correct. So we have the "back-up" strategy $R_{\langle 0\rangle}$, which believes that R_λ will enumerate its follower into its target set. This strategy will act only once its guess appears correct, i.e., after R_λ enumerates its follower into its target set. Then and only then does this strategy wake up and appoint its follower.

The above is extended inductively on the tree of strategies in a natural way. For instance, for $\sigma = 0110$, there is a strategy R_σ for $\mathcal{R}_4$, which guesses that R_λ and R_{011} do not enumerate their followers, but R_0 and R_{01} do. This strategy waits for R_λ and R_{011} to appoint followers, and for R_0 and R_{01} to both appoint followers and then enumerate then. Once all of these conditions are met, it appoints its own follower and proceeds to act as $\mathcal{R}_4$ would have acted in the original proof of the Friedberg-Muchnik Theorem.

More precisely, at stage s, we have a guess σ_s of length s, which is defined inductively. Only those strategies R_τ with $\tau \preccurlyeq \sigma_s$ get to act at stage s. We say that such τ are *visited* at stage s. Suppose we have defined $\sigma_s \upharpoonright n$. Let $\tau = \sigma_s \upharpoonright n$. Then we allow R_τ to act as follows. Let us suppose that $n = 2e$, the odd case being symmetric. If R_τ has previously put a number into B, then it does nothing. In this case, if $n < s$ then $\sigma(n) = 0$. Otherwise, R_τ acts in one of three ways.

1. If R_τ does not currently have a follower, then it appoints a follower x_τ greater than any number seen in the construction so far. In this case, if $n < s$ then $\sigma_s(n) = 1$.

2. Otherwise, if $\Phi_e^A(x_\tau)[s]\downarrow= 0 = B_s(x_\tau)$, then R_τ enumerates x_τ into B. In this case, if $n < s$ then $\sigma_s(n) = 0$.

3. Otherwise, R_τ does nothing. In this case, if $n < s$ then $\sigma_s(n) = 1$.

The strategy R_λ has a true outcome i, which is 0 if it eventually enumerates its witness into B, and 1 otherwise. Similarly, R_i has a true outcome. The *true path* TP of the construction is the unique path such that $\mathrm{TP}(n)$ is the true outcome of $R_{\mathrm{TP}\upharpoonright n}$. Note that, in the above construction, $\mathrm{TP}(n) = \lim_s \sigma_s(n)$, but with an eye to more complicated tree constructions, we in general define the true path of such a construction as the leftmost path visited infinitely often (that is, the path extending those guesses τ such that τ is the leftmost guess of length $|\tau|$ that is visited infinitely often). The idea behind this definition is that the strategies on the true path are the ones that actually succeed in meeting their respective requirements.

In the construction described above, it is easy to verify that this is indeed the case. Let $\tau \in \mathrm{TP}$. Again, let us assume that $|\tau| = 2e$, the odd case being symmetric. At the first stage s at which τ is visited, a follower x_τ is appointed. Since followers are always chosen to be fresh large numbers, x_τ will not be enumerated into B by any strategy other than R_τ. So if $\Phi_e^A(x_\tau)\uparrow$ or $\Phi_e^A(x_\tau)\downarrow= 1$, then $\mathcal{R}_{|\tau|}$ is met. Otherwise, there is a stage $t > s$ such that $\Phi_e^A(x_\tau)[t]\downarrow= 0 = B_t(x_\tau)$ and τ is visited at stage t. Then R_τ enumerates x_τ into B.

Nodes to the right of $\tau 0$ will never again be visited, so the corresponding strategies will never again act. Nodes to the left of τ are never visited at all, so the corresponding strategies never act. Nodes extending $\tau 0$ will not have been visited before stage t, so the corresponding strategies will pick followers greater than $\varphi_e^A(x_\tau)[t]$. If $n < |\tau|$ then there are two possibilities. If $\tau(n) = 1$ then 1 is the true outcome of $R_{\tau\upharpoonright n}$, and hence that strategy never enumerates its witness. If $\tau(n) = 0$ then $R_{\tau\upharpoonright n}$ must have already enumerated its witness by stage s, since otherwise τ would not have been visited at that stage. In either case, we see that strategies corresponding to nodes extended by τ do not enumerate numbers during or after stage t.

The upshot of the analysis in the previous paragraph is that no strategy enumerates a number into $A \restriction \varphi_e^A(x_\tau)[t]$ during or after stage t, so by the use principle, $\Phi_e^A(x_\tau)\downarrow = 0 \neq B(x_\tau)$, and hence $\mathcal{R}_\tau$ is met.

It is natural to wonder why we should introduce all this machinery. For the Friedberg-Muchnik Theorem itself, as well as for most finite injury arguments, the payoff in new insight provided by this reorganization of the construction does not really balance the additional notational and conceptual burden. However, the above concepts were developed in the last thirty years to make *infinite* injury arguments comprehensible.

The key difference between finite injury and infinite injury arguments is the following. In an infinite injury argument, the action of a given requirements $\mathcal{R}$ may be infinitary. Obviously, we cannot simply restart weaker priority requirements every time $\mathcal{R}$ acts. Instead, we can have multiple strategies for weaker priority requirements, depending on whether or not $\mathcal{R}$ acts infinitely often. Thus, in a basic setup of this sort, the left outcome of $\mathcal{R}$, representing the guess that $\mathcal{R}$ acts infinitely often, is visited each time $\mathcal{R}$ acts.

In the Friedberg-Muchnik argument, the approximation to the true path moves only to the left as time goes on, since the guessed outcome of a strategy can change from 1 to 0, but never the other way. Thus, the true path is computable in $\emptyset'$. In infinite injury arguments, the approximation to the true path can move both left and right. Assuming that the tree is finitely branching, this possibility means that, in general, the true path is computable only in $\emptyset''$. (It clearly is computable in $\emptyset''$ because, letting TP_s be the unique string of length s visited at stage s,

$$\sigma \prec \mathrm{TP} \text{ iff } \exists^\infty s\,(\sigma \prec \mathrm{TP}_s) \wedge \exists^{<\infty} s\,(\mathrm{TP}_s <_{\mathrm{lex}} \sigma)).$$

In the following sections, we will give examples of infinite injury priority tree constructions to illustrate the application of this mechanism.

5.11.2 The minimal pair method

Our first example of infinite injury argument is the construction of a minimal pair of c.e. degrees.

Theorem 5.11.1 (Lachlan[166], Yates [323]). *There exist noncomputable c.e. sets A and B such that every set computable in both A and B is computable. The degrees of such sets are said to form a* minimal pair.

Proof. We construct A and B in stages to satisfy the following requirements for each $e \in \mathbb{N}$.[5]

$$\mathcal{R}_e : \overline{A} \neq W_e.$$
$$\mathcal{Q}_e : \overline{B} \neq W_e.$$
$$\mathcal{N}_e : \Phi_e^A = \Phi_e^B \text{ total} \implies \Phi_e^A \text{ computable.}$$

We arrange these in a priority list as in previous constructions.

We meet the $\mathcal{R}$- and $\mathcal{Q}$-requirements by a Friedberg-Muchnik type strategy. For instance, to satisfy $\mathcal{R}_e$, we pick a follower x, targeted for A, and wait until x enters W_e. If this event never happens, then $x \in \overline{A} \cap \overline{W_e}$, so $\mathcal{R}_e$ is met. If x does enter W_e, then we put x into A, thus again meeting $\mathcal{R}_e$.

The $\mathcal{N}$-requirements are trickier to meet. We first discuss how to meet a single $\mathcal{N}_e$ in isolation, and then look at the coherence problems between the various requirements and the solution to these provided by the use of a tree of strategies.

We follow standard conventions, in particular assuming that all uses at stage s are bounded by s. As in the proof of the Sacks Splitting Theorem (Theorem 5.10.1), we have a length of agreement function

$$l(e,s) := \max\{n : \forall k < n\, (\Phi_e^A(k)[s]\downarrow = \Phi_e^B(k)[s]\downarrow)\},$$

a maximum length of agreement function

$$m(e,s) := \max\{l(e,t) : t \leqslant s\},$$

and an associated use function

$$u(e,s) := \max\{\varphi_e^A(l(e,t)-1)[t],\ \varphi_e^B(l(e,t)-1)[t] : t \leqslant s\},$$

using the convention that use functions are monotone increasing where defined. We say that a stage s is *e-expansionary* if $l(e,s) > m(e,s-1)$, that is, if the current length of agreement surpasses the previous high water mark.

The key idea for meeting $\mathcal{N}_e$ is the following. Suppose that $n < l(e,s)$, so that $\Phi_e^A(n)[s]\downarrow = \Phi_e^B(n)[s]\downarrow$. Now suppose that we allow numbers to enter A at will, but "freeze" the computation $\Phi_e^B(n)[s]$ by not allowing

[5] Clearly, meeting the $\mathcal{R}$- and $\mathcal{Q}$-requirements is enough to ensure that A and B are noncomputable. As for the $\mathcal{N}$-requirements, it might seem at first glance that we need stronger requirements, of the form $\Phi_i^A = \Phi_j^B$ total $\implies \Phi_i^A$ computable. However, suppose that we are able to meet the $\mathcal{N}$-requirements. Then we cannot have $A = B$, since the $\mathcal{N}$-requirements would then force A and B to be computable. So there is an n such that $A(n) \neq B(n)$. Let us assume without loss of generality that $n \in B$. If $f \leqslant_{\mathrm{T}} A, B$ then there are i and j such that $\Phi_i^A = \Phi_j^B = f$. But also, there is an e such that, for all oracles X, we have $\Phi_e^X = \Phi_i^X$ if $n \notin X$ and $\Phi_e^X = \Phi_j^X$ if $n \in X$. For this e, we have $\Phi_e^A = \Phi_e^B = f$. So if such an f is total, then $\mathcal{N}_e$ ensures that it is computable. This argument is known as *Posner's trick*. It is, of course, merely a notational convenience.

any number to enter B below $\varphi_e^B(n)[s]$ until the next e-expansionary stage $t > s$. At this stage t, we again have $\Phi_e^A(n)[t]\downarrow = \Phi_e^B(n)[t]\downarrow$. But we also have $\Phi_e^B(n)[t] = \Phi_e^B(n)[s]$, by the use principle. Now suppose that we start allowing numbers to enter B at will, but freeze the computation $\Phi_e^A(n)[t]$ by not allowing any number to enter A below $\varphi_e^A(n)[t]$ until the next e-expansionary stage $u > t$. Then again $\Phi_e^A(n)[u]\downarrow = \Phi_e^B(n)[u]\downarrow$, but also $\Phi_e^A(n)[u] = \Phi_e^A(n)[t] = \Phi_e^B(n)[t] = \Phi_e^B(n)[s] = \Phi_e^A(n)[s]$. (Note that we are not saying that these *computations* are the same, only that they have the same *value*. It may well be that $\varphi_e^A(n)[u] > \varphi_e^A(n)[s]$, for example.) So if we keep to this strategy, alternately freezing the A-side and the B-side of our agreeing computations, then we ensure that, if $\Phi_e^A = \Phi_e^B$ is total (which implies that there are infinitely many e-expansionary stages), then $\Phi_e^A(n) = \Phi_e^A(n)[s]$. So if we follow this strategy for all n, then we can compute Φ_e^A, and hence $\mathcal{N}_e$ is met.

Figure 5.2 illustrates the above idea.

In summary, the strategy for meeting $\mathcal{N}_e$ is to wait until an e-expansionary stage s, then impose a *restraint* $u(e,s)$ on A, then wait until the next e-expansionary stage t, lift the restraint on A, and impose a restraint $u(e,t)$ on B, and then continue in this way, alternating which set is restrained at each new e-expansionary stage. Note that there is an important difference between this strategy and the one employed in the proof of the Sacks Splitting Theorem. There we argued that the length of agreement associated with a given requirement could not go to infinity. Here, however, it may well be the case that $\lim_s l(e,s) = \infty$, and hence $\lim_s u(e,s) = \infty$. Thus, the restraints imposed by the strategy for $\mathcal{N}_e$ may tend to infinity.

How do the $\mathcal{R}$- and $\mathcal{Q}$-requirements of weaker priority deal with this possibility? Consider a requirement $\mathcal{R}_i$ weaker than $\mathcal{N}_e$. We have two strategies for $\mathcal{R}_i$. One strategy guesses that there are only finitely many e-expansionary stages. This strategy picks a fresh large follower. Each time a new e-expansionary stage occurs, the strategy is *initialized*, which means that it must pick a new fresh large follower. Otherwise, the strategy acts exactly as described above. That is, it waits until its current follower x enters W_i, if ever, then puts x into A.

The other strategy for $\mathcal{R}_i$ guesses that there are infinitely many e-expansionary stages. It picks a follower x. If x enters W_i at stage s, then it wants to put x into A, but it may be restrained from doing so by the strategy for $\mathcal{N}_e$. However, if its guess is correct then there will be an e-expansionary stage $t \geqslant s$ at which the restraint on A is dropped. At that stage, x can be put into A.

Coherence. When we consider multiple $\mathcal{N}$-requirements, we run into a problem. Consider two requirements $\mathcal{N}_e$ and $\mathcal{N}_i$, with the first having stronger priority. Let $s_0 < s_1 < \cdots$ be the e-expansionary stages and $t_0 < t_1 < \cdots$ be the i-expansionary stages. During the interval $[s_0, s_1)$, weaker strategies are prevented from putting certain numbers into A by $\mathcal{N}_e$. At stage s_1, this restraint is lifted, but if we have $t_0 \leqslant s_1 < t_1$, then at

A_s

$\Phi_{i,s}$ Stage s

f is (i,j)-expansionary

$\Phi_{j,s}$

B_s

x enters

A_{s+1}

$\Phi_{i,s}$ is destroyed Stage $s+1$.

f x enters A

destroying the A-side.

B_{s+1}

no change

$A_{s'}$

Stage $t > s' \geqslant s+1$.

f Before recovery we

protect the B-side.

$B_{s'}$

restrained

A_t Stage t. Recovery.

$\Phi_{i,t}$ (now)

f_s t is (i,j) expansionary

Φ_i^A has recovered.

B_t

Figure 5.2. Stage s and stage t

stage s_1 it will be $\mathcal{N}_i$ that prevents weaker strategies from putting certain numbers into A. When $\mathcal{N}_i$ drops that restraint at stage t_1, we might have a new restraint on A imposed by $\mathcal{N}_e$ (if say $s_2 \leqslant t_1 < s_3$). Thus, although individually $\mathcal{N}_e$ and $\mathcal{N}_i$ each provide infinitely many stages at which they allow weaker strategies to put numbers into A, the two strategies *together* might conspire to block weaker strategies from *ever* putting numbers into A.

This problem is overcome by having two strategies for $\mathcal{N}_i$. The one guessing that there are only finitely many e-expansionary stages has no problems. On the other hand, the one guessing that there infinitely many e-expansionary stages acts only at such stages, and in particular calculates its expansionary stages based only on such stages, which forces its expansionary stages to be nested within the e-expansionary stages. Thus the actions of $\mathcal{N}_e$ and $\mathcal{N}_i$ are forced to cohere.

We now turn to the formal details of the construction. Note that the details of the construction below may at first appear somewhat mysterious, in that it may be unclear how they actually implement the strategies described above. However, the verification section should clarify things.

The Priority Tree. We use the tree $\mathcal{T} = \{\infty, f\}^{<\omega}$. (Of course, this tree is the same as $2^{<\omega}$, but renaming the nodes should help in understanding the construction.) To each $\sigma \in \mathcal{T}$, we assign strategies N_σ, R_σ, and Q_σ for $\mathcal{N}_{|\sigma|}$, $\mathcal{R}_{|\sigma|}$, and $\mathcal{Q}_{|\sigma|}$, respectively. (It is only the $\mathcal{N}$-requirements that really need to be on the tree, since only they have requirements about which we care, but this assignment is notationally convenient.) (See Figure 5.1.) Again we use lexicographical ordering, with ∞ to the left of f.

Definition 5.11.2. We define the notions of *σ-stage*, $m(\sigma, s)$, and *σ-expansionary stage* by induction on $|\sigma|$.

(i) Every stage is a λ-stage.

(ii) Suppose that s is a τ-stage. Let $e = |\tau|$. Let

$$m(\tau, s) = \max\{l(e, t) : t < s \text{ is a } \tau\text{-stage}\}.$$

If $l(e, s) > m(\tau, s)$ then we say that s is τ-expansionary and declare s to be a $\tau^\frown\infty$-stage. Otherwise, we declare s to be a $\tau^\frown f$-stage.

Let TP_s be the unique σ of length s such that s is a σ-stage.

Definition 5.11.3. We say that R_σ *requires attention* at a σ-stage $s > |\sigma|$ if $W_{|\sigma|}[s] \cap A_s = \emptyset$ and one of the following holds.

(i) R_σ currently has no follower.

(ii) R_σ has a follower $x \in W_{|\sigma|}[s]$.

The definition of Q_σ requiring attention is analogous.

Construction. At stage s, proceed as follows.
Step 1. Compute TP_s. Initialize all strategies attached to nodes to the right of TP_s. For the R- and Q-strategies, this initialization means that their followers are canceled.
Step 2. Find the strongest priority $\mathcal{R}$- or $\mathcal{Q}$-requirement that has a strategy requiring attention at stage s. Let us suppose that this requirement is $\mathcal{R}_e$ (the $\mathcal{Q}$-requirement case being analogous), and that R_σ is the corresponding strategy requiring attention at stage s. (Note that s must be a σ-stage.) We say that R_σ *acts* at stage s. Initialize all strategies attached to nodes properly extending σ.

If R_σ does not currently have a follower, appoint a fresh large follower for R_σ. Otherwise, enumerate R_σ's follower into A.
End of Construction.

Verification. Let the true path TP be the leftmost path through $\mathcal{T}$ visited infinitely often. In other words, let TP be the unique path through $\mathcal{T}$ such that

$$\sigma \prec \mathrm{TP} \text{ iff } \exists^\infty s\,(\sigma \prec \mathrm{TP}_s) \wedge \exists^{<\infty} s\,(\mathrm{TP}_s <_{\mathrm{lex}} \sigma).$$

We write $\tau \leqslant \mathrm{TP}$ to mean that τ is either on or to the left of the true path.

Lemma 5.11.4. *Each $\mathcal{R}$- and $\mathcal{Q}$-requirement is met.*

Proof. Consider $\mathcal{R}_e$ (the $\mathcal{Q}$-requirement case being analogous). Let $\sigma = \mathrm{TP} \upharpoonright e$. Assume by induction that the strategies attached to proper prefixes of σ act only finitely often, and let s be the least stage such that

1. no strategy attached to a proper prefix of σ acts after stage s and
2. TP_t is not to the left of σ for all $t > s$.

By the minimality of s, it must be the case that R_σ is initialized at stage s. Thus, at the next σ-stage $t > s$, it will be assigned a follower x. Since R_σ cannot be initialized after stage t, this follower is permanent. By the way followers are chosen, x will not be put into A unless x enters W_e. If x enters W_e at stage $u > t$, then at the first σ-stage $v \geqslant u$, the strategy R_σ will act, and x will enter A. In any case, R_σ succeeds in meeting $\mathcal{R}_e$. Note that R_σ acts at most twice after stage s, and hence the induction can continue. □

Lemma 5.11.5. *Each $\mathcal{N}$-requirement is met.*

Proof. Consider $\mathcal{N}_e$. Let $\sigma = \mathrm{TP} \upharpoonright e$. If $\sigma^\frown f \prec TP$ then there are only finitely many e-expansionary stages, so $\Phi_e(A) \neq \Phi_e(B)$, and hence $\mathcal{N}_e$ is met.

So suppose that $\sigma^\frown\infty \prec TP$ and that Φ_e^A is total. We will show that Φ_e^A is computable. Let s be the least stage such that

1. no strategy attached to a prefix of σ (including σ itself) acts after stage s and
2. TP_t is not to the left of σ for all $t > s$.

To compute $\Phi_e^A(n)$, find the least $\sigma\widehat{\ }\infty$-stage $t_0 > s$ such that $l(e,s) > n$. We claim that $\Phi_e^A(n) = \Phi_e^A(n)[t_0]$.

Let $t_0 < t_1 < \cdots$ be the $\sigma\widehat{\ }\infty$-stages greater than or equal to t_0. Each such stage is e-expansionary, so we have $\Phi_e^A(n)[t_i] = \Phi_e^B(n)[t_i]$ for all i. We claim that, for each i, we have either $\Phi_e^A(n)[t_{i+1}] = \Phi_e^A(n)[t_i]$ or $\Phi_e^B(n)[t_{i+1}] = \Phi_e^B(n)[t_i]$. Assuming this claim, it follows easily that $\Phi_e^A(n)[t_0] = \Phi_e^A(n)[t_1] = \cdots$, which implies that $\Phi_e^A(n) = \Phi_e^A(n)[t_0]$.

To establish the claim, fix i. At stage t_i, we initialize all strategies to the right of $\sigma\widehat{\ }\infty$. Thus, any follower of such a strategy appointed after stage t_i must be larger than any number seen in the construction by stage t_i, and in particular larger than $\varphi_e^A(n)[t_i]$ and $\varphi_e^B(n)[t_i]$. By the choice of s, strategies above or to the left of σ do not act after stage s. Thus, the only strategies that can put numbers into $A \upharpoonright \varphi_e^A(n)[t_i]$ or $B \upharpoonright \varphi_e^B(n)[t_i]$ between stages t_i and t_{i+1} are the ones associated with extensions of $\sigma\widehat{\ }\infty$. But such a strategy cannot act except at a $\sigma\widehat{\ }\infty$-stage, and at each stage at most one strategy gets to act. Thus, at most one strategy associated with an extension of $\sigma\widehat{\ }\infty$ can act between stages t_i and t_{i+1}, and hence only one number can enter one of $A \upharpoonright \varphi_e^A(n)[t_i]$ or $B \upharpoonright \varphi_e^B(n)[t_i]$ between stages t_i and t_{i+1}. So either $A \upharpoonright \varphi_e^A(n)[t_{i+1}] = A \upharpoonright \varphi_e^A(n)[t_i]$ or $B \upharpoonright \varphi_e^B(n)[t_{i+1}] = B \upharpoonright \varphi_e^B(n)[t_i]$, which implies that either $\Phi_e^A(n)[t_{i+1}] = \Phi_e^A(n)[t_i]$ or $\Phi_e^B(n)[t_{i+1}] = \Phi_e^B(n)[t_i]$. As mentioned above, this shows that $\Phi_e^A(n) = \Phi_e^A(n)[t_0]$, as desired. □

These two lemmas show that all requirements are met, which concludes the proof of the theorem. □

Readers unfamiliar with the methodology above should test themselves by proving the following result.

Theorem 5.11.6 (Ambos-Spies [4], Downey and Welch [98])**.** *There is a noncomputable c.e. set A such that if $C \sqcup D = A$ is a c.e. splitting of A, then the degrees of C and D form a minimal pair.*

Hint.. The proof is similar to that of Theorem 5.11.1, using the length of agreement function

$$\begin{aligned} l(i,j,k,m,s) = \max\{x : \forall y < x\, [&\Phi_i^{W_k}(y)[s]\downarrow = \Phi_j^{W_m}(y)[s]\downarrow \;\wedge \\ &\forall z \leqslant \varphi_i^{W_k}(y)[s], \varphi_j^{W_m}(y)[s]\, (W_k[s] \upharpoonright z \sqcup W_m[s] \upharpoonright z = A_s \upharpoonright z)]. \end{aligned}$$

□

5.11.3 High computably enumerable degrees

Another example of an infinite injury argument is the construction of an incomplete high c.e. degree. Recall that a degree $\mathbf{a}$ is high if $\mathbf{a}' \geqslant \mathbf{0}''$. We begin with a few definitions.

Definition 5.11.7. Let $A^{[e]} = \{\langle e, n\rangle : \langle e, n\rangle \in A\}$. We call $A^{[e]}$ the *e-th column* of A.

A set A is *piecewise trivial* if for all e, the set $A^{[e]}$ is either finite or equal to $\mathbb{N}^{[e]}$.

A subset B of A is a *thick* subset of A if for all e, we have $A^{[e]} =^* B^{[e]}$ (i.e., the symmetric difference of $A^{[e]}$ and $B^{[e]}$ is finite).

Lemma 5.11.8. (i) *There is a piecewise trivial c.e. set A such that $A^{[e]}$ is infinite iff Φ_e is total.*

(ii) *If B is a thick subset of such a set A, then B is high.*

Proof. (i) Let $A = \{\langle e, n\rangle : e \in \mathbb{N} \wedge \forall k \leqslant n\, \Phi_e(k)\downarrow\}$. Then A is c.e., and clearly $A^{[e]}$ is infinite iff $A^{[e]} = \mathbb{N}^{[e]}$ iff Φ_e is total.

(ii) Define a reduction Γ by letting $\Gamma^X(e, s) = 1$ if $\langle e, s\rangle \in X$ and $\Gamma^X(e, s) = 0$ otherwise. If Φ_e is total then $A^{[e]} = \mathbb{N}^{[e]}$, so $B^{[e]}$ is coinfinite, which implies that $\lim_s \Gamma^B(e, s) = 1$. On the other hand, if Φ_e is not total then $A^{[e]}$ is finite, so $B^{[e]}$ is finite, which implies that $\lim_s \Gamma^B(e, s) = 0$. Thus, the function f defined by $f(e) = \lim_s \Gamma^B(e, s)$ is total. By the relativized form of the limit lemma, $f \leqslant_T B'$. So B' can decide whether a given Φ_e is total or not. By Theorem 5.6.6, $\emptyset'' \leqslant_T B'$. □

Thus, to construct an incomplete high c.e. degree, it suffices to prove the following result, which is a weak form of the Thickness Lemma discussed in the next section.

Theorem 5.11.9 (Shoenfield [272])**.** *Let C be a noncomputable c.e. set, and let A be a piecewise trivial c.e. set. Then there is a c.e. thick subset B of A such that $C \nleqslant_T B$.*

Proof. We construct $B \subseteq A$ to meet the following requirements for all $e \in \mathbb{N}$.

$$\mathcal{R}_e : |A^{[e]} - B^{[e]}| < \infty.$$
$$\mathcal{N}_e : \Phi_e^B \neq C.$$

We give the intuition and formal details of the construction, and sketch out its verification, leaving some details to the reader.

To meet $\mathcal{R}_e$, we must make sure that almost all of the e-th column of A gets into B. To meet $\mathcal{N}_e$, we use the strategy already employed in the proof of Sacks' Splitting Theorem. (Theorem 5.10.1.) That is, we measure the length of agreement

$$l(e, s) := \max\{n : \forall k < n\, \Phi_e^B(k)[s] = C_s(k)\},$$

with the idea of preserving B_s on the use $\varphi_e^B(n)[s]$ for all $n < l(e, s)$.

The problem comes from the interaction of this preservation strategy with the strategies for stronger priority $\mathcal{R}$-requirements, since these may be infinitary. It might be the case that we infinitely often try to preserve an agreeing computation $\Phi_e^B(n)[s] = C_s(n)$, only to have some $\langle i, x\rangle$ enter A with $i < e$ and $\langle i, x\rangle < \varphi_e^B(n)[s]$. Since we must put almost every such pair into B to meet $\mathcal{R}_i$, our strategy for meeting $\mathcal{N}_e$ might be injured infinitely often.

On the other hand, we do know that $\mathcal{R}_i$ can do only one of two things. Either $A^{[i]}$ is finite, and hence $\mathcal{R}_i$ stops injuring $\mathcal{N}_e$ after some time, or almost all of $A^{[i]}$ is put into B. In the latter case, the numbers put into B for the sake of $\mathcal{R}_i$ form a computable set (since A is piecewise trivial). It is easy to adapt the strategy for $\mathcal{N}_e$ to deal with this case. Let S be the computable set of numbers put into B for the sake of $\mathcal{R}_i$. We can then proceed with the Sacks preservation strategy, except that we do not believe a computation $\Phi_e^B(k)[s]$ unless B_s already contains every element of $S \upharpoonright \varphi_e^B(k)[s]$. This modification prevents the action taken for $\mathcal{R}_i$ from ever injuring the strategy for meeting $\mathcal{N}_i$.

Of course, we do not know which of the two possibilities for the action of $\mathcal{R}_i$ actually happens, so, as before, we will have multiple strategies for $\mathcal{N}_e$, representing guesses as to whether or not $A^{[i]}$ is finite. There are several ways to organize this argument as a priority tree construction.[6] We give the details of one such construction.

The N-strategies do not have interesting outcomes, so we use the same tree $\mathcal{T} = \{\infty, f\}^{<\omega}$ as in the proof of the Minimal Pair Theorem (Theorem 5.11.1), with the same lexicographic ordering as before (i.e., the ordering induced by regarding ∞ as being to the left of f). To each $\sigma \in \mathcal{T}$ of length e we associate strategies R_σ and N_σ for meeting $\mathcal{R}_e$ and $\mathcal{N}_e$, respectively.

Definition 5.11.10. We define the notions of σ-stage, σ-believable computation, σ-restraint $r(\sigma, s)$, σ-expansionary stage, and σ-length of agreement $l(\sigma, s)$ by induction on $|\sigma|$ and s as follows.

Every stage is a λ-stage, and $r(\sigma, 0) = 0$ for all σ. As usual, if we do not explicitly define $r(\sigma, s)$ at stage s, then $r(\sigma, s) = r(\sigma, s - 1)$.

Suppose that $s > 0$ is a σ-stage, and let $e = |\sigma|$. A computation $\Phi_e^B(k)[s] = C_s(k)$ is *σ-believable* if for all $\tau\widehat{\ }\infty \preccurlyeq \sigma$ and all x, if $r(\tau, s) <$

[6]It is often the case in an infinite injury construction that we have substantive choices in defining the priority tree. In particular, some authors prefer to encode as much information as possible about the behavior of a strategy into its outomes, while others prefer to encode only the information that weaker strategies need to have. It is worth keeping this fact in mind when reading such constructions.

$\langle|\tau|, x\rangle < \varphi_e^B(k)[s]$, then $\langle|\tau|, x\rangle \in B_s$. Let

$$l(\sigma, s) = \max\{n : \forall k < n\,(\Phi_e^B(k)[s] = C_s(k) \text{ via a } \sigma\text{-believable computation})\}.$$

Say that s is *σ-expansionary* if $l(\sigma, s) > \max_{t<s} l(\sigma, t)$. Let

$$r(\sigma, s) = \max(\{r(\tau, s) : \tau <_{\text{lex}} \sigma\} \cup \{\varphi_e^B(k)[s] : k < l(\sigma, s)\}).$$

Let t be the last σ-stage before s, or 0 if there have been no such stages. If $|A_s^{[e]}| > |A_t^{[e]}|$ then s is a $\sigma\widehat{\ }\infty$-stage. Otherwise, s is a $\sigma\widehat{\ }f$-stage.

Construction. The construction is now rather simple. At stage s, let σ_s be the unique string of length s such that s is a σ_s-stage. For each $e < s$ and each $\langle e, x\rangle \in A_s$ that is not yet in B_s, if $\langle e, x\rangle > r(\sigma_s \upharpoonright i, s)$ then put $\langle e, x\rangle$ into B.
End of Construction.

We now sketch the verification that this construction succeeds in meeting all requirements. As usual, the true path TP of the construction is the leftmost path visited infinitely often.

Let $\sigma \in$ TP, let $e = |\sigma|$ and assume by induction that $\lim_t r(\tau, t)$ is well-defined for all $\tau \prec \sigma$. Let s be a stage such that

1. $r(\tau, t)$ has reached a limit by stage s for every $\tau \prec \sigma$,
2. the construction never moves to the left of σ after stage s,
3. $B \upharpoonright r(\tau) = B_s \upharpoonright r(\tau)$, and
4. $B_s^{[i]} = B^{[i]}$ for all $i < e$ such that $\sigma(i) = f$.

Note that item 4 makes sense because if $\sigma \in$ TP and $\sigma(i) = f$, then there is a τ of length i such that $\tau\widehat{\ }f \in$ TP, which means that $A^{[i]}$ is finite, and hence so is $B^{[i]}$.

It is now not hard to argue, as in the proof of Sacks' Splitting Theorem, that if $t > s$ is a σ-stage and $n < l(\sigma, t)$, then the computation $\Phi_e^B(n)[s]$ is preserved forever. (The key fact is that this computation is σ-believable, and hence cannot be injured by stronger priority strategies.) Again as in the proof of Sacks' Splitting Theorem, it must be the case that $\lim_t l(\sigma, t)$ exists, and hence $\mathcal{N}_e$ is met.

It is also not hard to argue now that $r(\sigma) := \lim_t r(\sigma, t)$ is well-defined. Let u be a stage by which this limit has been reached. Thus every $\langle e, x\rangle > r(\sigma)$ that enters A after stage u is eventually put into B, and hence $\mathcal{R}_e$ is met. $\square$

5.11.4 The Thickness Lemma

Theorem 5.11.9 is only a weak form of the real Thickness Lemma of Shoenfield. To state the full version, we need a new definition.

Definition 5.11.11. A set A is *piecewise computable* if $A^{[e]}$ is computable for each e.

Theorem 5.11.12 (Thickness Lemma, Shoenfield [272])**.** *Let C be a noncomputable c.e. set, and let A be a piecewise computable c.e. set. Then there is a c.e. thick subset B of A such that $C \nleqslant_{\mathrm{T}} B$.*

Proof Sketch. We briefly sketch how to modify the proof of Theorem 5.11.9. Recall that in that result, we assumed that A was piecewise trivial. Now we have the weaker assumption that A is piecewise computable, that is, every column of A is computable. Thus, for each e, there is a c.e. set $W_{g(e)} \subseteq \mathbb{N}^{[e]}$ such that $W_{g(e)}$ is the complement of $A^{[e]}$ in $\mathbb{N}^{[e]}$ (meaning that $W_{g(e)} \sqcup A^{[e]} = \mathbb{N}^{[e]}$).

The key to how we used the piecewise triviality of A in Theorem 5.11.9 was in the definition of σ-believable computation. Recall that for σ with $e = |\sigma|$, we said that a computation $\Phi_e^B(k)[s] = C_s(k)$ at a σ-stage s was σ-believable if for all $\tau\widehat{\ }\infty \preccurlyeq \sigma$ and all x, if $r(\tau, s) < \langle|\tau|, x\rangle < \varphi_e^B(k)[s]$, then $\langle|\tau|, x\rangle \in B_s$. This definition relied on the fact that, if $\tau\widehat{\ }\infty \in \mathrm{TP}$, then every $\langle|\tau|, x\rangle > r(\tau, s)$ was eventually put into B. The corresponding fact here is that every $\langle|\tau|, x\rangle > r(\tau, s)$ that is not in $W_{g(|\tau|)}$ is eventually put into B.

So in order to adjust the definition of σ-believable computation, for each $\tau \prec \sigma$, we need to know an index j such that $W_j = W_{g(|\tau|)}$. Since we cannot compute such a j from $|\tau|$, we must guess it along the tree of strategies. That is, for each τ, instead of the outcomes ∞ and f, we now have an outcome j for each $j \in \mathbb{N}$, representing a guess that $W_j = W_{g(|\tau|)}$. The outcome j is taken to be correct at a stage s if j is the least number for which we see the length of agreement between $\mathbb{N}^{[e]}$ and $A^{[e]} \sqcup W_j$ increase at stage s.

Now a computation $\Phi_e^B(k)[s] = C_s(k)$ at a σ-stage s is σ-believable if for each $\tau\widehat{\ }j \preccurlyeq \sigma$ and all x, if $r(\tau, s) < \langle|\tau|, x\rangle < \varphi_e^B(k)[s]$, then $\langle|\tau|, x\rangle \in B_s \cup W_j$. The rest of the construction is essentially the same as before. $\square$

We make some remarks for the reader who, like the senior author, was brought up with the "old" techniques using the "hat trick" and the "window lemma" of Soare [280]. It is rather ironical that the very first result that used the infinite injury method in its proof was the Thickness Lemma, since, like the Density Theorem of the next section, it has a proof *not* using priority trees that is combinatorially much easier to present. In a sense, this fact shows an inherent shortcoming of the tree technique in that often more information needs to be represented on the tree than is absolutely necessary for the proof of the theorem. To demonstrate this point, and since it is somewhat instructive, we now sketch the original proof of the Thickness Lemma.

We have the same requirements, but we define $\widehat{\Phi}^B(n)[s]$ to be $\Phi^B(n)[s]$ unless some number less than $\varphi^B(n)[s]$ enters B at stage s, in which case we declare that $\widehat{\Phi}^B(n)[s] \uparrow$. This is called the *hat convention*. Using this convention, we can generate the hatted length of agreement $\widehat{l}(e,s)$ with $\widehat{\Phi}$ in place of Φ, and so forth. Finally, we define the *restraint function*

$$\widehat{r}(e,s) = \max\{\widehat{\varphi}^B(n)[s] : n < \widehat{l}(e,s)\}.$$

The construction is to put $\langle e,x\rangle \in A_s$ into B at stage s if it is not yet there and $\langle e,x\rangle > \max\{\widehat{r}(j,s) : j \leqslant e\}$.

To verify that this construction succeeds in meeting all our requirements, it is enough to show that, for each e, if we define the *restraint set*

$$\widehat{I}_{e,s} = \{x : \exists v \leqslant s[x \leqslant \widehat{r}(e,v) \wedge x \in B_{s+1} - B_v\},$$

then the injury set is computable, $B^{(e)} =^* A^{(e)}$, and N_e is met, all by simultaneous induction, the key idea being the "window lemma" . This lemma states that the *lim inf* of the restraint $\widehat{R}(e,s) = \max\{\widehat{r}(j,s) : j \leqslant e\}$, is finite.

The point is that to verify these facts we do not actually need to know in the construction what the complement of $A^{[e]}$ is. This is only used in the verification. We refer the reader to Soare [280, Theorem VIII.1.1] for more details.

There is a strong form of the Thickness Lemma that is implicit in the work of Lachlan, Robinson, Shoenfield, Sacks, Soare and others.

Theorem 5.11.13 (Thickness Lemma-strong form)**.** *Given a set $\emptyset <_{\mathrm{T}} C \leqslant \emptyset'$, and a computable enumerable set A there is a computable enumerable set $B \subseteq A$ such that $B \leqslant_{\mathrm{T}} A$ and*

(i) For all e, $C \nleqslant_{\mathrm{T}} A^{(e)}$ implies ($C \nleqslant_{\mathrm{T}} B$ and B is a thick subset of A), and

(ii) for all e, $C \nleqslant_{\mathrm{T}} A^{(e)}$ implies for all $j \leqslant e$,

$$C \neq \Phi_j^B \wedge B^{(j)} =^* A^{(j)}.$$

Furthermore an index for B can be obtained uniformly from indices for A and C.

Theorem 5.11.13 can be proved with much the same methods as those used to prove the Thickenss Lemma. Soare showed that many results of classical computability theory can be obtained from the Strong Thickness Lemma. For instance, here is an example from [280].

Corollary 5.11.14 (Sacks [259])**.** *Let $\mathbf{a}_0 < \mathbf{a}_1 < \cdots$ be an infinite ascending sequence of uniformly c.e. degrees. Then there is an incomplete c.e. degree $\mathbf{b}$ such that $\mathbf{a}_0 < \mathbf{a}_1 < \cdots < \mathbf{b}$. Thus $\mathbf{0}'$ is not a minimal upper bound for any such sequence of degrees.*

Proof. Let h be a computable function such that $\mathbf{a}_i = \deg(W_{h(i)})$ for all i. Let $A = \{\langle i,n\rangle : n \in W_{h(i)}\}$. Let $C = \emptyset'$ and apply the Strong Thickness

Lemma to get a c.e. set $B \subseteq A$ such that $C \not\leqslant_{\mathrm{T}} B$ and $B^{[i]} =^* A^{[i]}$ (which implies that $\deg(B^{[i]}) = \mathbf{a}_i$) for all i. Let $\mathbf{b} = \deg(B)$. $\square$

5.12 The Density Theorem

One of the classic applications of the infinite injury method is the extension of the Friedberg-Muchnik Theorem showing that not only are there intermediate degrees, but the structure forms a dense partial ordering. It is possible to give a short and elegant proof of this result *not* using trees, but it is quite hard to understand how this proof works. Therefore we in this section, we will give a proof using trees.

Theorem 5.12.1 (Sacks' Density Theorem, Sacks [261])**.** *Given c.e. sets B, C with $B <_{\mathrm{T}} C$, there are c.e. sets A_1, A_2, such that $B \leqslant_{\mathrm{T}} A_1 \oplus B, A_2 \oplus B \leqslant_{\mathrm{T}} C$, $A_1 \oplus B \mid_{\mathrm{T}} A_2 \oplus B$.*

Proof. We will construct A_1, A_2 satisfying the following requirements:

$$\begin{aligned} R_{2e+1}:&\quad \Gamma_e(A_2 \oplus B) \neq A_1, \\ R_{2e}:&\quad \Gamma_e(A_1 \oplus B) \neq A_2. \end{aligned}$$

Additionally, we must meet the global (permitting) requirement:

$$A_1 \oplus B, A_2 \oplus B \leqslant_{\mathrm{T}} C.$$

We must perform three tasks:

1. A Friedberg-Muchnik strategy;
2. permitting 1, that is, showing that the strategy of 1 can be performed below C; and
3. coping with the coding of B and how to modify 2 to deal with this.

We will treat these tasks in order.
The Friedberg(-Muchnik) strategy. Consider the satisfaction of the requirement R_{2e}. The strategy is the following:

1. pick a follower x;
2. wait till $\Gamma_{e,s}(B_s \oplus A_{1,s}, x) \downarrow = 0$;
3. put x into A_2 and preserve $B_s \oplus A_{1,s} \upharpoonright \gamma_{e,s}(x)$.

We refer to the above as the *underlying* Friedberg strategy. It is not correct as stated, but serves as a basis for future modification. The basic problem is that we can't control B, so it could change all these computations. Furthermore we can only put numbers into A_2 when permitted to do so by C. We deal with this latter problem first.
Adding permitting

In place of a single follower x we will use a potentially infinite collection $\{x(e,i,s) : i \in \mathbb{N}\}$ of followers or *coding markers*, attempting to code "$i \in C$"to meet R_{2e}. They are picked inductively as follows.

1. Suppose that we have already have $x(e,i,s)$ for $i = 1, \cdots, n$, and all these have been *realized*. Pick a big, unrealized follower $x(e, n+1, s)$ targeted for A_2.

2. Wait until $\Gamma_{e,t}(B_s \oplus A_{1,t}; x(e, n+1, s)) \downarrow= 0$. Declare $x(e, n+1, s)$ as *realized* and go to 1. When realization occurs, we also will initialize lower priority requirements. (This initialization freezes the potential win (should we ever be able to put $x(e, n+1, s)$ into A_2), since no number below t targeted for A_1 is left alive (with priority R_{2e}) after stage t.) We call this an $x(e, n+1, s)$-*set up* to win R_{2e}.

3. Repeat until C permits a realized follower. That is, wait till i enters C_t, then put $x(e,i,s)$ into A_2 at such a stage t.

Without considering the effect of B, this works since C is not computable. If infinitely many $x(e,i,s)$ were appointed but none permitted, we could compute C. To compute $C(i)$ wait till $x(e,i,s)$ is realized. We remark that this permitting is slightly unusual: do not wait for C to permit $x(e,i,s)$ but simply i.

Coping with the effect of B-coding. Here is a first approximation to the actual strategy we use for coping with B-coding.

First notice that even *a set-up at stage s* can be injured by B changing an $\gamma_{i,s}(x)$ after stage s. To see this, we start by considering the naive approach of beginning anew on x each time it's set up is B-injured. That is, we go back to waiting for waiting until $\Gamma_{e,t}(B_s \oplus A_{1,t}; x(e, n+1, s)) \downarrow= 0$. At such a stage we could try initializing exactly as above.

The effect of this naive approach could even be fatal for a single x. We might make a set up, then B destroys it. We might try again, then B destroys it. Notice that for a fixed x this cycle can repeat infinitely often only if B permits below the use of a fixed x infinitely often.

Of course, in this situation, we actually meet R_{2e}, since $\gamma_{i,s}(x) \to \infty$. *But, the problem is not what happens with R_{2e} but what happens to R_{2j+1} for $j \geqslant e$ since they are all initialized infinitely often. The effect on these low priority requirements is fatal.*

"We have won the R_{2e}-battle, but lost the war."

Our solution to this dilemma is to notice that this ugly situation can only happen if for some $x(e,n,s)$, the Γ-use is unbounded. Now were we to *know* that for some n, the Γ-use of $x(e,n,s)$ is unbounded, then we could do the construction ignoring this requirement. It is met *inter alia.*

Unfortunately, we can only see that the use is unbounded for a fixed n, if it appears unbounded infinitely often. This is an infinite outcome. It can be guessed on the strategy tree.

The trick is to make the effect on the construction of this outcome computable, and hence lower priority R_{2j+1} *guessing* the infinite outcome being witnessed by n will be able to cope with this version.

Here is our modified basic module for R_{2e}.

1. if $x(e, n, s)$ is realized, and n enters C at stage s, put $x(e, n, s)$ into A_2.
2. if $x(e, n, s)$ is realized, and $B_s \upharpoonright \gamma_e(x(e, n, s), s)$ changes (n least), cancel $x(e, m, s)$ for $m > n$. $x(e, n, s)$ becomes unrealized again.
3. if not (a) and (b), and $x(e, n, s)$ is unrealized, and $l(e, s) > x(e, n, s)$, then declare that $x(e, n, s)$ as realized, and pick a big $x(e, n+1, s)$ as the next unrealized follower for R_{2e}.

Lemma 5.12.2. *At any stage s, the followers of R_{2e} look like*

$$x(e, 0, s), \cdots, x(e, n, s)$$

with $x(e, 0, s), \cdots, x(e, n-1, s)$ all realized and $x(e, n, s)$ unrealized. Note that being realized means that its use currently exists.

Lemma 5.12.3. *This module wins one R_{2e}.*

Proof. Suppose not. Then $\Gamma_e(A_1 \oplus B) = A_2$. Consequently, for all n, $\lim_s x(e, n, s) = x(e, n)$ exists. The reason is that we change $x(e, n, s)$ only if the use of $x(e, n-1, s)$ changes.

Moreover, B can compute the final incarnation of $x(e, n, s)$. We only change $x(e, n, s)$ when $B_s \upharpoonright \gamma_e(x(e, j, s), s)$ changes for some $j \leqslant n$. So B knows $x(e, 0)$ $x(e, 0$ never changes. If B goes to a stage where B stops changing on the use of $x(e, 0)$, then it can determine $x(e, 1)$, since now $x(e, 1)$ will never change The result is that B can figure our the final $x(e, i)$ inductively on the assumption that all uses come to rest.

Now we claim that B can compute C. To figure out $C(n)$, B computes a stage s where $x(e, n+1, s)$ is final. Then $n \in C$ iff $n \in C_s$. (Otherwise, n would enter C, permitting $x(e, n, s)$ and winning the requirement.) $\square$

The module above is actually how we act for R_0. Basic outcomes for R_{2e} (we know that $\Gamma_e(A_1 \oplus B) \neq A_2$:

(i, u): $\gamma_e(x(e, i))$ is unbounded;

(i, f): $l(e, s)$ stops growing, with witness $x(e, i)$.

We use these outcomes to generate the *Priority Tree* $\mathcal{T} = \{(i, u), (i, f) : i \in \omega\}^{<\omega}$ with $(i-1, f) <_L (i, u) <_L (i, f)$.

Now we discuss how lower priority requirements can live in the environments provided by R_{2e}. Assume first $e = 0$.

There are infinitely many versions of R_1 one for each outcome above.

1. R_1 living below (i, f):
   Here, R_1 is guessing that R_0 acts finitely. hence, this version can be initialized each time when we act for the sake of $x(0, j, s)$ for $j \leqslant i$. So this version simply implements the basic module.

2. R_1 living below (i, u):
   This version " knows" that i is the *least* number such that
   (a) $\lim_s x(0, i, s) = x(0, i)$ exists;
   (b) $x(0, j, s) \to \infty$ monotonically for $j > i$;
   (c) and the reason that this happens because B keeps changing $\gamma_0(x(0, i))$.

Hence, as with the thickness lemma, R_1 will only believe a $\Gamma_{e(1)}$-computation if $x(0, i + 1, s)$ is greater than the use of that computation.

Notice that this version of R_1 has numbers which are smaller than the apparent uses of e.g. $x(0, i + 1, s)$. But we can't let R_1 injure R_0. For instance, at some stage we might visit this version of R_1, give it a follower $x(1, j, s)$. This follower might well get realized. However, whilst it is realized we might have appointed many $x(0, n, s)$ for $n \geqslant i$. Since it *might* be that this version of R_1 is actually correct, we would not let the appointment of these $x(0, n, s)$-realizations cancel these $x(1, j, s)$.

It does not seem reasonable, at least from R_0's point of view, to allow $x(1, j, s)$ to injure R_0 *unless its guess looks correct.*

How might $x(1, j, s)$ injure R_1? *Answer: it could do so by being enumerated into A_1, killing some $x(0, n, s)$-set up.* Note that, but the way we appoint followers to be large, the only $x(0, n, s)$-set ups it might injure would necessarily have $n \geqslant i$.

Now when might we want to put $x(1, j, s)$ into A_1? Answer: when j enters C at some stage s. Now here is the difficulty: C has just permitted $x(1, j, s)$ to be enumerated into A_1. However, R_0 could have a $x(0, i, s)$-set up which is realized, which specifically forbids us to enumerate $x(1, j, s)$ into A_1. Now we come to the *main idea* of the density theorem:

The reason that $x(1, j, s)$ is forbidden is that there is some Γ_0-use which we want to preserve. Now since this version of R_1 is guessing (i, u), it is actually guessing that this use will be B-injured at some later stage. Thus at the stage that j enters C we promise *to put $x(1, j, s)$ into A_1 should the version of R_1 become accessible before the follower is initialized. That is, should B injure the $x(0, i, s)$ set up again. The key point is that B can sort this all out. B, being able to compute C will know if j ever enters C after $x(0, j, s)$ is appointed. If not then $x(0, j, s) \notin C$. If yes, then we will run the construction to see if $x(0, j, s)$ is still alive at the stage t where C permits j. We then see what the B-conditions are for (i, u) to be accessible again. If they don't ever occur, $x(1, j, s) \notin C$. If they do then we can run the construction till that stage $t' > t$ and see if $x(1, j, s)$ enters. Thus the A_i remain computable from B.*

The method above, invented by Sacks, is called *delayed permitting.* It is a subtle fact that while the true path of the construction remains only computable from $\mathbf{0}''$, the *fate of a particular follower* is always computable from B.

The verification that this version of R_1 meets the requirement should (i, u) be the true outcome of R_0 is essentially the same, since every C-permitted follower eventually has no R_0 restraint on it. We turn to the formal details.

Notation: For $\alpha \in \mathcal{T}, x(\alpha, i, s)$ will replace the notation $x(e, i, s)$ for the version of R_e at α.

Construction

The construction proceeds in substages $t \leqslant s$ at stage s. At substage t, we will either consider λ (substage 0) or we will have already generated $\alpha(s, t-1)$ and wish to generate $\alpha(s, t)$.

Let $\alpha = \alpha(s, t-1)$. With out loss of generality, suppose that $|\alpha| = 2e$, and so α is devoted to solving R_{2e}. If $x(\alpha, 0, s)$ is currently undefined, pick a big number y and let $\alpha(x, 0, s) = y$. Declare that $\alpha(s, t) = \alpha_s = \alpha(s, t-1)\widehat{\ }(0, f)$. Initialize all $\gamma \not\leqslant_{\mathrm{T}} \alpha\widehat{\ }(0, f)$.

Let $l(\alpha, s)$ be the α**-correct length** of the agreement. Here we say that $\Gamma_{e,s}(A_{1,s} \oplus B_s; q)$ is α-correct if for all $\eta\widehat{\ }(j, u) \subseteq \alpha$, if $x(\eta, j+1, s)\downarrow$, then $x(\eta, j+1, s) > \gamma_{e,s}(A_{1,s} \oplus B_s; q)$.

If $x(\alpha, 0, s)$ is currently defined, adopt the first of the following that pertains:

Case 1 For some (least) i with $x = x(\alpha, i, s)\downarrow$ and realized, i has entered C since the last α-stage, and the α-correct length of agreement has exceeded x since the last α-stage.

- *Action* Set $\alpha_s = \alpha(s, t) = \alpha\widehat{\ }t$, and initialize all $\tau \not\leqslant \alpha_s$. Put x into A_2.

Case 2 Case 1 fails and for some least i with $x = x(\alpha, i, s)$ realized, B has injured the $\Gamma_e(A_1 \oplus B_s; x)$ computation since the last α-stage.

- *Action* Declare x as unrealized. Cancel all $x(\alpha, j, s)$ for $j > i$. Set $\alpha(s, t) = \alpha\widehat{\ }(i, u)$. Initialize all τ with $\tau \not\leqslant_L \alpha\widehat{\ }(i, u)$ and $\alpha\widehat{\ }(i, u) \not\subseteq \tau$.

Case 3 Not case 1 nor 2, and for some unrealized follower $x = x(\alpha, i, s)$, we have $l(\alpha, s) > x$.

- *Action* Declare x as realized and pick a big number $x(\alpha, i+1, s)$. $\alpha(s, t) = \alpha\widehat{\ }(i+1, f) = \alpha_s$. Initialize all $\gamma \not\leqslant_L \alpha\widehat{\ }(i+1, f)$.

Case 4 Otherwise.

- *Action* Find the least (necessarily unrealized) $x(\alpha, i, s)\downarrow$. Set $\alpha(s, t) = \alpha\widehat{\ }(i, f)$. Initialize all $\gamma \not\leqslant_L \alpha\widehat{\ }(i, f)$ and all $\tau \supseteq \alpha\widehat{\ }(j, f)$ for all j.

End of construction

Verification

Let TP denote the true path.

Lemma 5.12.4. $A_i \leqslant_{\mathrm{T}} C$.

Proof. For x to enter, e.g., A_2, then x must be chosen as a follower by stage x. If $x \in A_{2,x}$, and x is a follower, it is of the form $x(\alpha, i, s)$ for some i. Find the stage where $C_s \upharpoonright i = C \upharpoonright i$. Then if $x \notin A_{2,s+1}$ and x is not yet canceled, x will only enter if there is another α-stage, as x is not canceled at stage $s+1$, yet $s+1$ is not an α-stage, it follows that $\alpha \not\leqslant_L \alpha_{s+1}$. By the cancellation that occurs in case 4, it follows further that there is a $i' < j$ and an η with $\eta\widehat{\ }(i', u) \subset \alpha$ and $\eta\widehat{\ }(j, -) \subset \alpha_{s+1}$. This can only be another α-stage if there is another $\eta\widehat{\ }(i', u)$-stage. Now C can use B to figure out if the use of $x(\eta, i', s)$ is B-correct. If it is, then $x \notin A_2$. If not, then go to a stage t where B changes the use. Then t is either a $\eta\widehat{\ }(i', u)$-stage, or x has been canceled by α at t. Continuing this, we see $A_2 \leqslant_{\mathrm{T}} C$. □

Lemma 5.12.5. *For all e,*

1. *R_e is met,* $\liminf_{\alpha_s \subset TP} |\alpha_s| = \infty$*, and*
2. *if $\alpha \leqslant_L TP$, α is initialized only finitely often).*

Proof. Let $\alpha \subseteq TP$ and suppose s_0 is such that $s > s_0, \alpha \leqslant_L \alpha_s$ and α is not initialized after s_0. Since α in never initialized, any $x(\alpha, i, s)$ is uncanceled except by $j < i$. Now argue as in the basic module, except using α-correct computations at α-stages, we will know that R_{2e} is met at α. If $\alpha\widehat{\ }(i, u) \subseteq TP$, then $\gamma \supseteq \alpha\widehat{\ }(i, u)$ is only initialized if $\alpha_s \leqslant_L \alpha\widehat{\ }(i, u)$, which only happens finitely often. If $\alpha\widehat{\ }(i, t) \subset TP$, then the α-module can act only finitely often. The lemma follows. □

□

5.13 Jump Theorems

We will see that the jump operator plays an important role in the study of algorithmic randomness. In this section, we look at several classic results about the range of the jump operator, whose proofs are combinations of techniques we have already seen. Of course, if $\mathbf{d} = \mathbf{a}'$ then $\mathbf{d} \geqslant \mathbf{0}'$. The following result is a converse to this fact.

Theorem 5.13.1 (Friedberg Completeness Criterion)**.** *If* $\mathbf{d} \geqslant \mathbf{0}'$ *then there is a degree* $\mathbf{a}$ *such that* $\mathbf{a}' = \mathbf{a} \vee \mathbf{0}' = \mathbf{d}$.

Proof. This is a finite extension argument. Let $D \in \mathbf{d}$. We construct a set A in stages, and let $\mathbf{a} = \deg(A)$. At stage 0, let $A_0 = \lambda$.

At odd stages $s = 2e+1$, we force the jump (i.e., decide whether $e \in A'$). We ask $\emptyset'$ whether there is a $\sigma \succ A_{s-1}$ such that $\Phi_e^\sigma(e)\downarrow$. If so, we search for such a σ and let $A_s = \sigma$. If not, we let $A_s = A_{s-1}$. Note that, in this case, we have ensured that $\Phi_e^A(e)\uparrow$.

At even stages $s = 2e+2$, we code $D(e)$ into A by letting $A_s = A_{s-1}D(e)$.

This concludes the construction. We have $A \oplus \emptyset' \leqslant_{\mathrm{T}} A'$ (which is always the case for any set A), so we can complete the proof by showing that $A' \leqslant_{\mathrm{T}} D$ and $D \leqslant_{\mathrm{T}} A \oplus \emptyset'$.

Since $\emptyset' \leqslant_{\mathrm{T}} D$, we can carry out the construction computably in D. To decide whether $e \in A'$, we simply run the construction until stage $2e+1$. At this stage, we decide whether $e \in A'$. Thus $A' \leqslant_{\mathrm{T}} D$.

An $\emptyset'$ oracle can tell us how to obtain A_{2e+1} given A_{2e}, while an A oracle can tell us how to obtain A_{2e+2} given A_{2e+1}, so using $A \oplus \emptyset'$, we can compute A_{2n+2} for any given n. Since $D(n)$ is the last element of A_{2n+2}, we can compute $D(n)$ using $A \oplus \emptyset'$. Thus $D \leqslant_{\mathrm{T}} A \oplus \emptyset'$. $\square$

The above proof should be viewed as a combination of a coding argument with the construction of a low set, done simultaneously. This kind of combination is a recurrent theme in the proofs of results such as the ones in this section.

It is not hard to show that if $A \leqslant_{\mathrm{T}} \emptyset'$ then A' is c.e. in $\emptyset'$. Since also $\emptyset' \leqslant_{\mathrm{T}} A'$, we say that A' is *computably enumerable in and above* $\emptyset'$, abbreviated by CEA($\emptyset'$). The following is an elaboration of the previous result.

Theorem 5.13.2 (Shoenfield Jump Inversion Theorem [270])**.** *If D is CEA($\emptyset'$) then there is an $A \leqslant_{\mathrm{T}} \emptyset'$ such that $A' \equiv_{\mathrm{T}} D$.*

The proof of this theorem is of some technical interest, since it involves more than just the finite extension method. Even more technical methods allow us to prove the following "jump and join" theorem.

Theorem 5.13.3 (Posner-Robinson Complementation Theorem, Posner and Robinson [240])**.** *If $\mathbf{a} \geqslant \mathbf{0}'$ and $\mathbf{0} < \mathbf{b} < \mathbf{a}$, then there is a $\mathbf{c}$ with $\mathbf{c} \cup \mathbf{b} = \mathbf{a}$ and $\mathbf{c} \cap \mathbf{b} = \mathbf{0}$.*

However, we will not prove either of the above results here. The proof of the second is quite involved[7], and we won't do the second but will rather sketch the proof of the following result, which generalizes Shoenfield's Jump Inversion Theorem.

[7]The original proof breaks into two cases according to the complexity of $\mathbf{b}$. We mention that Slaman and Steel [**?**] gave a uniform (but again difficult) proof of Theorem 5.13.3, and additionally showed that the complement $\mathbf{c}$ can be chosen as a 1-generic degree. Finally, recently, building on unpublished work of Seetapun and Slaman [**?**], Lewis shoed that for $\mathbf{a} = \mathbf{0}'$, the complement $\mathbf{c}$ can be shosen as a minimal degree.

Theorem 5.13.4 (Sacks [260]). *If D is CEA($\emptyset'$) then there is a c.e. set A such that $A' \equiv_{\mathrm{T}} D$.*

Proof sketch. We only sketch the proof of this beautiful result, since, from a modern viewpoint, it contains no new ideas beyond what we have seen so far. Let D be CEA($\emptyset'$). Then D is Σ_2^0, so there is an approximation $\{D_s\}_{s\in\mathbb{N}}$ such that $n \in D$ iff there is an s_n such that $n \in D_t$ for all $t > s_n$. Arguing as in the proof of Lemma 5.11.8, we have a set B such that $B^{[e]}$ is an initial segment of $\mathbb{N}^{[e]}$ and equals $\mathbb{N}^{[e]}$ iff $e \in D$.

We need to make the jump of A high enough to compute D, which we do by meeting for e the requirement

$$P_e : \ A^{[e]} =^* B^{[e]}.$$

As in Lemma 5.11.8, this action ensures that A' can compute D.

We also need to keep the jump of A computable in D, which we do by controlling the jump as in Theorem 5.9.3.

For P_e we will make sure that $A^{(e)} =^* B^{(e)}$, and hence, as with the high degree construction this makes sure that $S \leqslant_{\mathrm{T}} A'$. This is done as best we can: if we desire to enumerate an element from $B_s^{(e)}$ into A and we are not restrained from doing so, then enumerate it. This requirement will have nodes on the priority tree $\mathcal{T}$ which will have outcomes $\{\infty, f\}$. The first says that $B^{(e)} = \mathbb{N}^{(e)}$, and the second says that $B^{(e)} =^* \emptyset$. Naturally, in the end we would need to argue that if the outcome on the true path is $\nu\widehat{\ }\infty$ then almost all of $B^{(e)}$ will be emptied into $A^{(e)}$. *We refer to P_e nodes as* coding nodes.

For the N_e we try to make A low. *But now we make the set low on the tree.* The environment that such a lowness node lives in is that there will a finite number of coding nodes above it that it needs to cooperate with. For example, N_0 needs to cope with P_0, say. The version of N_0 guessing the finite outcome simply acts like a normal lowness requirement. However, the version of N_0 below the infinite outcome of P_0 "knows" that (in this case) *all* of $\mathbb{N}^{(0)}$ will be emptied into A, because that coding action has higher priority than N_0's restraint. (For later requirements this would be "almost all".) *Thus, as with the thickness lemma, this version of N_0 "doesn't believe" a $\Phi_0^A(0)[s]$ computation unless*

$$A_s^{(0)} \upharpoonright \varphi_0(0,s) = \mathbb{N}^{(0)} \upharpoonright \varphi_0(0,s).$$

The full construction works in the standard inductive way. As with the density theorem, whilst S cannot figure out the true path of the construction, it can sort out the fate of some coding marker, by deciding if a restraint is B-correct or not.

The reader who is not quite clear on how this construction works in detail is urged to write out the full construction and its verification. $\square$

The above result can be extended to higher jumps as follows.

Theorem 5.13.5 (Sacks [260]). *If D is CEA($\emptyset^{(n)}$) then there is a c.e. set A such that $A^{(n)} \equiv_{\mathrm{T}} D$.*

Proof. This result is proved by induction. Suppose that D is CEA($\emptyset^{(n+1)}$). Then we know that there is a set B that is CEA($\emptyset^{(n)}$) with $B' \equiv_{\mathrm{T}} D$, by the relativized form of Theorem 5.13.4. By induction, there is a c.e. set A with $A^{(n)} \equiv_{\mathrm{T}} B$. Then $A^{(n+1)} \equiv_{\mathrm{T}} D$. $\square$

There are even extensions of the above result to transfinite ordinals in place of n.

The Sacks Jump and Density Theorems have been extraordinarily influential. They have been extended and generalized in many ways. Roughly speaking, any "reasonable" set of requirements that do not explicitly contradict obvious partial ordering considerations (such as $\mathbf{a} \leqslant \mathbf{b} \Rightarrow \mathbf{a}' \leqslant \mathbf{b}'$) is realizable. Here is one example.

Theorem 5.13.6 (Jump Interpolation Theorem, Robinson [252]). *Let C and D be c.e. sets with $C <_{\mathrm{T}} D$, and let S be CEA(D) with $C' \leqslant_{\mathrm{T}} S$. Then there is a c.e. set A such that $C <_{\mathrm{T}} A <_{\mathrm{T}} D$ and $A' \equiv_{\mathrm{T}} S$.*

The methods for proving the Jump Interpolation Theorem are much the same as the ones we have seen. One can even prove vast generalizations for n-jumps with additional ordering requirements. For the last word on this subject, see Lempp and Lerman [173, 174].

5.14 Hyperimmune-free degrees

The notions of hyperimmune and hyperimmune-free degree have many applications in computability theory, and in the study of its interaction with algorithmic randomness. There are several ways to define these notions. We will adopt what is probably the simplest definition, using the concept of domination. A function g is *dominated* by a function f if $g(n) \leqslant f(n)$ for almost all n. It is sometimes technically useful to work with the following closely related concept: A function g is *majorized* by a function f if $g(n) \leqslant f(n)$ for *all* n.

Definition 5.14.1 (Miller and Martin [218]). A degree $\mathbf{a}$ is *hyperimmune* if there is a function $f \leqslant_{\mathrm{T}} \mathbf{a}$ that is not dominated by any computable function (or, equivalently, not majorized by any computable function). Otherwise, $\mathbf{a}$ is *hyperimmune-free*.

While $\mathbf{0}$ is clearly hyperimmune-free, all other Δ^0_2 degrees are hyperimmune, as shown by the following result.

Proposition 5.14.2 (Miller and Martin [218]). *If $\mathbf{a} < \mathbf{b} \leqslant \mathbf{a}'$, then $\mathbf{b}$ is hyperimmune. In particular, every nonzero degree below $\mathbf{0}'$, and hence every nonzero c.e. degree, is hyperimmune.*

Proof. We do the proof for the case $\mathbf{a} = \mathbf{0}$. The general result follows by a straightforward relativization.

Let B be a set such that $\emptyset <_{\mathrm{T}} B \leqslant_{\mathrm{T}} \emptyset'$. We need to find a function $g \leqslant_{\mathrm{T}} B$ that is not majorized by any computable function.

Since B is Δ^0_2, it has a computable approximation $\{B_s\}_{s \in \mathbb{N}}$. Define

$$g(n) = \mu s \geqslant x\,(B_s \upharpoonright n = B \upharpoonright n).$$

Notice that $g(n)$ is *not* the stage s by which the approximation to $B \upharpoonright n$ has stabilized (so that $B_t \upharpoonright n$ is correct for all $t \geqslant s$), but rather the first stage s at which $B_s \upharpoonright n$ is correct. Clearly, $g \leqslant_{\mathrm{T}} B$.

We claim that no computable function majorizes g. Suppose that h is computable and majorizes g. We claim that B is computable. To compute $B \upharpoonright m$, search for an $n > m$ such that $B_t \upharpoonright m = B_n \upharpoonright m$ for all $t \in [n, h(n)]$. Such an n must exist because there is a stage at which the approximation to $B \upharpoonright m$ stabilizes. By the definition of g and the choice of h, we have $g(n) \in [n, h(n)]$, so $B \upharpoonright m = B_{g(n)} \upharpoonright m = B_n \upharpoonright m$. Thus B is computable, which is a contradiction. □

On the other hand, there do exist nonzero hyperimmune-free degrees.

Theorem 5.14.3 (Miller and Martin [218]). *There is a nonzero hyperimmune-free degree.*

Proof. We define a noncomputable set A of hyperimmune-free degree, using a technique known as *forcing with computable perfect trees.* A *function tree* is a function $T : 2^{<\omega} \to 2^{<\omega}$ such that $T(\sigma 0)$ and $T(\sigma 1)$ are incompatible extensions of $T(\sigma)$. In this section and the next, we will refer to function trees simply as "trees". For a tree T, let $[T]$ be the collection of all X for which there is an infinite sequence β such that $T(\sigma) \prec X$ for all $\sigma \prec \beta$. We build a sequence $\{T_i\}_{i \in \mathbb{N}}$ of computable trees such that $[T_0] \supseteq [T_1] \supseteq [T_2] \supseteq \cdots$. Since each $[T_i]$ is closed, $\bigcap_n [T_n] \neq \emptyset$. We will take A to be any element of this intersection. At stage $2e+1$, we ensure that $A \neq W_e$, while at stage $2e+2$, we ensure that Φ^A_e is majorized by a computable function.

At stage 0, we let T_0 be the identity map.

At stage $2e+1$, we are given T_{2e}. Let $i \in \{0,1\}$ be such that $T_{2e}(i) \neq W_e \upharpoonright |T_{2e}(i)|$. Since $T_{2e}(0)$ and $T_{2e}(1)$ are incompatible, such an i must exist. Now define T_{2e+1} by letting $T_{2e+1}(\sigma) = T_{2e}(i\sigma)$. Notice that what this means is that $[T_{2e+1}]$ consists exactly of those elements of $[T_{2e}]$ that extend $T_{2e}(i)$, and hence, if $A \in [T_{2e+1}]$ then $A \neq W_e$.

At stage $2e+2$, we are given a computable tree T_{2e+1}. We wish to build T_{2e+2} to ensure that either

(i) if $A \in [T_{2e+2}]$ then Φ^A_e is not total, or

(ii) there is a computable function f such that if $A \in [T_{2e+2}]$ then $\Phi^A_e(n) \leqslant f(n)$ for all n.

First suppose there are n and σ such that $\Phi_e^{T_{2e+1}(\tau)}(n)\uparrow$ for all $\tau \succcurlyeq \sigma$. Then define T_{2e+2} by letting $T_{2e+2}(\nu) = T_{2e+1}(\sigma\nu)$. This definition clearly ensures that if $A \in [T_{2e+2}]$ then $\Phi_e^A(n)\uparrow$.

Now suppose there are no such n and σ. Then define T_{2e+2} as follows. For the empty string λ, search for a σ_λ such that $\Phi_e^{T_{2e+1}(\sigma_\lambda)}(0)\downarrow$ and define $T_{2e+2}(\lambda) = T_{2e+1}(\sigma_\lambda)$. Having defined σ_τ, search for incompatible extensions $\sigma_{\tau 0}$ and $\sigma_{\tau 1}$ of σ_τ such that $\Phi_e^{T_{2e+1}(\sigma_{\tau i})}(|\tau|)\downarrow$ for $i = 0, 1$, and define $T_{2e+2}(\tau i) = T_{2e+1}(\sigma_{\tau i})$. This process ensures that for all n and all τ of length n, we have $\Phi_e^{T_{2e+2}(\tau)}(n)\downarrow$. Furthermore, it ensures that T_{2e+2} is computable, so that we can define a computable function f by letting $f(n) = \max\{\Phi_e^{T_{2e+2}(\tau)}(n) : |\tau| = n\}$. Now, if $A \in [T_{2e+2}]$ then $\Phi_e^A(n) \leqslant f(n)$. □

Note that the above construction can be carried out effectively using $\emptyset''$ as an oracle, so there are nonzero Δ_3^0 hyperimmune-free degrees.

The name "hyperimmune degree" comes from the notion of a hyperimmune set, which we now define. A *strong array* is a computable collection of disjoint finite sets $\{F_i\}_{i\in\mathbb{N}}$ (which means not only that the F_i are uniformly computable, but that the function $i \mapsto \max F_i$ is computable). A set A is *hyperimmune* if for all strong arrays $\{F_i\}_{i\in\mathbb{N}}$, there is an i such that $F_i \subset \overline{A}$. A c.e. set is *hypersimple* if its complement is hyperimmune.

Given the terminology, one would expect that a hyperimmune degree is one that contains a hyperimmune set. We will show that this is indeed the case by first introducing another equivalent characterization of the hyperimmune degrees. The *principal function* of a set $A = \{a_0 < a_1 < \cdots\}$ is the function p_A defined by $p_A(n) = a_n$.

Lemma 5.14.4 (Miller and Martin [218])**.** *A degree* $\mathbf{a}$ *is hyperimmune iff* $\mathbf{a}$ *contains a set* A *such that* p_A *is not majorized by any computable function.*

Proof. Since $p_A \leqslant_{\mathrm{T}} A$, the "if" direction is obvious. For the other direction, suppose that there is a function $f \leqslant_{\mathrm{T}} \mathbf{a}$ that is not majorized by any computable function. We can assume that f is increasing. Let $B \in \mathbf{a}$ and let $b_0 < b_1 < \cdots$ be the elements of B. Let $A = \{f(b_n) : n \in \mathbb{N}\}$. Then $A \in \mathbf{a}$, and $p_A(n) \geqslant f(n)$ for all n, so p_A is not majorized by any computable function. □

Theorem 5.14.5 (Kuznecov, Medvedev [201], Uspenskii [307])**.** *A degree is hyperimmune iff it contains a hyperimmune set.*

Proof. Suppose that A is not hyperimmune. Then there is a strong array $\{F_i\}_{i\in\mathbb{N}}$ such that $A \cap F_i \neq \emptyset$ for all i. Let $f(n) = \max \bigcup_{i\leqslant n} F_i$. Then f is computable, and $p_A(n) \leqslant f(n)$ for all n, so $\deg(A)$ is not hyperimmune.

Now suppose that $\mathbf{a}$ is not hyperimmune. Then there is an $A \in \mathbf{a}$ and a computable function f such that $p_A(n) \leqslant f(n)$ for all n. Let $F_0 = [0, f(0)]$. Given F_i, let $k_i = \max F_i + 1$ and let $F_{i+1} = [k_i, f(k_i)]$. Then $p_A(k_i) \in A \cap F_i$ for all i, so A is not hyperimmune. □

5.15 Minimal degrees

The proof of Theorem 5.14.3 uses ideas that go back to a fundamental theorem of Spector.

Definition 5.15.1. A degree $\mathbf{a} > \mathbf{0}$ is *minimal* if there is no degree $\mathbf{b}$ with $\mathbf{0} < \mathbf{b} < \mathbf{a}$.

Theorem 5.15.2 (Spector [286])**.** *There is a minimal degree below* $\mathbf{0}''$.

Proof. As with Theorem 5.14.3, the proof uses forcing with computable perfect trees. As before, we build a sequence $\{T_i\}_{i\in\mathbb{N}}$ of computable trees such that $[T_0] \supseteq [T_1] \supseteq [T_2] \supseteq \cdots$, and take our set A of minimal degree to be any element of $\bigcap_n [T_n]$.

At stage 0, we let T_0 be the identity map. (Recall that our trees here are function trees, that is, functions $T : 2^{<\omega} \to 2^{<\omega}$ such that $T(\sigma 0)$ and $T(\sigma 1)$ are incompatible extensions of $T(\sigma)$.)

The action at stage $2e+1$ is exactly the same as in the proof of Theorem 5.14.3. That is, we let $i \in \{0,1\}$ be such that $T_{2e}(i) \neq W_e \upharpoonright |T_{2e}(i)|$ and define T_{2e+1} by letting $T_{2e+1}(\sigma) = T_{2e}(i\sigma)$. As before, this action ensures that $A \neq W_e$.

At stage $2e+2$, we want to meet the requirement

$$\mathcal{R}_e : \ \Phi_e^A \text{ total} \implies \Phi_e^A \equiv_{\mathrm{T}} \emptyset \vee A \leqslant_{\mathrm{T}} \Phi_e^A.$$

We are given a computable tree T_{2e+1} and build a computable subtree T_{2e+2}.

The first thing we do is follow the method used in the proof of Theorem 5.14.3 to try to force the nontotality of Φ_e^A. We ask $\emptyset''$ whether there are n and σ such that $\Phi_e^{T_{2e+1}(\tau)}(n)\uparrow$ for all $\tau \succcurlyeq \sigma$. If so, we define T_{2e+2} by letting $T_{2e+2}(\nu) = T_{2e+1}(\sigma\nu)$. This definition clearly ensures that $\Phi_e^A(n)\uparrow$, and hence that $\mathcal{R}_e$ is met.

If there there are no such n and σ, then we know that Φ_e is total on all elements of $[T_{2e+1}]$. So we now try to ensure that Φ_e^A is computable.

The critical question to ask $\emptyset''$ is whether there is a σ such that for all n and all $\tau_0, \tau_1 \succcurlyeq \sigma$, if $\Phi_e^{T_{2e+1}(\tau_i)}(n)\downarrow$ for $i = 0,1$, then $\Phi_e^{T_{2e+1}(\tau_0)}(n) = \Phi_e^{T_{2e+1}(\tau_1)}(n)$. If this question has a positive answer, then we define T_{2e+2} by letting $T_{2e+2}(\nu) = T_{2e+1}(\sigma\nu)$. Then, to compute $\Phi_e^A(n)$, we simply need to find a μ such that $\Phi_e^{T_{2e+2}(\mu)}(n)\downarrow$, and we are guaranteed that $\Phi_e^A(n) = \Phi_e^{T_{2e+2}(\mu)}(n)$. Thus, in this case, Φ_e^A is computable, and hence $\mathcal{R}_e$ is met.

If the answer to the above question is negative, then we need to ensure that $\Phi_e^A \leqslant_{\mathrm{T}} A$. The crucial new idea is Spector's notion of an *e-splitting tree*.[8]

[8] We are being a bit free and easy with history. The notion of being e-splitting was first used by Spector, but the influential use of *trees* in the construction of a minimal degree actually first occurred in Shoenfield's book [273].

Definition 5.15.3. We say that a tree T is *e-splitting* if for each σ, there is an n such that $\Phi_e^{T(\sigma 0)}(n)\downarrow \neq \Phi_e^{T(\sigma 1)}(n)\downarrow$.

The crucial fact about e-splitting trees is the following.

Lemma 5.15.4. *Let T be e-splitting and let $B \in [T]$. Then $\Phi_e^B \geqslant_{\mathrm{T}} B$.*

Proof. Suppose we are given Φ_e^B. Search for an n_0 such that $\Phi_e^{T(0)}(n_0)\downarrow \neq \Phi_e^{T(1)}(n_0)\downarrow$. Let i be such that $\Phi_e^B(n_0) = \Phi_e^{T(i)}(n_0)$. Then we know that $T(i) \prec B$. Now find an n_1 such that $\Phi_e^{T(i0)}(n_1)\downarrow \neq \Phi_e^{T(i1)}(n_1)\downarrow$. Let j be such that $\Phi_e^B(n_1) = \Phi_e^{T(ij)}(n_1)$. Then $T(ij) \prec B$. Continuing in this way, we can compute more and more of B. Thus $B \leqslant_{\mathrm{T}} \Phi_e^B$. ☐

So to meet $\mathcal{R}_e$, it is enough to ensure that T_{2e+2} is e-splitting. But recall that we are now in the case in which for each σ there are n and $\tau_0, \tau_1 \succ \sigma$ such that $\Phi_e^{T_{2e+1}(\tau_0)}(n)\downarrow \neq \Phi_e^{T_{2e+1}(\tau_1)}(n)\downarrow$. So we can define T_{2e+2} as follows. Let $T_{2e+2}(\lambda) = T_{2e+1}(\lambda)$. Suppose we have defined $T_{2e+2}(\mu)$ to equal $T_{2e+1}(\sigma)$ for some σ. Search for τ_0 and τ_1 as above and define $T_{2e+2}(\mu i) = T_{2e+1}(\tau_i)$ for $i = 0, 1$. Then T_{2e+2} is computable and e-splitting. ☐

A crucial difference between hyperimmune-free degrees and minimal degrees is given by the following result.

Theorem 5.15.5 (Yates [325], Cooper [54])**.** *Every nonzero c.e. degree bounds a minimal degree.*

In particular, there are minimal degrees below $\mathbf{0}'$ (which was first shown by Sacks [259]), and hence hyperimmune minimal degrees. The method used to prove Theorem 5.15.5, known as the *full approximation method*, is quite involved and would take us a little too far afield. We refer the reader to Lerman [176] or Odifreddi [234] for more details.

Minimal degrees form a very interesting class. We finish by quoting two important results about their possible Turing degrees.

Theorem 5.15.6 (Jockusch and Posner [135])**.** *All minimal degrees are GL_2. That is, if $\mathbf{a}$ is a minimal degree then $\mathbf{a}'' \leqslant (\mathbf{a} \vee \mathbf{0}')'$.*

This result improved an earlier one by Cooper, who showed that no minimal degree can be high and below $\mathbf{0}'$. Cooper [55] also proved the following definitive result.

Theorem 5.15.7 (Cooper [55])**.** *If $\mathbf{b} > \mathbf{0}'$ then there is a minimal degree $\mathbf{a}$ with $\mathbf{a}' = \mathbf{b}$.*

Thus there is an analog of the Friedberg Jump Theorem for minimal degrees. It is not possible to prove an analog of the Shoenfield Jump Theorem since Downey, Lempp, and Shore [88] showed that there are degrees CEA($\emptyset'$) and low over $\mathbf{0}'$ that are not jumps of minimal degrees. Cooper

[56] claimed a characterization of the degrees CEA($\emptyset'$) that are jumps of minimal degrees below $\mathbf{0}'$, but no details were given.

5.16 Π^0_1 and Σ^0_1 classes

5.16.1 Basics

A *tree* is a subset of $2^{<\omega}$ closed under initial segments. A *path* through a tree T is an infinite sequence $P \in 2^\omega$ such that if $\sigma \prec P$ then $\sigma \in T$. The collection of paths through T is denoted by $[T]$. A subset of 2^ω is a Π^0_1 *class* if it is equal to $[T]$ for some computable tree T.

Remark 5.16.1. Sometimes a Π^0_1 class is defined to be the set of paths through a computable subtree T of $\omega^{<\omega}$. Such a class is *computably bounded* if there is a computable function f such that for each $\sigma \in T$, if $\sigma n \in T$ then $n \leqslant f(\sigma)$. The study of computably bounded Π^0_1 classes reduces to the study of the special case of Π^0_1 subclasses of 2^ω, since every computably bounded Π^0_1 class is computably equivalent to a Π^0_1 subclass of 2^ω. We will restrict our attention to such classes, so for us a Π^0_1 class will mean a Π^0_1 subclass of 2^ω.

An equivalent formulation is that C is a Π^0_1 class if there is a computable relation R such that

$$C = \{\alpha \in 2^\omega : \forall n\, R(\alpha \upharpoonright n)\}.$$

It is not hard to show that this definition is equivalent to the one via computable trees.

There is a host of natural examples of Π^0_1 classes important to several branches of logic.

Example 5.16.2.

- Let A and B be disjoint c.e. sets. The collection of *separating sets* $\{X : X \supseteq A \wedge X \cap B = \emptyset\}$ is a Π^0_1 class.

- A special case of the above is an *effectively inseparable* pair, that is, a pair of disjoint c.e. sets A and B for which there is a computable function f such that for all disjoint c.e. $W_e \supseteq A$ and $W_j \supseteq B$, we have $f(e,j) \notin W_e \cup W_j$. This is the two variable version of a creative set.

- Another example of a Π^0_1 class is the class of (codes of) extensions of a complete consistent theory. By the proof of Gödel's Incompleteness Theorem, for a theory such as Peano Arithmetic (PA), the set A of sentences provable from PA and the set B of sentences refutable from PA form an effectively inseparable pair.

The complement of a Π^0_1 class is a Σ^0_1 *class*. Thus C is a Σ^0_1 class if there is a computable relation R such that

$$C = \{\alpha : \exists n \, R(\alpha \upharpoonright n)\}.$$

We can think of Σ^0_1 classes as the analogs for infinite sequences of c.e. sets for finite strings.

An important fact about Σ^0_1 classes is that they are determined by clopen sets.

Observation 5.16.3. *Let C be a Σ^0_1 class. Then there is a c.e. set $W \subseteq 2^{<\omega}$ such that $C = \bigcup\{[\sigma] : \sigma \in W\}$.*

We will not need much of the theory of Π^0_1 and Σ^0_1 classes, but a few facts will be important, such as the following bits of folklore.

Proposition 5.16.4. *Let C be a Π^0_1 class with only finitely many members. Then every member of C is computable.*

Proof. Let T be a computable tree such that $C = [T]$. Then T has only finitely many paths. Let P be a path through T. There must be a $\sigma \prec P$ such that no other path through T extends σ. Thus, for any $n > |\sigma|$, there is exactly one $\tau \succ \sigma$ such that the portion of T above τ is infinite. (Here we appeal to König's Lemma, which says that if a finitely branching tree is infinite, then it has a path.) So to compute $P \upharpoonright n$ for $n > |\sigma|$, we look for an $m \geqslant n$ such that exactly one $\tau \succ \sigma$ of length n has an extension of length m in T. Then $P \upharpoonright n = \tau$. $\square$

Corollary 5.16.5. *Let C be a nonempty Π^0_1 class with no computable members. Then $|C| = 2^{\aleph_0}$.*

Proof. Let T be a computable tree such that $C = [T]$. Let S be the set of *extendible* elements of T, that is, those σ such that there is an extension of σ in $[T]$. If every $\sigma \in S$ has two incompatible extensions in S, then it is easy to show that there are continuum many paths through T. Otherwise, there is a $\sigma \in S$ with a unique extension $P \in [T]$. As in the previous proof, P is computable. $\square$

Note also that the question of whether a given computable tree T is finite is an existential one, since it amounts to asking whether there is an n such that no string of length n is in T, and hence it can be decided by $\emptyset'$.

5.16.2 Π^0_n and Σ^0_n classes

There is a hierarchy of classes, akin to the arithmetical hierarchy of sets.

Definition 5.16.6. A collection C of reals is a Π^0_n *class* if there is a computable relation R such that

$$C = \{\alpha : \forall k_1 \, \exists k_2 \cdots Q k_n \, R(\alpha \upharpoonright k_1, \alpha \upharpoonright k_2, \ldots, \alpha \upharpoonright k_n)\},$$

where the quantifiers alternate and hence $Q = \forall$ if n is odd and $Q = \exists$ is n is even. The complement of a Π^0_n class is a Σ^0_n *class*.

These definitions can also be relativized to a given set X (by letting R be X-computable) to obtain the notions of Π^X_n class and Σ^X_n class. Note that a Π^0_{n+1} class is also a $\Pi^{\emptyset^{(n)}}_1$ class. However, the converse is not always true. For instance, every $\Pi^{\emptyset'}_1$ class is closed, but there are Π^0_2 classes that are not closed, such as the class of all α such that $\alpha(m) = 0$ for infinitely many m, which can be written as $\{\alpha : \forall k\, \exists m\, (m > k \wedge \alpha(m) = 0)\}$.

5.16.3 Basis Theorems

One of the most important classes of results on Π^0_1 classes is that of *basis theorems*. A basis theorem for Π^0_1 classes states that every nonempty Π^0_1 class has a member of a certain type. (Henceforth, all Π^0_1 classes will be taken to be nonempty without further comment.) For instance, it is not hard to check that the lexicographically least element of a Π^0_1 class (i.e., the leftmost path of a computable tree T such that $[T] = C$) has c.e. degree.[9] This fact establishes the C.E. Basis Theorem of Jockusch and Soare, which states that every Π^0_1 class has a member of c.e. degree. This results is a refinement of the earlier Kreisel Basis Theorem, which states that every Π^0_1 class has a Δ^0_2 member.

The following is the most famous and widely applicable basis theorem.

Theorem 5.16.7 (Low Basis Theorem, Jockusch and Soare [137])**.** *Every Π^0_1 class has a low member.*

Proof. Let $C = [T]$ with T a computable tree. We define a sequence $T = T_0 \subseteq T_1 \subseteq \cdots$ of infinite computable trees such that if $P \in \bigcap_e [T_e]$ then P is low. (Note that there must be such a P, since each $[T_e]$ is closed.)

Suppose that we have defined T_e, and let

$$U_e = \{\sigma : \Phi^\sigma_e(e)[|\sigma|]\uparrow\}.$$

Then U_e is a computable tree. If $U_e \cap T_e$ is infinite, then let $T_{e+1} = T_e \cap U_e$. Otherwise, let $T_{e+1} = T_e$. Note that either $\Phi^P(e)\uparrow$ for all $P \in [T_{e+1}]$ or $\Phi^P(e)\downarrow$ for all $P \in [T_{e+1}]$.

Now suppose that $P \in \bigcap_e [T_e]$. We can perform the above construction computably in $\emptyset'$, since $\emptyset'$ can decide the question of whether $U_e \cap T_e$ is finite. Thus $\emptyset'$ can decide whether $\Phi^P(e)\downarrow$ for a given e, and hence P is low. $\square$

[9] Such an element is in fact a *left-c.e. real*, a concept that will be defined in Chapter 8.

Actually, the above proof above has a consequence that will be useful below. We call a set A *superlow* if $A' \equiv_{\mathrm{tt}} \emptyset'$. Marcus Schaeffer observed that the proof of the Low Basis Theorem actually produces a superlow set.

Corollary 5.16.8 (Schaeffer)**.** *Every Π^0_1 class has a superlow member.*

Another important basis theorem is provided by the hyperimmune-free degrees.

Theorem 5.16.9 (Hyperimmune-Free Basis Theorem, Jockusch and Soare [137])**.** *Every Π^0_1 class has a member of hyperimmune-free degree.*

Proof. Let C be a Π^0_1 class, and let T be a computable tree such that $C = [T]$. We can carry out a construction like that in the proof of Miller and Martin's Theorem 5.14.3 within T. We begin with $T_0 = T$. We do not need the odd stages of that construction since we do not need to force noncomputability (as $\mathbf{0}$ is hyperimmune-free). Thus we deal with Φ_e at stage $e+1$.

We are given T_e and we want to build T_{e+1} to ensure that either

(i) if $A \in [T_{e+1}]$ then Φ^A_e is not total, or

(ii) there is a computable function f such that if $A \in [T_{e+1}]$ then $\Phi^A_e(n) \leqslant f(n)$ for all n.

Let

$$U^n_e = \{\sigma \in T_e : \Phi^\sigma_e(n)\uparrow\}.$$

There are two cases.

If there is an n such that U^n_e is infinite, then let $T_{e+1} = U^n_e$. In this case, if $A \in [T_{e+1}]$ then $\Phi^A_e(n)\uparrow$.

Otherwise, let $T_{e+1} = T_e$. In this case, for each n we can compute a number $l(n)$ such that no string $\sigma \in T_e$ of length $l(n)$ is in U^n_e. For any such σ, we have $\Phi^\sigma_e(n)\downarrow$. Define the computable function f by

$$f(n) = \max\{\Phi^\sigma_e(n) : \sigma \in T_e \wedge |\sigma| = l(n)\}.$$

Then $A \in [T_{e+1}]$ implies that $\Phi^A_e(n) \leqslant f(n)$ for all n.

So if we let $A \in \bigcap_e [T_e]$, then A is a member of C and has hyperimmune-free degree. $\square$

In the above construction, if T has no computable members, then Corollary 5.16.5 implies that each T_e must have size $2^{\aleph_0}$. It is not hard to adjust the construction in this case to obtain continuum many members of C of hyperimmune-free degree. Thus, a Π^0_1 class with no computable members has continuum many members of hyperimmune-free degree.

The following result is an immediate consequence of the Hyperimmune-Free Basis Theorem and Theorem 5.14.2, but it was first explicitly articulated by Kautz [140], who gave an interesting direct proof.

Corollary 5.16.10 (Jockusch and Soare [137], Kautz [140]). *Every Π^0_1 class has a member that is not computably enumerable in any set of lower Turing degree.*

Although the existence of minimal degrees (Theorem 5.15.2) was proved by techniques similar to those used to prove the Hyperimmune-Free Basis Theorem, there is no "minimal basis theorem", since there are Π^0_1 classes with no members of minimal degree. Using techniques beyond the scope of this book, Groszek and Slaman [121] proved the following elegant result.

Theorem 5.16.11 (Groszek and Slaman [121]). *There is a Π^0_1 class such that every member either is c.e. and noncomputable, or has minimal degree.*

We end this section by looking at examples of *nonbasis* theorems.

Proposition 5.16.12. *The intersection of all bases for the Π^0_1 classes is the collection of computable sets, and hence there is no minimal basis.*

Proof. If A is a computable set then $\{A\}$ is a Π^0_1 class, so every basis for the Π^0_1 classes must include A. On the other hand, if B is noncomputable, then every Π^0_1 class must contain a member other than B, so the collection of all sets other than B is a basis for the Π^0_1 classes. □

For our next result, we will need the following definition and lemma.

Definition 5.16.13. An infinite set A is *effectively immune* if there is a computable function f such that for all e, if $W_e \subseteq A$ then $|W_e| \leqslant f(e)$.

Post gave the following construction of an effectively immune co-c.e. set A. At stage s, for each $e < s$, if $W_e[s] \subseteq A_s$ and there is an $x \in W_e[s]$ such that $x > 2e$, then put the least such x into $\overline{A}$. Then A is infinite, and is effectively immune via the function $e \mapsto 2e$.

Lemma 5.16.14 (Martin [197]). *If a c.e. set B computes an effectively immune set, then B is Turing complete.*

Proof. Let B be c.e., and let $A \leqslant_{\mathrm{T}} B$ be effectively immune, as witnessed by the computable function f. Let Γ be a reduction such that $\Gamma^B = A$.

For each k we build a c.e. set $W_{h(k)}$, where the index $h(k)$ is given by the Recursion Theorem. Initially, $W_{h(k)}$ is empty. If k enters $\emptyset'$ at stage t then we wait until a stage $s \geqslant t$ at which there is a q such that $\Gamma^B(n)[s]\downarrow$ for all $n < q$ and $|\Gamma^B[s] \upharpoonright q| > f(h(k))$. We then let $W_{h(k)} = \Gamma^B[s] \upharpoonright q$. This action ensures that $|W_{h(k)}| > f(h(k))$, so we must have $W_{h(k)} \nsubseteq A$. Thus,

$$\Gamma^B[s] \upharpoonright q = W_{h(k)} \neq A \upharpoonright q = \Gamma^B \upharpoonright q.$$

So to compute $\emptyset'(k)$ from B, we simply look for a stage s at which there is a q such that $|\Gamma^B[s] \upharpoonright q| > f(h(k))$ and the computation $\Gamma^B[s] \upharpoonright q$ is B-correct. If k is not in $\emptyset'$ by stage s, it cannot later enter $\emptyset'$. □

Theorem 5.16.15 (Jockusch and Soare [136]). (i) *The incomplete c.e. degrees do not form a basis for the Π^0_1 classes.*

(ii) *Let* $\mathbf{a}$ *be an incomplete c.e. degree. Then the degrees less than or equal to* $\mathbf{a}$ *do not form a basis for the* Π^0_1 *classes.*

Proof. By Lemma 5.16.14, to prove both parts of the theorem, it is enough to find a Π^0_1 class whose members are all effectively immune. Let A be Post's effectively immune set described after Definition 5.16.13. An infinite subset of an effectively immune set is clearly also effectively immune, so it is enough to find a Π^0_1 class all of whose members are infinite subsets of A.

It is clear from the construction of A that $|\overline{A} \upharpoonright 2e| \leqslant e$ for all e. Thus $A \cap [2^k - 1, 2^{k+1} - 2] \neq \emptyset$ for all k.[10] Let

$$\mathcal{P} = \{B : B \subseteq A \wedge \forall k\, (B \cap [2^k - 1, 2^{k+1} - 2] \neq \emptyset)\}.$$

Since A is co-c.e., $\mathcal{P}$ is a Π^0_1 class, and it is nonempty because $A \in \mathcal{P}$. Furthermore, every element of $\mathcal{P}$ is an infinite subset of A. As explained in the previous paragraph, these facts suffice to establish the theorem. $\square$

5.16.4 Generalizing the Low Basis Theorem

We would like to extend the Low Basis Theorem to other jumps. Of course, we cannot hope to prove that for every degree $\mathbf{a} \geqslant \mathbf{0}'$, every Π^0_1 class has a member whose jump has degree $\mathbf{a}$, since there are Π^0_1 classes whose members are all computable. We can prove this result, however, if we restrict our attention to *special* Π^0_1 classes, which are those with no computable members.

Theorem 5.16.16 (Folklore, see Cenzer [41], Odifreddi [233, Exercise V.5.33])**.** *If* C *is a* Π^0_1 *class with no computable members and* $\mathbf{a} \geqslant \mathbf{0}'$, *then there is a* $P \in C$ *such that* $\deg(P') = \mathbf{a}$.

Proof. This is an extension of the Friedberg completeness criterion [111] to the me members of Π^0_1 classes and its proof is a straightforward generalization of the Low Basis Theorem of Jockusch and Soare [136]. Thus let $S \geqslant_{\mathrm{T}} \emptyset'$

This time let $C = [T]$ with T computable. Remember that T has $2^{\aleph_0}$ many paths as it has no computable members. Again We define an infinite sequence of computable trees, $\{T_e : e \in \omega\}$ so that $P \in \cap_e [T_e]$ and $P \in C$. Each of the tree will have $2^{\aleph_0}$ many paths.

We build a mapping $f_e : 2^{<\omega} \mapsto T_e$. We will write $T(\nu)$ for the image of ν under this mapping. For T_0 the mapping is the identity.

Let $T_0 = T$.

Stage $2e + 1$ By induction, assume that T_{2e} is defined.Let

$$U_{2e} = T_{2e} \cap \{\sigma : \Phi^{\sigma}_{e,|\sigma|}(e) \uparrow\}.$$

[10]Notice that this fact shows that A is not hyperimmune. In fact, it is not hard to show that a co-c.e. set X is not hyperimmune iff there is a Π^0_1 class all of whose members are infinite subsets of X.

Then U_{2e} forms a computable subtree of T_{2e} which has *a fortiori* no computable paths. Then again let $T_{2e+1} = T_{2e}$ if U_{2e} is finite, and $T_{2e+1} = U_{2e}$ if infinite. This keeps the jump down as with the Low Basis Theorem.
Stage $2e+2$. After stage $2e+1$ we will have a tree T_{2e+1} with no computable paths, and which is computable.

Notice that, if $\tilde{T}$ is any computable tree with no computable paths, then for σ on $\tilde{T}$ it is only a $\emptyset'$-computable question whether $[\sigma] \cap \tilde{T}$ is infinite (and hence uncountable with no computable paths.) We merely have to ask if σ has an extension in $\tilde{T}_s$ for all $s > |\sigma|$.

Thus, $\emptyset'$ can construct a mapping $f : 2^{<\omega} \mapsto T_{2e+1}$ which picks out a perfect subtree of T_{2e+1}. We ask that T_{2e+2} consist of the subtree of T_{2e} extending $f(0)$ if $e \notin S$, and extending $f(1)$ if $e \in S$. (This is the analog of the Friedberg coding step.)

Then we can choose $P \in \cap_e[T_e]$, and we have forced $P' \leqslant_{\mathrm{T}} \emptyset' \oplus S \equiv_{\mathrm{T}} S$ as the whole construction is computable in $\emptyset' \oplus S$. Additionally, $P \oplus \emptyset'$ can simulate the construction to decide if $e \in S$ and hence the result follows. $\square$

Corollary 5.16.17. *The set $\{\mathbf{b} \geqslant \mathbf{0}'\} \cup \{\mathbf{0}\}$ forms a basis for the Π^0_1 classes.*

5.17 Strong reducibilities and Post's Program

The last theorem from the previous section used hypersimple sets. We will see them again later, since they turn out to be related to the study of algorithmic randomness. Sets of this form were originally introduced by Post [238] in an attempt to solve Post's Problem. Although hypersimple sets can be Turing complete Post [238] proved that no hypersimple set can be tt-complete. Later, Friedberg and Rogers [113] proved that no hypersimple set can be wtt-complete. This result has been extended as follows.

Theorem 5.17.1 (Downey and Jockusch [83])**.** *No hypersimple set is wtt-cuppable. That is, if A is hypersimple and $W <_{\mathrm{wtt}} \emptyset'$, then $A \oplus W \ngeqslant_{\mathrm{wtt}} \emptyset'$.*

Proof. Suppose that A is hypersimple, $W \leqslant_{\mathrm{wtt}} \emptyset'$, and $A \oplus W \geqslant_{\mathrm{wtt}} \emptyset'$. We show that $W \geqslant_{\mathrm{wtt}} \emptyset'$. We will build a c.e. set B. By the Recursion Theorem, we can assume that we have a wtt-reduction $\Gamma^{A \oplus W} = B$ with computable use γ. Without loss of generality, we can assume that $\gamma(x)$ is even for all x.

We define a strong array $\{F_n\}_{n \in \omega}$ and an auxiliary function f as follows. Let $F_0 = \{x : 0 \leqslant x \leqslant \gamma(0)/2\}$ and $f(0) = \gamma(0)/2$. Given F_n and $f(n)$, pick $f(n)+1$ many new fresh numbers $b_0^{n+1} < \cdots < b_{f(n)}^{n+1}$, let $F_{n+1} = \{x : f(n) < x \leqslant \gamma(b_{f(n)}^{n+1})/2\}$ and let $f(n+1) = \gamma(b_{f(n)}^{n+1})/2$. Since the F_n form a strong array, there are infinitely many n with $F_n \subseteq A$. We can therefore enumerate an increasing computable sequence $n_0 < n_1 < \cdots$ such that $F_{n_i} \subseteq A$ for all i.

Now for each i, we wait until a stage s_i such that $F_{n_i} \subseteq A_{s_i}$. If i enters $\emptyset'$ after stage s_i, we put b_0^{i+1} into B. If later A changes below $\gamma(i)/2$, we put b_1^{i+1} into B. We continue in this manner, enumerating the elements of

CHANGE HERE

Once we figure out such a subsequence, we *then* define Δ. Having defined $\Delta^{A_s} \upharpoonright i-1$, we use the next $F_{x_i} \subseteq A$ to define $\Delta^{A_s}(i)$. Specifically we will define $\Delta^{A_s}(i)$ will have use $f(x_i)+1$, and it is first defined at $s = s_i$ after the members of F_{x_i} enter A. Naturally we set this to be 0 unless we are lucky in that i is already in $\emptyset'$, in which case we would set it to be 1, and maintain that henceforth.

The key point is that if $\Gamma^{A_s \oplus W_s} \upharpoonright b^{x_i}_{f(x_i)} = B_s \upharpoonright b^{x_i}_{f(x_i)}$, but that $A - A_s$ can change at most $f(x_i) - 1$ times after $\Delta^{A_s}(i)$ is defined (since the rest have *already entered* A. But we have the power to change A_t via B_t at least $f(x_i)+1$ many times, and hence we can force a W change, giving the reduction.

That is, if i enters $\emptyset' - \emptyset'_s$ then use the interval $[f(x_i), \ldots, b^{x_i}_{f(x_i)}]$ enumerated one at a time into B, waiting for either A or W to change till eventually we cause a $A - A_s \upharpoonright \delta(i)$ change. $\square$

Post's original program to find a "thinness" property of the lattice of supersets of a c.e. set guaranteeing Turing incompleteness of the given set was eventually proven to fail. Although Sacks [262] constructed a maximal set (i.e., a coinfinite c.e. set M such that if $W \supseteq M$ is c.e. then either W is cofinite or $W - M$ is finite) that is Turing incomplete, it is also possible to construct a maximal set that is Turing complete. Indeed, Soare [278] showed that the maximal c.e. sets form an orbit in the lattice of c.e. sets under inclusion, and hence there is no definable property that can be added to maximality to guarantee incompleteness. Eventually, Cholak, Downey, and Stob [49] showed that no property of the lattice of c.e. supersets of a c.e. set *alone* can guarantee incompleteness. Finally, Harrington and Soare [123] did find an elementary property of c.e. sets (that is, one that is first order definable in the language of the lattice of c.e. sets) that does guarantee incompleteness. There is a large amount of fascinating material here. For instance, Cholak and Harrington [48] have shown that one can define all "double jump" classes in the lattice of computably enumerable sets using infinitary formulas. The methods are intricate and rely on analyzing the failure of the "automorphism machinery" first developed by Soare and later refined by himself and others, particularly Cholak and Harrington.

Properties like simplicity and hypersimplicity do have implications for the degrees of sets related to a given c.e. set. For instance, Stob [297] showed that a c.e. set is simple iff it does not have c.e. supersets of all c.e. degrees. Downey [67] proved that if A is hypersimple then there is a c.e.

degree $\mathbf{b} \leqslant \deg(A)$ such that if $A_1 \sqcup A_2$ is a c.e. splitting of A, then neither of the A_i has degree $\mathbf{b}$.

5.18 PA degrees

The collection of sets separating two disjoint c.e. sets is a natural Π^0_1 class. As a consequence, the set of consistent extensions of a consistent theory, and the set of complete extensions of a consistent theory both form Π^0_1 classes. Jockusch and Soare [136, 137] prove that the class of degrees of members of a Π^0_1 class coincide with the class of degrees of complete extensions of a computably axiomatizable first order theory. This was later extended by Hanf [122] for finitely axiomatizable theories.

In this section we will look at complete extensions of Peano Arithmetic.

Definition 5.18.1 (PA degree)**.** We say that a degree $\mathbf{a}$ is PA iff it is the degree of a complete extension of Peano Arithmetic.

Theorem 5.18.2 (Scott Basis Theorem, Scott [267])**.** *If S is a consistent theory extending PA, then the sets computable from S form a basis for the Π^0_1 classes.*

Proof. Let T be a computable tree. We wish to compute a path through T using S. To do so, we define by induction a sequence $\sigma_0 \prec \sigma_1 \prec \cdots$ of strings in T, with $|\sigma_i| = i$.

Let σ_0 be the empty string. Now suppose that σ_n has been defined in such a way that there is a path through T extending σ_n. If σ_n has only one extension of length $n+1$ in T, then let σ_{n+1} be that extension. Otherwise, both $\sigma_n 0$ and $\sigma_n 1$ are in T. For $i \in \{0, 1\}$, let θ_i denote the sentence

$$\exists m\, (\sigma_n i \text{ has an extension of length } m \text{ in } T \text{ but } \sigma_n(1-i) \text{ does not}).$$

Note that these sentences can be expressed in the language of first-order arithmetic, as shown in the proof of Gödel's Incompleteness Theorem.

CHANGE THIS

We have $PA \vDash \neg(\theta_0 \wedge \theta_1)$, so

If $\sigma_n i$ can be extended to a path through T but $\sigma_n(1-i)$ cannot, then θ_i is true. In this case, $PA \vDash \theta_i$, so $S \vDash \theta_i$.

We have $PA \vDash \neg(\theta_0 \wedge \theta_1)$. Thus, since S is complete,

It cannot be the case that both θ_0 and θ_1 are true.

If θ_i is true then $PA \vDash \theta_i$, and hence $S \vDash \theta_i$.

Also $PA \vDash \neg(\varphi_0 \wedge \varphi_1)$, so $S \nvDash \varphi_0$ or $S \vDash \varphi_1$. Then let σ_{n+1} be a φ_i so that $S \nvDash \varphi_{1-i}$. It is easy to show by induction on n that σ_n has infinitely many extensions in T. ☐

One way to construct PA degree is to use effectively inseparable pairs. Let C be a separating set for an effectively inseparable pair. The by the well-known Lindenbaum construction, we can construct a complete extension T

of Peano arithmetic computable from C. We may construct $T \leqslant_{\mathrm{T}} C$ by the universality property of effectively inseparable pairs, there is an effective procedure, which, when given the indices of any two disjoint c.e. set A and B yields an index of a computable function f such that $f^{-1}(C)$ separates A and B. The summary of this is the following.

Theorem 5.18.3 (Jockusch and Soare [136]). *Suppose that C is the degree of a separating set of an effectively inseparable pair. Then C has PA degree.*

In unpublished work, Solovay characterized the PA degrees as follows.

Theorem 5.18.4 (Solovay). *The following are equivalent:*

(i) $\mathbf{a}$ *is PA.*

(ii) $\mathbf{a}$ *is the degree of a consistent extension of PA.*

(iii) The degrees below $\mathbf{a}$ *form a basis for* Π^0_1 *classes.*

Proof. This result is pretty well known. We will follow the proof of Odifreddi [233]. It is enough to prove that the PA degrees are closed upwards. Let F be a PA degrees computable in a set C. We build a tree T of complete extensions of PA computable in F. This uses a Henkin-type construction.

Let $\{\varphi_n : n \in \mathbb{N}\}$ be an enumeration of the sentences of arithmetic. Let $F_\lambda = PA$. Suppose that we have F_σ. Now given φ_n we need to add one of φ_n or $\overline{\varphi_n}$ to F_σ. This time we let ψ_0 hold iff

$$\exists m(m \text{ codes a proof of of } \varphi_n \text{ in } F_\sigma$$

$$\text{and no smaller } m' \text{ codes a proof of } \overline{\varphi_n}).$$

We define ψ_1 similarly with the roles of φ_n and $\overline{\varphi_n}$ interchanged.

As F_σ is a finite extension of PA, ψ_i is provable in F if true as they are Σ^0_1. Since F_σ is consistent not both are provable. Thus computably from F_σ we can decide which is provable, if any. If ψ_1 is provable let $F'_\sigma = F_\sigma \cup \{\overline{\varphi_n}\}$, and otherwise let $F'_\sigma = F_\sigma \cup \{\varphi_n\}$.

Now since the set of provable and refutable formulae of PA are effectively inseparable, there is a computable function g which, given a disjoint pair (A, B) of c.e. sets extending them, computes a sentence ν consistent with both of them. Our action is to set $F_{\sigma 0} = F'_\sigma \cup \{\nu\}$, and $F_{\sigma 1} = F'_\sigma \cup \{\overline{\nu}\}$.

Then if $F \leqslant_{\mathrm{T}} C$, we see that $\cup_{\sigma \preccurlyeq C} F_\sigma$ is a complete extension of PA of the same degrees as C. $\square$

The exact classification of the PA degrees is unknown. Here are some known results.

Theorem 5.18.5 (Jockusch and Soare [136]). *No consistent extension of PA can have minimal degree or incomplete c.e. degree.*

Proof. By the Scott Basis Theorem, if there was one of incomplete c.e. degree, then sets computable from an incomplete c.e. degree would form a basis for Π^0_1 classes. This contradicts Theorem 5.16.15. For the minimal degree case, consider the Π^0_1 class define by putting $A \oplus B$ into it if for all e, $A(2e) \neq \Phi_e(2e)$, $B(e) \neq \Phi_e(e)$, and $A(2e+1) \neq \Phi_e^B(2e+1)$. Then a member of this class $A \oplus B$ cannot be computable (by the first two conditions) and must have $A \not\leqslant_T B$ by the last one. Thus this class has no computable members nor ones of minimal degree. □

Theorem 5.18.6 (Jockusch and Soare [136, 137]). *PA degrees include low degrees and* $\mathbf{0}'$.

Proof. The existence of low PA degrees follows by the Low Basis Theorem. (Historically, this was the Jockusch and Soare's first application of the Low Basis Theorem.) By the c.e. basis theorem there must be one of c.e. degree and by the previous result it cannot be incomplete. □

5.19 Fixed-point free and diagonally noncomputable functions

A total function f is *fixed-point free* if $W_{f(e)} \neq W_e$ for all e. By the Recursion Theorem, no computable function is fixed-point free, and indeed the fixed-point free functions can be thought of as those that avoid having fixed points in the sense of the Recursion Theorem. As we will see, fixed-point free functions have interesting ramifications in both classical computability theory and algorithmic information theory. A related concept is that of a *diagonally noncomputable* (*DNC*) function, where the total function g is DNC if $g(e) \neq \Phi_e(e)$ for all e.

Lemma 5.19.1 (Jockusch, Lerman, Soare, and Solovay [134]). *The following are equivalent.*

(i) *The set A computes a fixed-point free function.*

(ii) *The set A computes a total function h such that $\Phi_{h(e)} \neq \Phi_e$ for all e.*

(iii) *The set A computes a DNC function.*

(iv) *For each e there is a total function $h \leqslant_T A$ such that $h(n) \neq \Phi_e(n)$ for all n.*

(v) *For each total computable function f there is an index i with Φ_i^A total and $\Phi_i^A(n) \neq \Phi_{f(i)}(n)$ for all n.*

Proof. We begin by proving the equivalence of (i)–(iii).

Clearly (i) implies (ii), since if $W_{h(e)} \neq W_e$ then $\Phi_{h(e)} \neq \Phi_e$.

To show that (ii) implies (iii), suppose that h is as in (ii). Define $\Phi_{d(u)}$ as in the proof of the Recursion Theorem. That is, $\Phi_{d(u)}(z)$ is $\Phi_{\Phi_u(u)}(z)$ if $\Phi_u(u)\downarrow$, and $\Phi_{d(u)}(z)\uparrow$ otherwise. As we have seen, we can choose d to be a total computable function. Let $g = h \circ d$. Note that $g \leqslant_{\mathrm{T}} A$. Suppose that $g(e) = \Phi_e(e)$. Then by (ii) we have

$$\Phi_{d(e)} \neq \Phi_{h(d(e))} = \Phi_{g(e)} = \Phi_{\Phi_e(e)} = \Phi_{d(e)},$$

which is a contradiction. So g is DNC.

To show that (iii) implies (i), fix a partial computable function ψ such that, for all e, if $W_e \neq \emptyset$ then $\psi(e) \in W_e$, and let q be partial computable with $\Phi_{q(e)}(q(e)) = \psi(e)$ for all e. Let $g \leqslant_{\mathrm{T}} A$ be DNC, and define $f \leqslant_{\mathrm{T}} A$ so that $W_{f(e)} = \{g(q(e))\}$. Suppose that $W_e = W_{f(e)}$. Then $W_e \neq \emptyset$, so $\psi(e) \in W_e$, which implies that $\psi(e) = g(q(e))$. But g is DNC, so $g(q(e)) \neq \Phi_{q(e)}(q(e)) = \psi(e)$, which is a contradiction. So f is fixed-point free.

We now prove the equivalence of (iv) and (v).

To show that (iv) implies (v), let f be a computable function, and let e be such that $\Phi_e(\langle i,j\rangle) = \Phi_{f(i)}(j)$. Fix $h \leqslant_{\mathrm{T}} A$ satisfying (iv) for this e. For all i and n, let $h_i(n) = h(\langle i,n\rangle)$. Then there is a total computable function p with $\Phi^A_{p(i)} = h_i$ for all i. By the relativized form of the Recursion Theorem, there is an i such that $\Phi^A_{p(i)} = \Phi^A_i$. Then Φ^A_i is total, and

$$\Phi^A_i(n) = \Phi^A_{p(i)}(n) = h(\langle i,n\rangle) \neq \Phi_e(\langle i,n\rangle) = \Phi_{f(i)}(n).$$

To show that (v) implies (iv), assume that (iv) fails for some e. Let $f(i) = e$ for all i. Then for all i such that Φ^A_i is total, there is an n such that $\Phi^A_i(n) = \Phi_e(n) = \Phi_{f(i)}(n)$, so (v) fails.

Finally, we show that (iii) and (iv) are equivalent.

To show that (iii) implies (iv), given e, let f be a total computable function such that $\Phi_{f(n)}(x) = \Phi_e(n)$ for all n and x. Let $g \leqslant_{\mathrm{T}} A$ be DNC and let $h = g \circ f$. Then

$$\Phi_e(n) = \Phi_{f(n)}(f(n)) \neq g(f(n)) = h(n).$$

To show that (iv) implies (iii), let e be such that $\Phi_e(n) = \Phi_n(n)$ for all n, and let h be as in (iv). Then $h(n) \neq \Phi_e(n) = \Phi_n(n)$ for all n, so h is DNC. $\square$

Further characterizations of DNC functions, including ones in terms of Kolmogorov complexity, will be discussed in Chapter 11, particularly in Theorem 11.11.7. For DNC functions with range $\{0,1\}$, we have the following result.

Lemma 5.19.2 (Jockusch and Soare [136], Solovay (unpublished))**.** *A degree is PA iff it computes a $\{0,1\}$-valued DNC function.*

Proof. It is straightforward to define a Π^0_1 class $\mathcal{P}$ such that the characteristic function of any element of $\mathcal{P}$ is a $\{0,1\}$-valued DNC function. So by the Scott Basis Theorem, every PA degree computes a $\{0,1\}$-valued DNC function.

Now let A be a set such that there is a $\{0,1\}$-valued DNC function $g \leqslant_T A$. Then g is the characteristic function of a set separating $\{e : \Phi_e(e) = 1\}$ and $\{e : \Phi_e(e) = 0\}$. These two sets are effectively inseparable, so by Lemma 5.18.3, g has PA degree. Since the class of PA degrees is closed upwards, so does A. $\square$

The following is a classic result on the interaction between prefix-free functions and computable enumerability.

Theorem 5.19.3 (Arslanov's Completeness Criterion [17])**.** *A c.e. set is Turing complete iff it computes a fixed-point free function.*

Proof. It is easy to define a total $\emptyset'$-computable fixed-point free function. For the nontrivial implication, let A be a c.e. set that computes a fixed-point free function f. By speeding up the enumeration of A, we can assume we have a reduction $\Gamma^A = f$ such that for all s and all $n \leqslant s$, we have $\Gamma^A(n)[s]\downarrow$.

For each n we build a set $W_{h(n)}$, with the total computable function h given by the Recursion Theorem. Initially, $W_{h(n)} = \emptyset$. If $n \in \emptyset'[s]$ then for every $x \in W_{\Gamma^A(h(n))[s]}[s]$, we put x into $W_{h(n)}$ if it is not already there.

We claim we can compute $\emptyset'$ from A using h. To prove this claim, fix n. If $n \in \emptyset'$, then consider the stage s at which n enters $\emptyset'$. If the computation $\Gamma^A(h(n))$ has settled by stage s, then every $x \in W_{\Gamma^A(h(n))}$ is eventually put into $W_{h(n)}$, while no other x ever enters $W_{h(n)}$. So $W_{f(h(n))} = W_{\Gamma^A(h(n))} = W_{h(n)}$, contradicting the choice of f. Thus it must be the case that the computation $\Gamma^A(h(n))$ has not settled by stage s.

So, to decide whether $n \in \emptyset'$, we simply look for a stage s by which the computation $\Gamma^A(h(n))$ has settled (which we can do computably in A because A is c.e.). Then $n \in \emptyset'$ iff $n \in \emptyset'[s]$. $\square$

The above argument clearly works for wtt-reducibility as well, so a c.e. set is wtt-complete iff it wtt-computes a fixed-point free function. Arslanov [18] has investigated similar completeness criteria for other reducibilities such as tt-reducibility.

Antonin Kučera realized that fixed-point free functions have much to say about c.e. degrees, even beyond Arslanov's Completeness Criterion, and, as we will later see, about algorithmic randomness. In particular, in [156] he showed that if $f \leqslant_T \emptyset'$ is fixed-point free, then there is a noncomputable c.e. set $B \leqslant_T f$. We will prove this result in a slightly stronger form after introducing the following notion.

Definition 5.19.4 (Maass)**.** A coinfinite c.e. set A is *promptly simple* if there are a computable function f and an enumeration $\{A_s\}_{s\in\mathbb{N}}$ of A such that, for all e,

$$|W_e| = \infty \implies \exists^\infty x\, \exists s\, (x \in W_e[s] - W_e[s-1] \wedge x \in A_{f(s)}).$$

The idea of this definition is that x needs to enter A "promptly" after it enters W_e. We will say that a degree is promptly simple if it contains a promptly simple set. This concept was introduced by Wolfgang Maass in e.g. [195], where he studied the automorphism group of the lattice of c.e. sets. The usual simple set constructions (such as that of Post's hypersimple set) yield promptly simple sets.

We can now state the stronger form of Kučera's result mentioned above.

Theorem 5.19.5 (Kučera [156]). *If $f \leqslant_{\mathrm{T}} \emptyset'$ is fixed-point free then there is a promptly simple c.e. set $B \leqslant_{\mathrm{T}} f$.*

Proof. Let $f \leqslant_{\mathrm{T}} \emptyset'$ be fixed-point free, and fix a computable approximation $\{f_s\}_{s \in \mathbb{N}}$ to f. We build $B \leqslant_{\mathrm{T}} f$ to satisfy the following requirements for all e:

$$R_e : |W_e| = \infty \implies \exists x\, \exists s\, (x \in W_e[s] - W_e[s-1] \,\wedge\, x \in B_s).$$

To see that these requirements suffice to ensure the prompt simplicity of B (assuming that we also make B c.e. and coinfinite), notice that it is easy to define a computable function f such that for all e and sufficiently large n, there is an i such that $W_i = W_e \cap \{n, n+1, \dots\}$ and for $x \geqslant n$, if $x \in W_e[s] - W_e[s-1]$ then $x \in W_i[f(s)] - W_i[f(s)-1]$. So if the requirements are satisfied, then we have, for each e and n, that

$$|W_e| = \infty \implies \exists x > n\, \exists s\, (x \in W_e[s] - W_e[s-1] \,\wedge\, x \in B_{f(s)}).$$

During the construction we will define an auxiliary computable binary function h, whose index is given by the Recursion Theorem.

At stage s, act as follows for each $e \leqslant s$ such that

1. R_e is not yet met,
2. some $x > \max\{2e, h(e)\}$ is in $W_e[s] - W_e[s-1]$, and
3. $f_s(h(e)) = f_t(h(e))$ for all t with $x \leqslant t \leqslant s$.

Enumerate x into B. Using the Recursion Theorem, let $W_{h(e)} = W_{f_s(h(e))}$.

We first verify that B is promptly simple. Clearly, B is c.e., and it is coinfinite because at most one number is put into B for the sake of each R_e, and that number must be greater than $2e$. As argued above, it is now enough to show that each R_e is met. So suppose that $|W_e| = \infty$. Let u be such that $f_t(h(e)) = f(h(e))$ for all $t \geqslant u$. There must be some $x > \max\{2e, h(e), u\}$ and some s such that $x \in W_e[s] - W_e[s-1]$. If R_e is not yet met at stage s, then x will enter B at stage s, thus meeting R_e.

We now show that $B \leqslant_{\mathrm{T}} f$. Let

$$q(x) = \mu s > x\, \forall y \leqslant x\, (f_s(y) = f(y)).$$

Then $q \leqslant_{\mathrm{T}} f$. We claim that $x \in B$ iff $x \in B_{q(x)}$. Suppose for a contradiction that $x \in B - B_{q(x)}$. Then x must have entered B for the sake of some R_e at some stage $s > q(x)$. Thus $f_s(h(e)) = f_t(h(e))$ for all t with $x \leqslant t \leqslant s$.

However, $x < q(x) < s$, so $W_{h(e)} = W_{f_s(h(e))} = W_{f(h(e))}$, contradicting the fact that f is fixed-point free. $\square$

Theorem 5.19.5 can be used to give a priority free solution to Post's Problem. To wit: Let A be a low PA degree (which exists by the Low Basis Theorem). By Lemma 5.19.2, A computes a DNC function, so by Theorem 5.19.1, A computes a fixed-point free function. Thus A computes a promptly simple c.e. set, which is noncomputable and low, and hence a solution to Post's Problem.

Prompt simplicity has some striking structural consequences.

Lemma 5.19.6 (Ambos-Spies, Jockusch, Shore, and Soare [6]). *Let* $\mathbf{a}$ *be a promptly simple degree. Then the following hold*

(i) *The degree* $\mathbf{a}$ *is* noncappable*, which means that if* $\mathbf{b} > \mathbf{0}$ *is a c.e. degree, then* $\mathbf{a} \cap \mathbf{b} \neq \mathbf{0}$.

(ii) *There is a low c.e. degree* $\mathbf{b}$ *such that* $\mathbf{a} \cup \mathbf{b} = \mathbf{0}'$.

Proof. (i) Suppose that A is promptly simple with witness function f and enumeration $A = \cup_s A_s$. Let B be a given noncomputable c.e. set. We must build $C \leqslant_{\mathrm{T}} A, B$ meeting

$$R_e : W_e \neq \overline{C}.$$

To do this we build auxiliary computably enumerable sets $V_e = W_{h(e)}$ whose indices $h(e)$ are given by the recursion theorem with parameters. From time to time in the construction we will issue a command like "put x into $V_{e,\text{ at}s}$, and this will filter through the overheads of the recursion theorem to be reflected that x will enter $W_{h(e),t(s)}$ at some $t(s) > s$ where the function $s \mapsto t(s)$ is primitive recursive. For ease of notation, we will simply regard $t(s) = s$ so that x enters $W_{h(e),\text{ at }s}$.

Then the strategy is simple. While $W_e \cap C_s = \emptyset$, we pick a follower $x = \langle y, e\rangle$ targeted for C and wait till the very stage where $x \in W_{e,s}$. we would declare x as then *active* and pick a new follower $x' > x$. For an active and unused x, should $B_u \restriction x \neq B_{u-1} \restriction x$, we will immediately enumerate x into $W_{h(e),\text{ at }u}$, and compute $f(u)$. If $x \in A_{f(u)}$ then put $x \in C_{f(u)+1}$. If $x \notin A_{f(u)}$, simply declare x as used.

It is easy to see that if we suppose that $W_e \neq \overline{C}$, since B is noncomputable, there will be infinitely many active and used followers. But then $W_{h(e)}$ is infinite, and hence by prompt simplicity one will enter C. Note that $C \leqslant B, A$ by permitting, and the computability of f.

(ii) Suppose that A is promptly simple as above with witness f. We will build a low set $B = \cup_s B_s$ in stages to satisfy the requirements

$$N_e : \exists^\infty t_s \Phi_{e,t_s}^{B_{t_s}}(e) \downarrow \rightarrow \Phi_e^B(e),$$

where $t_0, t_1, \ldots$ are a computable sequence of stages to be defined within the construction. Additionally, we must define a procedure $\Gamma^{A \oplus B} = K$.

This procedure is constructed as we would expect. To wit, if some n enters $K - K_s$ then we need to change one of $A_s \upharpoonright \gamma(n,s)$ or $B_s \upharpoonright \gamma(n,s)$ to change the current definition of $\Gamma^{A_s \oplus B_s}(n) = 0$ to 1. The problem, or course, is that some $\gamma(n,s) < \varphi_{e,s}(e)$, and we might be trying to preserve some $\Phi_{e,s}^{B_s}(e)\downarrow$ computation.

The main idea is to only really allow this to happen if $n < e$ and then let the finite injury method sort things out.

The construction is as follows. Again for the sake of each e we will build a test set $V_e = W_{h(e)}$. If, at some stage s we see that $\Phi_{e,s}^{B_s}(e)\downarrow$, *but* $\gamma(e-1,s) < \varphi_{e,s}(e)$, then we will immediately put $\gamma(e-1,s)$ into $V_{e,s+1}$ and raise $\gamma(k,s+1)$ to be large and fresh for all $k \geqslant e-1$. (Notice that this will entail either having a B-change or an A-change below $\gamma(e-1,s)$.)

Then we run the enumeration of A until stage $f(s)$ and see if some number $\leqslant \gamma(e-1,s)$ enters $A_{f(s)}$. If A so promptly permits $\gamma(e-1,s)$ then we need to do nothing else, save to set $t_s = s$, so that $\Phi_{e,t_s}^{B_{t_s}}(e)\downarrow$. If A does not promptly permit, then we will enumerate $\gamma(e-1,s)$ into $B_{s+1} - B_s$ (destroying the $\Phi_{e,s}^{B_s}(e)\downarrow$-computation) and set $t_s = s+1$, with the hat convention that now $\Phi_{e,t_s}^{B_{t_s}}(e)\uparrow$.

The full argument is a simple application of the finite injury method. □

Thus the cappable c.e. degrees and the promptly simple degrees are disjoint and all promptly simple degrees are *low cuppable*. Actually, Ambos-Spies, Jockusch, Shore, and Soare [6] proved that the promptly simple degrees and the cappable degrees form an algebraic decomposition of the c.e. degrees into a strong filter and an ideal. Furthermore, the low cuppable c.e. degrees coincide with the promptly simple degrees. The proofs of these results are technical, and would take us too far afield.

Corollary 5.19.7 (Kučera [156])**.** *Let* $\mathbf{a}$ *and* $\mathbf{b}$ *be* Δ_2^0 *degrees both of which compute fixed-point free functions. Then* $\mathbf{a}$ *and* $\mathbf{b}$ *do not form a minimal pair.*

Proof. By Theorem 5.19.5, each of $\mathbf{a}$ and $\mathbf{b}$ bounds a promptly simple degree. By Lemma 5.19.6, these promptly simple degrees are noncappable, and hence do not form a minimal pair. □

We remark that we will later see that for sufficiently random degrees $\mathbf{a}$ and $\mathbf{b}$, $\mathbf{a} \cap \mathbf{b} = \mathbf{0}$. We also later see in Theorem 11.5.1 that random reals are fixed point free, but the minimal pair phenomenon above only happens for 2-random reals. There are random reals below $\emptyset'$, but the Theorem above says that they do not form minimal pairs. Recently, Hirschfeldt, Nies, and Stephan [125] have shown that for random reals below $\mathbf{0}'$, the only sets Turing below both of them are what are called the "K-trivial" reals we meet in Chapter 15.

5.20 Direct coding into DNC degrees

In [157], Kučera gave a method of coding into DNC degrees. To do this he used the following variation of the notion of a DNC degree.

Definition 5.20.1. We say that a total function f is *generally noncomputable* (GNC) iff for all x, y,

$$f(\langle x, y\rangle) \neq \varphi_x(y).$$

The arguments above show that the degrees of members of $\{0,1\}$-valued GNC functions are exactly the degrees of $\{0,1\}$-valued DNC degrees; that is, the PA degrees. Let G_0 denote the Π^0_1 class of $\{0,1\}$-valued GNC functions.

Theorem 5.20.2 (Kučera [157])**.** *Let $A \subseteq G_0$ be a nonempty Π^0_1 class. Let C be any given set. Then there is a function $g \in A$ with $g(\langle x_0, y\rangle) = C(y)$ for all y, where the index x_0 can be found effectively from the index of A.*

Proof. Using the Recursion Theorem, choose x_0 such that, for all y, $\varphi_{x_0}(y)\downarrow$ iff there is a finite string τ such that

$$A \cap \{f : f(\langle x_0, j\rangle) = \tau(j),\ \text{for all } j < |\tau|\} = \emptyset.$$

We note that this is a Σ^0_1 condition. If there is a string τ satisfying it, let τ_0 be the first such τ. Then we note

(i) $\varphi_{x_0}(j) = 1 - \tau_0(j)$, for $j < |\tau_0|$, and

(ii) $\varphi_{x_0}(j) = 0$, for $j \geqslant |\tau_0|$.

Further observe that if there is such a τ_0, then each function $g \in G_0$ would satisfy $g(\langle x_0, j\rangle) = \tau_0(j)$, for $j < |\tau_0|$, and hence $A \cap \{f : f(\langle x_0, j\rangle) = \tau(j),\ \text{for all } j < |\tau|\} = A$. This would imply $A = \emptyset$, a contradiction. Thus we conclude there is no τ, and hence $\varphi_{x_0}(y)\uparrow$ for all y. Moreover, since no τ exists, for every string τ there is a function $g \in A$ such that

$$g(\langle x_0, j\rangle) = \tau(j)$$

for all $j < |\tau|$. By compactness, for all C there is a $g \in A$ such that for all y, $g(\langle x_0, y\rangle) = C(y)$. ☐

Kučera [157] (page 224, Remarks 2 and 3), make several remarks about the proof. He says that using $\langle x_0, y\rangle$ at these places where $\varphi_{x_0}(y)\uparrow$ for all y is an analog of the use of *flexible formulae* in axiomatizable theories first introduced and studied by A. Mostowski in [219] and Kripke in [154], with respect to Gödel's Incompleteness Theorem.

Kučera also remarks that the proof yields x_0 independently of the fact $A \neq \emptyset$. If $A = \emptyset$, then we will find some τ_0 with $\varphi_{x_0}(y) = 1 - \tau_0(y)$ for all $y < |\tau_0|$, but we can control $\varphi_{x_0}(y)$ for $y \geqslant |\tau_0|$. For an $y \geqslant |\tau_0|$, we can require that either

(i) $\varphi_{x_0}(y) = 0$ as in the proof, or

(ii) $\varphi_{x_0}(y)\uparrow$ *and*

$$\forall C \exists g \in B \forall y (y > |\tau_0| \to g(\langle x_0, y\rangle) = C(y)),$$

where B is a nonempty Π^0_1 class given in advance with $A \subseteq B \subseteq G_0$.

Furthermore, since in each nonempty $A \subseteq G_0$ we can thus effectively find infinitely many coding locations, we get an effective mapping from 2^ω to the nonempty subclasses of A which assigns a nonempty $\Pi^{0,C}_1$ class to each set C. If C is computable, then the image

$$A(C, x_0) = \{f : f \in A \wedge \forall y (f(\langle x_0, y\rangle = C(y))\}$$

is again a nonempty Π^0_1 class.

Coding is now possible as well as forcing with Π^0_1 classes. For instance, we can run the proof above to establish the following (using the proof of the Low Basis Theorem).

Corollary 5.20.3 (Kučera, unpubl.). *(i) Suppose that* $\mathbf{c}$ *is an low degree. Then there is a low PA degree* $\mathbf{a} \geqslant \mathbf{c}$.

(ii) Hence for any low degree $\mathbf{c}$ *there is a c.e. degree* $\mathbf{b}$ *such that* $\mathbf{b} \cup \mathbf{c}$ *is low.*

Proof. For (ii), by Kučera's priority free solution to Post's Problem, there is a c.e. degree below $\mathbf{a}$. □

We remark that (ii) above is of interest since Andrew Lewis [183] has constructed a single minimal complement (and hence a low_2 complement) for all the nonzero c.e. degrees. We remark that Kučera also used this method to construct an PA degrees of various jump classes.

5.21 Array noncomputability and traceability

In this section, we introduce the *array noncomputable degrees*, introduced by Downey, Jockusch, and Stob [84, 85]. The original definition of this class was in terms of *very strong arrays.* Recall that a collection of canonical finite sets $\mathcal{F} = \{D_{f(n)} : n \in \mathbb{N}\}$ is called a strong array if $D_{f(n)} \cap D_{f(m)} = \emptyset$ for all $m \neq n$.

Definition 5.21.1 (Downey, Jockusch, and Stob [84]). (i) A strong array $\{D_{f(n)} : n \in \mathbb{N}\}$ is called a *very strong array* if $|D_{f(n)}| > |D_{f(m)}|$ for all $n > m$.

(ii) For a very strong array $\mathcal{F} = \{D_{f(n)} : n \in \mathbb{N}\}$, we say that a c.e. set A is $\mathcal{F}$*-array noncomputable* ($\mathcal{F}$*-a.n.c.*) if for each c.e. set W there exists a k such that

$$W \cap D_{f(k)} = A \cap D_{f(k)}.$$

This definition was designed to capture a certain kind of multiple permitting construction. The intuition is that for A to be $\mathcal{F}$-a.n.c., A needs $|D_{f(k)}|$ many permissions to agree with W on $D_{f(k)}$. As we will see below, *up to degree*, the choice of very strong array does not matter.

In Downey, Jockusch, and Stob [85], a new definition of array noncomputability was introduced, based on domination properties of functions. We first recall that $f \leqslant_{\text{wtt}} A$ (for a function f and a set A) means that there are an index e and a computable function b such that $f = \Phi_e^A$ and, furthermore, for each n, the use of the computation $\Phi_e^A(n)$ does not exceed $b(n)$. It is easily seen that $f \leqslant_{\text{wtt}} \emptyset'$ iff there are computable functions $h(.,.)$ and $p(.)$ such that, for all n, we have $f(n) = \lim_s h(n,s)$ and $|\{s : h(n,s) \neq h(n,s+1)\}| \leqslant p(n)$.

Definition 5.21.2. A degree $\mathbf{a}$ is *array noncomputable* if for each $f \leqslant_{\text{wtt}} \emptyset'$ there is a function g computable in $\mathbf{a}$ such that $g(n) \geqslant f(n)$ for infinitely many n. Otherwise, $\mathbf{a}$ is *array computable*.

The following results give further characterizations of array noncomputability, and shows that, for c.e. degrees, the two definitions of array noncomputability coincide, and the first definition is independent of the choice of very strong array.

Fix a computable enumeration $\{K_s\}$ of K, and define $m_K(n)$ to be the least s such that $K \upharpoonright n = K_s \upharpoonright n$, where $A \upharpoonright n = \{i < n : i \in A\}$.

Theorem 5.21.3 (Downey, Jockusch, Stob [85])**.** *Let* $\mathbf{a}$ *be a degree, and let* $\{F_n\}$ *be a very strong array. Then the following three conditions are equivalent:*

(i) $\mathbf{a}$ *is a.n.c.*

(ii) There is a function h *computable in* $\mathbf{a}$ *such that* $h(n) \geqslant m_K(n)$ *for infinitely many* n.

(iii) There is a function r *computable in* $\mathbf{a}$ *such that for all* e *there exists* n *with* $W_e \cap F_n = W_{e,r(n)} \cap F_n$.

Proof. To prove $(ii) \rightarrow (i)$, let f be given with $f \leqslant_{\text{wtt}} K$, and let h satisfy (ii). We must find g computable in $\mathbf{a}$ with $g(n) \geqslant f(n)$ for infinitely many n. Fix e and a computable function b such that $f(n) = \{e\}^K(n)$ with use at most $b(n)$ for all n. We may assume without loss of generality that h and b are increasing. To compute $g(n)$, let s be minimal such that $s > h(b(n+1))$ and $\{e\}_s^{K_s}(n)\downarrow$ with use at most $b(n)$, and let $g(n) = \{e\}_s^{K_s}(n)$. Clearly g is computable in $\mathbf{a}$. Let n and k be such that $b(n) \leqslant k \leqslant b(n+1)$ and $h(k) \geqslant m_K(k)$, and let s be as in the definition of $g(n)$. We have $s \geqslant h(b(n+1)) \geqslant h(k) \geqslant m_K(k) \geqslant m_K(b(n))$, so K_s and K are the same below the use of $\{e\}_s^{K_s}(n)$. Hence $g(n) = f(n)$. Since there are infinitely many j with $h(j) \geqslant m_K(j)$, there are infinitely many n for which k exists as described above, and hence $g(n) = f(n)$ for infinitely many n.

The implication $(i) \to (iii)$ is obtained by applying (i) to the function $f(n) = (\mu s)(\forall e \leqslant n)[W_e \cap F_n = W_{e,s} \cap F_n]$.

It remains only to show that $(iii) \to (ii)$. Let r witness (iii). Then for each e there are infinitely many n with $W_e \cap F_n = W_{e,r(n)} \cap F_n$. (If this fails, one obtains a contradiction by defining an index e' so that there is no n with $W_{e'} \cap F_n = W_{e',r(n)} \cap F_n$. If $x \in F_n$ for one of the finitely many n such that $W_e \cap F_n = W_{e,r(n)} \cap F_n$, let $\varphi_{e'}(x)$ converge in strictly more than $r(n)$ steps, and otherwise let $\varphi_{e'}(x)$ converge (if ever) in at least the same number of steps as $\varphi_e(x)$.) Hence it suffices to show that there is an e such that, for all n, $\mu s[W_{e,s} \cap F_n = W_e \cap F_n] \geqslant m_K(n)$, since then it follows that r also witnesses (ii). The set W_e will be $V = \cup_s V_s$, where V_s is defined as follows. The idea is to make V change on F_n whenever an element $< n$ enters K. Let $V_0 = \emptyset$. Given V_s, let $c_{n,s}$ be the least element (if any) of $F_n - V_s$, and let

$$V_{s+1} = V_s \cup \{c_{n,s} : (\exists z < n)[z \in K_{s+1} - K_s]\}.$$

Note that $|F_n \cap V| \leqslant |\{s : (\exists i < n)[i \in K_{s+1} - K_s]\}| \leqslant n < |F_n|$ so that $c_{n,s}$ is defined for all n and s. It follows that $(\mu s)[V_s \cap F_n = V \cap F_n] \geqslant m_K(n)$. By the proof of the Slowdown Lemma Lemma **??**, (which requires only that the sets $V_{e,s}$ be computable uniformly in e and s and not necessarily finite), there exists e such that $W_e = V$ and, for all s, $W_{e,s} \subseteq V_s$. Hence for all n, $\mu s[W_{e,s} \cap F_n = W_e \cap F_n] \geqslant m_K(n)$, and the proof is complete. $\square$

Since the truth of (i) of Theorem 5.21.3 does not depend on the choice of the very strong array $\{F_n\}$, it follows that the truth of (iii) is also independent of the choice of $\{F_n\}$. The following result which shows in particular that the notion of array noncomputability in Definition 1.1 is equivalent for c.e. degrees to the definition of array noncomputability in via very strong arrays.

Theorem 5.21.4 (Downey, Jockusch, Stob [84, 85])**.** *Let* $\mathbf{a}$ *be a c.e. degree and let* $\{F_n\}$ *be a very strong array. Then the following are equivalent:*

(i) $\mathbf{a}$ *is a.n.c.*

(ii) There is a c.e. set A *of degree* $\mathbf{a}$ *such that* $(\forall e)(\exists n)[W_e \cap F_n = A \cap F_n]$.

(iii) For all increasing unbounded computable functions h*,* $f \leqslant_{\text{wtt}} \emptyset'$ *via a reduction* $\Gamma^K = f$ *such that* $\gamma(x) \leqslant h(x)$*,* $\exists^\infty x g(x) > f(x)$.

Hence for c.e. degrees, g.a.n.c. and a.n.c. coincide.

Corollary 5.21.5. *The anc degrees are closed upwards.*

Proof. To prove $(ii) \to (i)$, we assume that (ii) holds and show that (iii) of Theorem 5.21.3 holds with $r(n) = (\mu s)[A_s \cap F_n = A \cap F_n]$, where $\{A_s\}$ is a computable enumeration of A. To do this, for each e let $V_e = \{x : (\exists s)[x \in W_{e,s} - A_s]\}$. Then V_e is a c.e. set so by (ii) there exists n such that $A \cap F_n = V_e \cap F_n$. It is then easily seen that $W_e \cap F_n = W_{e,r(n)} \cap F_n$ as needed to prove (iii) of Theorem 5.21.3.

For $(i) \rightarrow (ii)$, assume that $\mathbf{a}$ is (g.) a.n.c.. Let $\{F_n\}$ be any very strong array. We shall construct a c.e. set A computable in $\mathbf{a}$ such that $(\forall e)(\exists n)[W_e \cap F_n = A \cap F_n]$. (This suffices to prove (ii) by Lemma 5.21.5.) Let $f(n) = (\mu s)(\forall e \leqslant n)[W_{e,s} \cap F_{\langle e,n\rangle} = W_e \cap F_{\langle e,n\rangle}]$. Clearly $f \leqslant_{\text{wtt}} K$, so there exists g of degree at most $\mathbf{a}$ with $g(n) \geqslant f(n)$ for infinitely many n. By the Modulus Lemma, there is a computable function $h(n,s)$ and a function p computable in $\mathbf{a}$ such that $g(n) = h(n,s)$ for all $s \geqslant p(n)$. We now define the c.e. set A. It suffices to ensure that if $n \geqslant e$ and $g(n) \geqslant f(n)$ then $A \cap F_{\langle e,n\rangle} = W_e \cap F_{\langle e,n\rangle}$. Whenever $h(n,s) \neq h(n,s+1)$ and $e \leqslant n \leqslant s$, put all elements of $W_{e,h(n,s+1)} \cap F_{\langle e,n\rangle}$ into A at stage s, and let A be the set of all numbers obtained in this fashion. If $x \in A \cap F_{\langle e,n\rangle}$, then $x \in A_{p(n)}$, so A is computable in p and hence A has degree at most $\mathbf{a}$. Suppose now that $n \geqslant e$ and $g(n) \geqslant f(n)$. It follows from the definition of f that $W_{e,g(n)} \cap F_{\langle e,n\rangle} = W_e \cap F_{\langle e,n\rangle}$. Choose s as large as possible so that $h(n,s) \neq h(n,s+1)$. (There is no loss of generality in assuming there is at least one such $s \geqslant n$.) Then $h(n,s+1) = g(n)$ and so $A_{s+1} \cap F_{\langle e,n\rangle} = W_{e,h(n,s+1)} \cap F_{\langle e,n\rangle} = W_{e,g(n)} \cap F_{\langle e,n\rangle} = W_e \cap F_{\langle e,n\rangle}$. Furthermore, by the maximality of s, no elements of $F_{\langle e,n\rangle}$ enter A after $s+1$, so $A \cap F_{\langle e,n\rangle} = W_e \cap F_{\langle e,n\rangle}$, as needed to complete the proof. □

If references to computable enumerability are deleted from Theorem 5.21.4, then $(i) \rightarrow (ii)$ still holds but $(ii) \rightarrow (i)$ fails.

Actually, we do not need to consider all wtt-reductions in the definition of array noncomputability, but can restric ourselves to reductions with use bounded by the identity function. Such reductions have been by Csima [57] and Soare [281] in connection with work of Nabutovsky and Weinberger [**?**] in differential geometry, and (in the slightly modified form in which we allow the use to be bounded by the identity function plus a constant) give rise to the notion of sw-reducibility that will be studied in Section 13.5.

Lemma 5.21.6 (Downey and Hirschfeldt, generalizing [84])**.** $\mathbf{a}$ *is a.n.c. iff for all increasing unbounded computable functions h, $f \leqslant_{\text{wtt}} \emptyset'$ via a reduction $\Gamma^K = f$ such that $\gamma(x) \leqslant h(x)$, $\exists^\infty x g(x) > f(x)$.*

Proof. One direction is clear, so we prove the "only if" direction. So suppose that

- for all increasing unbounded computable functions h, $f \leqslant_{\text{wtt}} \emptyset'$ via a reduction $\Gamma^K = f$ such that $\gamma(x) \leqslant h(x)$, $\exists^\infty x g(x) > f(x)$.

Let $q \leqslant_{\text{wtt}} K$. Suppose that the use of this wtt-reduction is the computable function $p(x)$. Thus $\Theta^K(x) = f(x)$ with use $\theta(x) \leqslant p(x)$. Without loss of generality, we can suppose that $x \leqslant p(x) < p(x+1)$, and that f is increasing. Let $\Phi^K(z) = \max_{\{p(x) \leqslant z\}} \Gamma^K(x)$. Then $\varphi(z) \leqslant z$. As • holds, there is a function g computable from A such that $\exists^\infty n(g(n) > \Phi^K(n)$.

Let $\widehat{g}(x) = g(p(x+2))$. Then $g(n) > \Phi^K(n)$ implies $g(p(x+2)) > \Gamma^K(x)$ where x is largest with $px) \leqslant n$ $g(p(x+2)) > \Gamma^K(x)$ where x is largest with $p(x) \leqslant n$. Hence $\exists^\infty x[\widehat{g}(x) > \Gamma^K(x) = f(x)]$. As p is computable, $\widehat{g} \leqslant_{\mathrm{T}} A$, as required. $\square$

It is well known that an arbitrary degree $\mathbf{a}$ is in $\overline{GL_2}$ iff for each function f computable in $\mathbf{a} \cup \mathbf{0}'$ there is a function g computable in $\mathbf{a}$ such that $g(n) \geqslant f(n)$ for infinitely many n. From this we immediately obtain the following:

Corollary 5.21.7. *(i) For any degree* $\mathbf{a}$*, if* $\mathbf{a} \in \overline{GL_2}$*, then* $\mathbf{a}$ *is (g.) a.n.c.*

(ii) For an c.e. degree $\mathbf{a}$ *if* $\mathbf{a}$ *is not (g.) a.n.c. then* $\mathbf{a}$ *is low*$_2$.

In [85], Downey, Jockusch and Stob demonstrated that many results for $\overline{GL_2}$ degrees extend to a.n.c. degrees. However, the a.n.c degrees do not coincide with the $\overline{GL_2}$ degrees, nor do the a.n.c. c.e. degrees coincide with the nonlow$_2$ c.e. degrees.

Theorem 5.21.8 (Downey, Jockusch, and Stob [84])**.** *There exist low array noncomputable c.e. degrees.*

Proof. The proof is a straightforward combination of lowness and array noncomputability. Let $\mathcal{F} = \{F_x : x \in \mathbb{N}\}$ be a standard very strong array with F_x having $x+1$ elements. The requirements are

$$R_e : \exists x(W_e \cap F_x = A \cap F_x).$$

$$N_e : \exists^\infty s(\Phi_e^A(e)\downarrow [s] \to \Phi_e^A(e)\downarrow).$$

For the sake of R_e we simply pick a fresh x to devote to making $W_e \cap F_x = A \cap F_x$. We do so in the obvious way. Whenever a new element occurs in $F_x \cap W_e[s]$ put it into $A[s+1]$. This will need to happen at most $|F_x|$ many times. Each time it happens initialize the lower priority N_j. Similarly each time we see a N_e computation $\Phi_e^A(e)\downarrow [s]$ we initialize the lower priority R_j, and the result will follow in the usual finite injury manner. $\square$

There are a number of other characterizations of the array noncomputable c.e. degrees (see [84, 85]). For example, the a.n.c. c.e. degrees are precisely those that bound c.e. sets A_1, A_2, B_1, B_2 such that $A_1 \cap A_2 = B_1 \cap B_2 = \emptyset$ and every separating set for A_1, A_2 is Turing incomparable with every separating set for B_1, B_2. In fact, they are the degrees that bound disjoint c.e. sets A, B that have no separating set of degree $\mathbf{0}'$.

In some ways they resemble high c.e. degrees. One of the hallmark results for the high degrees are the results of Martin [197] and Soare [278] which together show that the maximal sets are an invariant class for the high degrees: namely every maximal set has high c.e. degree, and every high c.e. degree contains a maximal set (Martin), and all maximal sets are automorphic (Soare). Recently Cholak, Coles, Downey and Herrmann [47] proved an analog of this result for another structure for the anc degrees. Let $\mathcal{L}(P)$ denote the lattice of Π^0_1 classes. We say that a Π^0_1 class C is *thin* iff for all Pi^0_1 subclasses C', there is a clopen set F such that $C' = F \cap C$. A Π^0_1 class is called *perfect* if it contains no computable members. Finally the degree of a Π^0_1 class $C = [T]$, the collection of paths through the computable binary tree T is the Turing degree of the nonextendible nodes.

Theorem 5.21.9 (Cholak, Coles, Downey, Herrmann [47]). *(i) Every perfect thin Π^0_1 class has anc c.e. degree. Furthermore* (Downey, Jockusch, Stob [84]) *each anc c.e. degree contains a perfect thin Π^0_1 class.*

(ii) Any two perfect thin Π^0_1 classes are automorphic under automorphisms of $\mathcal{L}(P)$.

We will not prove Theorem 5.21.9, as it would take us beyond the scope of this book.

Of great relevance to our investigations, especially in Chapter [**?**], will be the following concept.

Definition 5.21.10 (Zambella [329]; see [301]). A set A, and the degree of A, are *c.e.-traceable* if there is a computable function p (called a *bound*) such that, for each function $f \leqslant_{\mathrm{T}} A$, there is a computable function h (called a *trace* for f) satisfying, for all n,

(i) $|W_{h(n)}| \leqslant p(n)$ and

(ii) $f(n) \in W_{h(n)}$.

It is not hard to check that the above definition does not change if we replace "there is a computable function p" by "for every unbounded nondecreasing computable function p". Since one can uniformly enumerate all c.e. traces for a fixed bound p, there is a universal trace with bound p, that is, one that traces each function $f \leqslant_{\mathrm{T}} A$ on almost all inputs.

Ishmukhametov [129] gave the following characterization of the array computable c.e. degrees.

Theorem 5.21.11 (Ishmukhametov [129]). *A c.e. degree is array computable iff it is c.e. traceable.*

Proof. Let $g \leqslant_{\mathrm{T}} A$. Then there is a approximation to g which is tied to the enumeration of A; namely $\Gamma^A(x) = g(x)[s]$, and will not change unless $A \upharpoonright \gamma(x)$ changes. If this use is bounded by a computable p then there will be at most $p(x)$ many mind changes. Hence A is c.e and c.e. traceable iff every function $g \leqslant_{\mathrm{T}} A$ has a p-c.e. approximation. Hence this happens iff $\mathbf{a}$ is array computable, by Lemma 5.21.6. $\square$

Using this characterization, Ishmukhametov proved the following remarkable theorem, which shows that the a.n.c. c.e. degrees are definable in the c.e. degrees. A degree $\mathbf{m}$ is a *strong minimal cover* of a degree $\mathbf{a} < \mathbf{m}$ if for all degrees $\mathbf{d} < \mathbf{m}$, we have $\mathbf{d} \leqslant \mathbf{a}$.

Theorem 5.21.12 (Ishmukhametov [129]). *A c.e. degree is array computable iff it has a strong minimal cover.*

Definition 5.21.10 can be strengthened as follows.

Definition 5.21.13 (Terwijn and Zambella [304]). The set A, and the degree of A, are *computably traceable* if the conditions in Definition 5.21.10 with the $W_{h(n)}$ replaced by canonical finite sets $D_{h(n)}$.

If A is computably traceable then each function $g \leqslant_{\mathrm{T}} A$ is dominated by the function $f(n) = \max D_{h(n)}$, where h is a trace for f. Thus, every computably traceable degree is hyperimmune-free. One may think of computable traceability as a uniform version of hyperimmune-freeness. Terwijn and Zambella [304] showed that a simple variation of the standard construction of hyperimmune-free sets by Miller and Martin [218] (Theorem 5.14.3) produces continuum many computably traceable sets. Indeed, the proof we gave of Theorem 5.14.3 produces a computably traceable set.

6
Kolmogorov Complexity of Finite Strings

This chapter is a brief introduction to Kolmogorov complexity and the theory of algorithmic randomness for finite strings. We will concentrate on a few fundamental aspects, in particular ones that will be useful in dealing with our main topic, the theory of algorithmic randomness for infinite sequences. A much fuller treatment of Kolmogorov complexity can be found in Li and Vitányi [185]. We will not discuss the philosophical roots of Kolmogorov complexity. Again, we refer to Li and Vitányi [185] and to van Lambalgen [314, 315] for a thorough discussion of the foundations of the subject.

We will mainly deal with two kinds of Kolmogorov complexity: plain and prefix-free (both defined below). There are two notational traditions in algorithmic information theory. One uses C for plain Kolmogorov complexity and K for prefix-free Kolmogorov complexity (sometimes referred to as prefix Kolmogorov complexity). The other uses K for plain Kolmogorov complexity and H for prefix-free Kolmogorov complexity. In line with [185], we adopt the former convention.

6.1 Plain Kolmogorov complexity

The main idea behind the theory of algorithmic randomness for finite strings is that a string σ is random if and only if it is "incompressible", that is, the only way to generate σ by an algorithm is to essentially hardwire σ into the algorithm, so that the minimal length of a program to generate

σ is essentially the same as that of σ itself. For instance, $0^{1000000}$ can be described by saying that we should repeat 0 1000000 times, an algorithm that can easily be described in less than 1000000 bits. On the other hand, if we were to toss a coin 1000000 times and record the outcome as a binary string, we would not expect this string to have a very short description.

We formalize this notion, first due to Solomonoff [282] (in a sense), but independently to Kolmogorov [151], as follows. Let $f : 2^{<\omega} \to 2^{<\omega}$ be a partial computable function. Then the *Kolmogorov complexity* of a string σ with respect to f is

$$C_f(\sigma) := \min\{|\tau| : f(\tau) = \sigma\},$$

where this minimum is taken to be ∞ if the set is empty. We say that σ is *random relative to* f if $C_f(\sigma) \geqslant |\sigma|$.

Here we think of f as a "description system". Relative to this system, the string described (or generated) by a given string τ is $f(\tau)$ (if this value is defined). Thus σ is random relative to f if and only if it has no descriptions shorter than itself, relative to the description system f.

We would like to get rid of the dependence on f by choosing a *universal* description system. Such a system should be able to simulate any other description system with at most a small increase in the length of descriptions. Thus we fix a partial computable function $U : 2^{<\omega} \to 2^{<\omega}$ that is *universal* in the sense that, for each partial computable function $f : 2^{<\omega} \to 2^{<\omega}$, there is a string ρ_f such that

$$\forall \sigma\, [U(\rho_f \sigma) = f(\sigma)].$$

We call ρ_f the *coding constant* of f in U.

It is easy to check that for any partial computable function $f : 2^{<\omega} \to 2^{<\omega}$, we have

$$C_U(x) \leqslant C_f(x) + O(1),$$

so U is a universal description system in the sense discussed above. Thus, it makes sense to define the *(plain) Kolmogorov complexity* (or *C-complexity*) of a binary string σ to be

$$C(\sigma) := C_U(\sigma).$$

Of course, a different choice of U would yield a different definition of C, but the two definitions would agree up to an additive constant. We will denote by $C(\sigma, \tau)$ the quantity $C(\langle \sigma, \tau \rangle)$.

The following results are straightforward to prove.

Lemma 6.1.1 (Kolmogorov [151])**.** (i) *$C(\sigma) \leqslant |\sigma| + O(1)$.*

(ii) *$C(\sigma\sigma) \leqslant C(\sigma) + O(1)$.*

(iii) *If $h : 2^{<\omega} \to 2^{<\omega}$ is a computable function, then $C(h(\sigma)) \leqslant C(\sigma) + O(1)$.*

For example, letting $f : 2^{<\omega} \to 2^{<\omega}$ be the identity function, we have $C(\sigma) \leqslant C_f(\sigma) + O(1) = |\sigma| + O(1)$. The other two parts of the lemma are left as exercises.

For $n \in \mathbb{N}$, we can think of n as a binary string σ (namely, its representation in binary), and define $C(n) := C(\sigma)$. By part (iii) of Lemma 6.1.1, it would not really matter if we chose to define $C(n)$ to be $C(0^n)$, say. Note that $C(n) \leqslant \log n + O(1)$, since the length of the binary representation of n is $\log n$.

Suppose that $C(\sigma) = n$. Then there is a first string string τ of length n such that the computation of $U(\tau)$ converges with value σ. More precisely, there is a least stage s such that $U(\tau)[s]\downarrow = \sigma$ for one or more strings τ of length n. We fix the lexicographically least such τ and denote it by σ^*_C.

The following folklore lemma is easy to check, since from σ^*_C we can compute both σ and $C(\sigma) = |\sigma^*_C|$.

Lemma 6.1.2. $C(\sigma, C(\sigma)) = C(\sigma^*_C) \pm O(1)$.

We say that a binary string σ is *C-random* if it is random relative to U, that is, $C(\sigma) \geqslant |\sigma|$. The following theorem says that such strings exist.

Theorem 6.1.3 (Solomonoff [282], Kolmogorov [151])**.** *For each n there exists a σ with $|\sigma| = n$ and $C(\sigma) \geqslant n$.*

Proof. There are only $2^n - 1$ binary strings of length less than n, so there are at most $2^n - 1$ binary strings with Kolmogorov complexity less than n. $\square$

Notice that the same kind of counting argument shows that, for any k and n,

$$|\{\sigma \in 2^n : C(\sigma) \geqslant |\sigma| - k\}| > 2^n(1 - 2^{-k}).$$

For instance, the number of strings of length 1000 that are "half-random" (i.e., those with Kolmogorov complexity at least 500) is greater than $2^{1000}(1 - 2^{-500})$). Thus, heuristically, almost every string is half-random.

Therefore the following summarizes the facts around which the theory of Kolmogorov complexity is based.

Corollary 6.1.4. *There is a constant c_C such that*

(i) $C(\sigma) \leqslant |\sigma| + c_C$ *for all σ, and*

(ii) $|\{\sigma \in 2^n : C(\sigma) \leqslant n + c_C - j\}| = O(2^{n-j})$ *for all n and j.*

We would like to have $C(\sigma\tau) \leqslant C(\sigma) + C(\tau) + O(1)$, but this is not the case. The problem is that we cannot simply concatenate descriptions of σ and τ, since we would have no way of knowing where one description ends and the other begins. In fact, sufficiently long strings always have fairly compressible initial segments, as we now see.

Lemma 6.1.5 (Martin-Löf [198])**.** *Fix k. If μ is sufficiently long then there is an initial segment σ of μ such that $C(\sigma) < |\sigma| - k$. Thus, for any fixed d, we have $\mu = \sigma\tau$ such that $C(\mu) > C(\sigma) + C(\tau) + d$.*

Proof. Let ν be an initial segment of μ, and let n be such that ν is the n-th string in the length-lexicographic ordering of $2^{<\omega}$. Let ρ be the string consisting of the next n bits of μ following ν, and let $\sigma = \nu\rho$. To generate σ we need only know ρ, since the length n of ρ will then give us ν, so there is a constant c such that $C(\sigma) \leqslant |\rho| + c$, where c does not depend on the choice of ν.[1] On the other hand, $|\sigma| = |\nu| + |\rho|$, so if we choose ν so that $|\nu| > c + k$, then $C(\sigma) < |\sigma| - k$.

For the second part of the lemma, let c be such that $C(\tau) \leqslant |\tau| + c$ for all τ, and let $k = c+d$. Let μ be a sufficiently large string such that $C(\mu) \geqslant |\mu|$. By the first part of the lemma, we can write $\mu = \sigma\tau$ so that $C(\sigma) < |\sigma| - k$. Then $C(\mu) \geqslant |\mu| = |\sigma| + |\tau| > C(\sigma) + k + C(\tau) - c = C(\sigma) + C(\tau) + d$. □

On the other hand, we do have the following bound, which has been shown to be fairly sharp, using techniques like that in the proof of Lemma 6.1.5.

Lemma 6.1.6. $C(\sigma\tau) \leqslant C(\sigma) + C(\tau) + 2\log|\sigma| + O(1)$.

Proof. □

It is not hard to replace the $2\log|\sigma|$ term in the above lemma by $2\log C(\sigma)$. We also get a similar bound for $C(\sigma, \tau)$, with a very similar proof:

Lemma 6.1.7. $C(\sigma\tau) \leqslant C(\sigma, \tau) \leqslant C(\sigma) + C(\tau) + 2C(\sigma) + O(1)$.

We will examine these issues further when we look at symmetry of information in Section 6.3. We will also look further at complexity dips in Theorem 9.2.1.

We will use Kolmogorov complexity to study issues related to the relative algorithmic randomness of infinite sequences, but it has also found many other applications. For instance, there is a very well-developed theory applying incompressibility to avoid counting in combinatorial arguments. This method is extensively used for lower bound arguments in complexity theory, and is one of the principal reasons for much of the interest in C-complexity. We will not dwell on such applications of Kolmogorov complexity here, since they are not directly related to our central theme, but will give two short but interesting examples. For more on this subject, see Chapter 6 of [185].

---

[1] It may be useful to readers unfamiliar with this kind of argument to give a more detailed justification here: For a string τ, let $m = |\tau|$ and let $f(\tau) = \zeta\tau$, where ζ is the m-th string in the length-lexicographic ordering of $2^{<\omega}$. Then f is a computable function and $f(\rho) = \nu\rho = \sigma$. So $C_f(\sigma) \leqslant |\rho|$, and hence $C(\sigma) \leqslant |\rho| + c$, where c is the coding constant of f in our fixed universal function U.

How large is the i-th prime p_i? Let $m \in \mathbb{N}$ be such that p_i is the largest prime divisor of m. To describe m we need only know i and $\frac{m}{p_i}$. Hence, by Lemma 6.1.7,

$$C(m) \leqslant C(i, \tfrac{m}{p_i}) + O(1) \leqslant C(i) + C(\tfrac{m}{p_i}) + 2\log^{(2)} i + O(1) \leqslant \\ \log i + \log \tfrac{m}{p_i} + 2\log^{(2)} i + O(1).$$

(The $2\log^{(2)} i$ term comes from the fact that $\log^{(2)} i$ is the length of the binary representation of i.) Now, by the usual counting argument (as in Theorem 6.1.3), there are infinitely many choices of m so that $C(m) \geqslant \log m - O(1)$. For such m, we have

$$\log m \leqslant \log i + \log \tfrac{m}{p_i} + 2\log^{(2)} i + O(1) = \\ \log i + \log m - \log p_i + 2\log^{(2)} i + O(1),$$

which implies that

$$\log p_i \leqslant \log i + 2\log^{(2)} i + O(1).$$

Hence $p_i \leqslant O(i \log^2 i)$, which is pretty close to the real answer of $i \log i$.

Another classic application of Kolmogorov complexity is the construction of an immune set, that is, a set with no infinite c.e. subset. Let

$$A = \{\sigma : C(\sigma) \geqslant \frac{|\sigma|}{2}\}.$$

Then A is immune. Indeed, suppose that A has an infinite c.e. subset B. Let $h(n)$ be the first string of length greater than n to enter B. Then

$$C(h(n)) \geqslant \frac{|h(n)|}{2} > \frac{n}{2},$$

since $h(n) \in A$, but we can generate $h(n)$ given n simply by running the enumeration of B until a string of length greater than n appears, so

$$C(h(n)) \leqslant C(n) + O(1) \leqslant \log n + O(1).$$

For large enough n, this is a contradiction.

6.2 Conditional complexity

It is often useful to measure the compressibility of a string *given another string*. To do so, we fix a *universal oracle machine*, that is, an oracle Turing machine U such that for each oracle Turing machine M there is a string ρ_M such that

$$\forall X\, \forall\sigma\, [U^X(\rho_f \sigma) = M^X(\sigma)].$$

For strings σ and τ, the *(plain) Kolmogorov complexity of σ given tau*, is

$$C(\sigma \mid \tau) = \min\{|\mu| : U^\tau(\mu) = \sigma\}.$$

Notice that this notion of conditional Kolmogorov complexity reduces to uncoditional Kolmogorov complexity by letting $\tau = \lambda$, the empty string. That is, $C(\sigma) = C(\sigma \mid \lambda) \pm O(1)$.

In Lemma 6.1.2, we saw that $C(\sigma_C^*) = C(\sigma, C(\sigma)) \pm O(1)$, in Lemma 6.1.2. We can say a little more using relative complexity.

Lemma 6.2.1 (Folklore)**.** *$C(\sigma_C^* \mid \sigma) = C(C(\sigma) \mid \sigma) \pm O(1)$.*

Proof. Clearly $C(C(\sigma) \mid \sigma) \leqslant C(\sigma_C^* \mid \sigma) + O(1)$, since $C(\sigma) = |\sigma_C^*|$. For the other direction, consider the program that given strings σ, τ, searches through strings of length σ until it finds ν with $|\nu| = \sigma$ and $U(\nu) = \tau$. □

Easy counting arguments give us the following basic result relating the size of a finite set to the complexity of its elements.

Theorem 6.2.2 (Kolmogorov [151])**.**

(i) *Let $A \subset 2^\omega$ be finite. Then for each τ there is a $\sigma \in A$ such that $C(\sigma \mid \tau) \geqslant \log |A|$.*

(ii) *Let $B \subseteq 2^\omega \times 2^{<\omega}$ be an infinite computably enumerable set such that each set of the form $B_\tau = \{\sigma : \langle \sigma, \tau \rangle \in B\}$ is finite. Then for every τ and every $\sigma \in B_\tau$, we have $C(\sigma \mid \tau) \leqslant \log |B_y| + O(1)$, where the constant term is independent of both σ and τ.*

Proof. Part (i) is a counting argument. For part (ii), consider the computable function enumerating B. It will enumerate elements of B_y in some order $x_1, \ldots, x_{|B_y|}$. We can describe x by the program B, y and the i such that $x = x_i$. □

6.3 Symmetry of information

The *(plain) information content* of a string τ in a string σ is defined as

$$I_C(\sigma : \tau) := C(\tau) - C(\tau \mid \sigma).$$

This notion is meant to capture the difference between the intrinsic difficulty of producing τ given σ and that of producing τ *without* σ. The following famous result expresses concisely the relationship between $I_C(\sigma : \tau)$ and $I_C(\tau : \sigma)$.

Theorem 6.3.1 (Symmetry of Information, Levin and Kolmogorov [332])**.**

$$I_C(\sigma : \tau) = I_C(\tau : \sigma) \pm O(\log C(\sigma, \tau)).$$

Note that it follows immediately from this result that $I_C(\sigma : \tau) = I_C(\tau : \sigma) \pm O(\log n)$, where $n = \max\{|\tau|, |\sigma|\}$.

Theorem 6.3.1 follows easily from the following restated version.

Theorem 6.3.2 (Symmetry of Information – Restated)**.**

$$C(\sigma, \tau) = C(\tau \mid \sigma) + C(\sigma) \pm O(\log C(\sigma, \tau)).$$

To obtain Theorem 6.3.1 from Theorem 6.3.2, we argue as follows. We have

$$C(\sigma, \tau) = C(\tau \mid \sigma) + C(\sigma) \pm O(\log C(\sigma, \tau))$$

and

$$C(\tau, \sigma) = C(\sigma \mid \tau) + C(\tau) \pm O(\log C(\tau, \sigma)).$$

But $C(\sigma, \tau) = C(\tau, \sigma) \pm O(1)$, so

$$\begin{aligned} &I_C(\sigma : \tau) = C(\tau) - C(\tau \mid \sigma) = \\ &\qquad C(\sigma) - C(\sigma \mid \tau) \pm O(\log C(\sigma, \tau)) = I_C(\tau : \sigma) \pm O(\log C(\sigma, \tau)). \end{aligned}$$

We know proceed with the proof of Theorem 6.3.2.

Proof of Theorem 6.3.2. The following neat proof follows that of Li and Vitanyi [185]. First it is easy to see that $C(\sigma, \tau) \leqslant C(\sigma) + C(\tau \mid \sigma) \pm O(\log C(\sigma, \tau))$, since we can describe $\langle \sigma, \tau \rangle$ via a description of σ, of τ given σ and an indication of where to delimit the two descriptions.

For the hard direction, that

$$C(\tau \mid \sigma) + C(\sigma) \leqslant C(\sigma, \tau) + O(\log C(\sigma, \tau)),$$

define two sets

$$A = \{\langle \mu, \nu \rangle : C(\mu, \nu) \leqslant C(\sigma, \tau)\}$$

and

$$A_\mu = \{\nu : \langle \mu, \nu \rangle \in A\}.$$

The set A is finite and computably enumerable given $C(\sigma, \tau)$, as is A_μ for each μ. Hence, one can generate τ given σ and $C(\sigma, \tau)$, along with τ's place in the enumeration order of A_σ. Thus

$$C(\tau \mid \sigma) \leqslant \log |A_\sigma| + 2 \log C(\sigma, \tau) + O(1).$$

Let $N = 2^{\log |A_\sigma| - 1}$. (Recall our convention that when we write $\log n$, we mean the base 2 logarithm of n, rounded up to the nearest integer. Thus we do not necessarily have $N = |A_\sigma|/2$. However, it is the case that $N \leqslant |A_\sigma|$.) Since $\log N = \log |A_\sigma| - 1$, we have

$$C(\tau \mid \sigma) \leqslant \log N + O(\log C(\sigma, \tau)). \tag{6.1}$$

Now consider the set $B = \{\mu : |A_\mu| \geqslant N\}$. Clearly, $\sigma \in B$. Furthermore, B is c.e. given $C(\sigma, \tau)$ and N, and

$$|B| \leqslant \frac{|A|}{N} \leqslant \frac{2^{C(\sigma,\tau)}}{N}.$$

Thus, to generate σ, we need only have $C(\sigma, \tau)$ and N, along with σ's position in the enumeration order of B. Now, N can be specified with $\log^{(2)} |A_\sigma|$ many bits, so

$$C(\sigma) \leqslant \log \frac{2^{C(\sigma,\tau)}}{N} + 2\log^{(2)} |A_\sigma| + 2\log C(\sigma, \tau) + O(1).$$

But $|A_\sigma| \leqslant 2^{C(\sigma,\tau)}$, so

$$C(\sigma) \leqslant C(\sigma, \tau) - \log N + O(\log C(\sigma, \tau)). \tag{6.2}$$

Combining (6.1) and (6.2), we have

$$C(\sigma) + C(\tau \mid \sigma) \leqslant C(\sigma, \tau) + O(\log C(\sigma, \tau)),$$

as required. □

6.4 Information-theoretic characterizations of computability

In this section, we establish some combinatorial facts about the number of strings of a specified complexity, and show that one can use Kolmogorov complexity to provide information-theoretic characterizations of computability. We begin with a result of Loveland [189]. If A is a computable set then we can generate $A \upharpoonright n$ simply by running the computation of $A(i)$ for all $i < n$. Thus $C(A \upharpoonright n \mid n) = O(1)$, where the constant depends on A. Loveland's result is a converse to this fact.

Theorem 6.4.1 (Loveland [189]). *Let X be an infinite computable set. For each e, there are only finitely many sets A such that $C(A \upharpoonright n \mid n) \leqslant e$ for all $n \in X$, and each such A is computable.*

Proof. Fix e. For each n, let $k_n = |\{\tau \in 2^{\leqslant e} \mid U^n(\tau)\downarrow\}|$. Since $k_n < 2^{e+1}$ for all n, there is a largest m such that $k_n = m$ for infinitely many $n \in X$, and there is an l such that $k_n \leqslant m$ for all $n \in X$ such that $n \geqslant l$. Furthermore, there is a computable sequence $n_0 < n_1 < \cdots \in X$ such that $n_0 \geqslant l$ and $k_{n_i} = m$ for all i. Note that the sets

$$S_i = \{\mu \mid C(\mu \mid n_i) \leqslant e\}$$

are uniformly computable, since for each i we can wait until we see m strings $\tau \in 2^{\leqslant e}$ with $U^n(\tau)\downarrow$, at which point we know that S_i is exactly the collection of values of $U^n(\tau)$ for such τ.

Let T be the tree consisting of all σ such that

$$\forall n_i \leqslant |\sigma| \, (\sigma \upharpoonright n_i \in S_i).$$

If $C(A \upharpoonright n \mid n) \leqslant e$ for all $n \in X$ then in particular $A \upharpoonright n_i \in S_i$ for all i, so A is a path of T. But the tree T is computable, and has width at most m, since for each n_i there are at most m strings of length n_i on T, so T has only finitely many paths, which means that each path of T is computable. □

It is interesting that one *cannot* weaken the hypothesis of Loveland's theorem to simply say that $C = (A \upharpoonright n \mid n) \leqslant e$ for infinitely many n. This is again a result of Loveland. One way to prove this is to use so-called n-strings. There is a constant e depending on U alone such that for all strings of the form $x_n = n00\ldots0$ of length $2n$, $C(x||x|) < e$. This fact can be used to code any subset of $\mathbb{N}$ as a language α with $C(\alpha \upharpoonright n|n) \leqslant e$ for infinitely many n, and hence the collection of such languages is uncountable. (Specifically, for instance, take $A = \{n_1, n_2, \ldots\}$, in order of magnitude. Now take α as $x_{n_1} x_{x_{n_1} n_2} \cdots$.)

One of the keys to Loveland's theorem is the uniformity implicit in $C(x \upharpoonright n|y \upharpoonright n)$. A much more interesting theorem, which also gives an information-theoretical characterization of computability, is the following of Chaitin which indicates a hidden uniformity in C.

Theorem 6.4.2 (Chaitin [43])**.** *Suppose that $C(\alpha \upharpoonright n) \leqslant C(n) + O(1)$ for all n (for an infinite computable set of n), or $C(\alpha \upharpoonright n) \leqslant \log n + O(1)$, for all n. Then α is computable (and conversely). Furthermore for a given constant $O(1) = d$, there are only finitely many ($O(2^d)$) such α.*

The proof of Chaitin's theorem involves a lemma of independent interest. The proof below is along the lines of Chaitin's, but we hope that it is somewhat less challenging than the original.

Let $D : \Sigma^* \mapsto \Sigma^*$ be partial computable. Then a *D-description* of σ is a pre-image of σ.

Lemma 6.4.3 (Chaitin [43])**.** *Let $f(d) = 2^{(d+c)}$, $c = c_{d,D}$ to be determined. Then for each $\sigma \in \Sigma^*$,*

$$|\{q : D(q) = \sigma \wedge |q| \leqslant C(\sigma) + d\}| \leqslant f_D(d).$$

That is, the number of D-descriptions of length $\leqslant C(\sigma) + d$, is bounded by an absolute constant depending upon d, D alone (and not on σ)

Note that this applies in the special case that D is the universal machine. The intuition to the proof below is that if there are too many D-descriptions, then using a listing of these, we will be able to shorten the shortest description, which will be a contradiction.

Proof. Let σ be given, and $k = C(\sigma) + d$. For each m there are at most $2^{k-m} - 1$ strings with $\geqslant 2^m$ D-descriptions of length $\leqslant k$, since there are

$2^k - 1$ strings in total. Given k, m we can effectively list strings σ with $\geqslant 2^m$ D-descriptions of length $< k$, uniformly in k, m. (Wait till you see 2^m q's of length $\leqslant k$ with $D(q) = \nu$ and and then put ν on the list $L_{k,m}$.) The list $L_{k,m}$ has length $\leqslant 2^{k-m}$.

If σ has $\geqslant 2^m$ D-descriptions of length $\leqslant k$, then σ can be specified by

- m
- a string q of length 2^{k-m},

the latter indicating the position of σ in $L_{k,m}$. This description has length bounded by $2\log m + k - m + c$ where c depends only upon D. If we choose m large enough so that $2\log m + k - m + c < k - d$, we can then get a description of σ of length $< k - d = C(\sigma)$. If we let $f(d)$ be 2^n where n is the least m with $2\log m + c + d < m$ then we are done. ☐

The next lemma tells us that there are relatively few string with short descriptions, and the number depends on d alone.

Lemma 6.4.4 (Chaitin [43]). *There is a computable h depending only on d ($h(d) = O(2^d)$) such that, for all n,*

$$|\{\sigma : |\sigma| = n \wedge C(\sigma) \leqslant C(n) + d\}| \leqslant h(d).$$

Proof. Consider the partial computable function D defined via $D(p)$ is $1^{|U(p)|}$. Then let $h(d) = f_D(d)$, with f given by the previous lemma. Suppose that $C(\sigma) \leqslant C(n) + d$, and pick the shortest p with $U(p) = \sigma$. Then p is a D-description of n and $|p| \leqslant C(n) + d$. Thus there at most $f(d)$ many $p's$, and hence σ's. ☐

We remark that these lemmas are absolute cornerstones of algorithmic information theory, and are initially highly counter-intuitive. One would somehow expect that, since the number of strings of length n grows, the number of strings describing σ within e of the length of σ^*_C would *grow* as a function of n. The lemmas above say that this expectation is *not* the case and the number only depends on e!

Proof. (Of Theorem 6.4.2, concluded) Let

$$T = \{\sigma : \forall p \subseteq \sigma (C(p) \leqslant \log |p| + d)\}.$$

If n is random then $C(n) = \log n + c$, so that by the second lemma above, the number of strings in T of length n is $\leqslant h(d)$. (This is similar to the proof of Loveland's Theorem.) Taking the maximum number $\leqslant h(d)$ attained infinitely often, we can then construct a computable subtree of the c.e. tree T, upon which x must be a path. Note that the number of paths is bounded by $h(d)$. ☐

Merkle and Stephan [**?**] observed an extension of this result.

Definition 6.4.5 (Merkle and Stephan [?]). Suppose that $I \subseteq \mathbb{N}$. We say that A is extendably (C, c) trivial on I iff for all $n \in I$, $K(A \upharpoonright n) \leqslant K(n)+c$.

Now the proofs above extend. The collection of strings which are extendibly (C, c) trivial (to infinite ones) forms a Π_1^I class of bounded width, and hence we can show that the following relativization of Chaitin's Theorem holds.

Theorem 6.4.6 (Merkle and Stephan [?]). *Suppose that A is extendably (C, c) trivial on I. Then $A \leqslant_T I$.*

We remark that there are a number of other counting results along the lines of Lemma 6.4.4, many due to the Moscow School. We limit ourselves to one more which will be used later when we consider lowness. It is possible this result was known earlier than the given reference.

Lemma 6.4.7 (Figueira, Nies and Stephan [105]). *For all d,*

$$|\{y : C(x, y) \leqslant C(x) + d\}| = \mathcal{O}(d^4 2^d).$$

Proof. By Lemma 6.4.4,

$$|\{\sigma : |\sigma| = n \wedge C(\sigma) \leqslant C(n) + d\}| = \mathcal{O}(2^d).$$

Let z_x denote the x-th string in the llex ordering of $2^{<\omega}$. Let c be such that, for all x, $C(x) \leqslant z_x + c$. Consider the partial computable function $f(x, y, d)$ which enumerates all strings p with $C(p) \leqslant z_x + d + c$, until it finds $p = y$. Then if p is the i-th string so enumerated we make $f(x, y, d)$ the number i written in binary, with initial zeroes so that $|f(x, y, d)| = z_x + d + c + 1$. As with the proof of Chaitin's theorem, it is always possible to write $f(x, y, d)$ in this way as there are at most $2^{z_x+d+c+1}$ such p. If no such string exists, then $f(x, y, p) \uparrow$. Now let x and d be given and consider y with $C(x, y) \leqslant C(x) + d$. Since $C(x, y) \leqslant z_x + d + c$, $f(x, y, d) \downarrow$. We have

$$C(f(x, y, d)) \leqslant C(x, y)+2|d|+\mathcal{O}(1) \leqslant C(x)+d+2|d|+mO(1), \text{ and hence,}$$

$$C(f(x, y, d)) \leqslant C(z_x + d + c + 1) + d + 4|d| + \mathcal{O}(1).$$

The lest inequality follows since we can calculate x from $z_x + d + c + 1$ and d. Let $n = z_x + d + c + 1$, and $d' = d + 4|d| + \mathcal{O}(1)$. For fixed x and d, we know that $y \mapsto F(x, y, d)$ is injective, and hence

$$|\{y : C(x, y) \leqslant C(x)+d\}| \leqslant |\{\sigma : |\sigma| = n \wedge C(\sigma) \leqslant C(n)+d\} = \mathcal{O}(2^d) = \mathcal{O}(d^4 2^d).$$

□

6.5 Prefix-free machines and complexity

In this section, we introduce the notion of prefix-free Kolmogorov complexity. Our main motivation for working with this notion is that we will later use it in studying algorithmic randomness of infinite sequences. But Chaitin, Levin, and others have argued that prefix-free complexity is the correct notion of descriptive complexity even for finite strings.

Their argument for the inadequacy of plain Kolmogorov complexity is the following. The intended meaning in saying that τ is a description of σ is that the bits of τ contain all the information necessary to obtain σ. But a Turing machine M might produce σ by using not only the bits of τ but also its length. In this way, τ actually represents $|\tau| + \log|\tau|$ many bits of information. Indeed, this fact is exploited in the proof of Lemma 6.1.5.

Another way to make this argument is that, if M is allowed to use the length of τ in computing σ, then there must be some kind of *termination symbol* T (such as a blank space) on M's input tape following the bits of τ. Thus to output a word in the alphabet $\{0, 1\}$, our machine uses an input in the alphabet $\{0, 1, T\}$, which we may view as cheating. If one accepts this argument then one ought to try to circumvent this shortcoming of plain Kolmogorov complexity. First Levin [179] and later Chaitin [43] suggested using *prefix-free machines* to do so.

A set $A \subseteq 2^{<\omega}$ is *prefix-free* if it is an antichain with respect to the natural partial ordering of $2^{<\omega}$; that is, for all $\sigma \in A$ and all τ properly extending σ, we have $\tau \notin A$.[2] A *prefix-free function* is one whose domain is prefix-free. Similarly, a *prefix-free Turing machine* is one whose domain is prefix-free. It is usual to consider such a machine as being *self-delimiting*, which means that it has a one-way read head that halts when the machine accepts the string described by the bits read so far. The point is that such a machine is forced to accept strings without knowing whether there are any more bits written on the input tape. This is a purely technical device that forces the machine to have a prefix-free domain, but it also highlights how the use of prefix-free machines circumvents the use of length to gain more information.[3]

[2] Joe Miller has pointed out that set of valid telephone numbers is a real-world example of a prefix-free set. Note that this prefix-freeness allows us to give a phone number using only the alphabet $\{0,\ldots,9\}$. Anil Nerode has remarked that there have been phone numbering systems in the past that were not prefix-free. In such a system, to give a phone number, we also need a termination symbol, such as a blank space.

[3] Another way to achieve this goal is to require the complexity measure to be "continuous", which gives rise to the notions of *monotone complexity* and *process complexity.* We will discuss this notion in Sections 9.5 and 10.1.2. Another related notion, which we will not discuss in this book, is *uniform complexity*; see Li-Vitanyi [185] and Barzdins [23] for more details. Whilst we do examine many other complexities in this book we are not exhaustive. One other example is Loveland's *decision complexity*, Kd, where we consider a machine as a c.e. collection of pairs (σ, τ) thinking that σ is a description of τ, but now insist that if (σ, τ) occurs, then also $(\sigma, \widehat{\tau})$ must also be enumerated, for all prefixes $\widehat{\tau}$ of τ. As with most complexities, Lovelands's variant gives another insight into descriptional complexity, and we refer to Loveland [189] for more details. In passing, notice that $Kd(\sigma) \leqslant^{+} C(\sigma)$ and in fact it is possoble for $Kd(\sigma) << C(\sigma)$. Thus C is not the minimal complexity studied. For more on this see Uskensky and Shen [310]. We

Proposition 6.5.1. (i) *If Φ is a prefix-free partial computable function, then there is a prefix-free (self-delimiting) machine M such that M computes Φ.*

(ii) *There is a universal (self-delimiting) prefix-free machine.*

Proof. (i) Let Φ be partial computable and prefix-free. We build a self-delimiting machine M in stages. Initially, M works on the empty string. At a given point in the action of M, let σ be the string consisting of the bits read by M so far. The read head does not scan the next symbol unless it sees some extension τ of σ such that $\Phi(\tau)\downarrow$. At this point, M scans the next symbol i on the read tape. If the string $\sigma' := \sigma i$ remains an initial segment of τ, then M reads the next symbol and repeats the above process with σ' in place of σ. Otherwise, M goes back to waiting for a string τ' extending σ' upon which Φ halts. Of course, if the string σ consisting of the bits read by M so far is ever such that $\Phi(\sigma)\downarrow$, then M simply outputs $\Phi(\sigma)$.

(ii) Let Ψ_e be the partial computable function defined as follows: $\Psi_e[s] = \Phi_e[s]$ if $\mathrm{dom}(\Phi_e[s])$ is prefix-free, and $\Psi_e[s] = \Psi_e[s-1]$ otherwise. Then $\{\Psi_e\}_{e\in\mathbb{N}}$ is an enumeration of all (and only) the prefix-free partial computable functions. We can then define

$$\Psi(1^e 0\sigma) = \Psi_e(\sigma),$$

which is evidently prefix-free and universal. By part (i), there is a self-delimiting machine computing Ψ. □

Fix the enumeration $\Psi_0, \Psi_1, \dots$ of prefix-free partial computable functions given in part (ii) of the above proof. We will later need the following version of the Recursion Theorem.

Lemma 6.5.2 (Recursion Theorem for prefix-free machines)**.** *Let $h : 2^{<\omega} \times 2^{<\omega} \to 2^{<\omega}$ be a partial computable function such that for any y, the function $\lambda x.h(x,y)$ is prefix-free. From an index for h, we can compute an index e such that $\Psi_e = \lambda x.h(x,e)$.*

Proof. Let f be a total computable function such that $\Phi_{f(e)} = \lambda x.h(x,e)$ for all e. By the Recursion Theorem, there is an e (which can be computed from an index for f, and hence from an index for h) such that $\Phi_{f(e)} = \Phi_e$. Since $\Phi_{f(e)}$ is prefix-free, so is Φ_e, and hence $\Psi_e = \Phi_e$. Thus we have $\Psi_e = \Phi_{f(e)} = \lambda x.h(x,e)$. □

Let U be a universal prefix-free machine, as constructed in the proof of Propositon 6.5.1. Such a machine is *minimal* in the sense that, if M is any

remark that decisnion complexity and uniform complexity have much less well-developed theories than prefix-free and plain complexity.

prefix-free machine, then $C_U(\sigma) \leqslant C_M(\sigma) + O(1)$. Thus we can define the *prefix-free Kolmogorov complexity* of a string σ to be

$$K(\tau) := C_U(\sigma).$$

As in the case of plain complexity, the choice of universal prefix-free machine does not affect the definition of K, up to an additive constant.

As we did for plain complexity, we associate a natural number n with its binary representation σ and let $K(n) = K(\sigma)$.

As was the case for plain complexity, we have the following simple but important result.

Lemma 6.5.3. *If $h : 2^{<\omega} \to 2^{<\omega}$ is a computable function, then $K(h(\sigma)) \leqslant K(\sigma) + O(1)$.*

Suppose that $K(\sigma) = n$. Then there is a first string string τ of length n such that the computation of $U(\tau)$ converges with value σ. More precisely, there is a least stage s such that $U(\tau)[s]\downarrow= \sigma$ for one or more strings τ of length n. We fix the lexicographically least such τ and denote it by σ^*.

One of the most important facts about K is that it can be approximated from above. Specifically, let $K_s(\sigma) = \min\{|\tau| : U(\tau)[s]\downarrow= \sigma\}$. In other words, $K_s(\sigma)$ is the length of the shortest description of σ provided by U in at most s many steps. Of course, there may be no such description. However, it will be technically useful to assume that $K_s(\sigma)$ is always defined. It is easy to show that there is a c such that $K(\sigma) \leqslant 2|\sigma| + c$ for all σ. (We will give much better upper bounds below.) Thus, if there is no τ with $U(\tau)[s]\downarrow= \sigma$, then let $K_s(\sigma) = 2|\sigma| + c$. The important properties of this approximation are that

1. The function $(s, \sigma) \mapsto K_s(\sigma)$ is computable,
2. $K_s(\sigma) \geqslant K_{s+1}(\sigma)$ for all s and σ, and
3. $K(\sigma) = \lim_s K_s(\sigma)$ for all σ.

After discussing an important indirect way to build prefix-free machines, we will develop some further basic properties of prefix-free Kolmogorov complexity.

6.6 The KC Theorem

A central tool in building prefix-free machines is an effective interpretation of an inequality of Kraft [153].[4] Clearly, if $A \subset 2^{<\omega}$ is prefix-free

[4] This result is usually known as the *Kraft-Chaitin Theorem*, as it appears in Chaitin [43], but it appeared earlier in Levin's dissertation [177], as stated in Levin [179], where it is proven using Shannon-Fano codes (giving slightly weaker constants). There is also

then $\sum_{\sigma \in A} 2^{-|\sigma|} \leqslant 1$. Conversely, suppose that we have a sequence $\{d_i\}_{i\in\mathbb{N}}$ of natural numbers such that $\sum_i 2^{-d_i} \leqslant 1$. Since we are not arguing effectively, by rearranging the d_i if necessary, we can assume that $d_0 \leqslant d_1 \leqslant d_2 \leqslant \cdots$. Let σ_i be the leftmost string of length d_i that does not extend any σ_j with $j < i$. An easy induction shows that such a string always exists. Thus we see that, for any sequence $\{d_i\}_{i\in\mathbb{N}}$ of natural numbers such that $\sum_i 2^{-d_i} \leqslant 1$, there is a prefix-free sequence $\{\sigma_i\}_{i\in\mathbb{N}}$ of strings such that $|\sigma_i| = d_i$ for all i. The KC Theorem is an effectivization of this fact.

Theorem 6.6.1 (KC Theorem; Levin [177], Schnorr [266], Chaitin [43])**.** *Let $\langle d_i, \tau_i\rangle_{i\in\mathbb{N}}$ be a computable sequence of pairs (which we call* requests*), with $d_i \in \mathbb{N}$ and $\tau_i \in 2^{<\omega}$, such that $\sum_i 2^{-d_i} \leqslant 1$. Then there is a prefix-free machine M and strings σ_i of length d_i such that $M(\sigma_i) = \tau_i$ for all i. Furthermore, an index for M can be obtained effectively from an index for our sequence of requests.*

Proof. It is enough to define effectively a prefix-free sequence of strings $\sigma_0, \sigma_1, \ldots$ with $|\sigma_n| = d_n$. The following organizational device is due to Joe Miller. For each n, let $x^n = x_1^n \ldots x_m^n$ be a binary string such that $0.x_1^n \ldots x_m^n = 1 - \sum_{j\leqslant n} 2^{-d_j}$. We will define the σ_n so that the following holds for each n: for each m with $x_m^n = 1$ there is a string μ_m^n of length m so that $S_n := \{\sigma_i : i \leqslant n\} \cup \{\mu_m^n : x_m^n = 1\}$ is prefix-free.

We begin by letting σ_0 be 0^{d_0}. Notice that $x_m^0 = 1$ iff $0 < m \leqslant d_0$, so if we define $\mu_m^0 = 0^{m-1}1$, then $\{\sigma_0\} \cup \{\mu_m^0 : x_m^0 = 1\}$ is prefix-free.

Now assume we have defined $\sigma_0, \ldots, \sigma_n$ and μ_m^n for $x_m^n = 1$ so that $S_n = \{\sigma_i : i \leqslant n\} \cup \{\mu_m^n : x_m^n = 1\}$ is prefix-free.

If $x_{d_{n+1}}^n = 1$ then x^{n+1} is the same as x^n except that $x_{d_{n+1}}^{n+1} = 0$. So we can let $\sigma_{n+1} = \mu_{d_{n+1}}^n$ and $\mu_m^{n+1} = \mu_m^n$ for all $m \neq d_{n+1}$, and then $S_{n+1} = \{\sigma_i : i \leqslant n+1\} \cup \{\mu_m^{n+1} : x_m^{n+1} = 1\}$ is equal to S_n, and hence is prefix-free.

Otherwise, find the largest $j < d_{n+1}$ such that $x_j^n = 1$. Such a j must exist since otherwise $1 - \sum_{j\leqslant n} 2^{-d_j} < 2^{-d_{n+1}}$, which would mean that $\sum_{j\leqslant n+1} 2^{-d_j} > 1$. In this case x^{n+1} is the same as x^n except for positions $j, \ldots, d_{n+1}$, where we have $x_j^{n+1} = 0$ and $x_m^{n+1} = 1$ for $j < m \leqslant d_{n+1}$. Let $\sigma_{n+1} = \mu_j^n 0^{d_{n+1}-j}$. For $m < j$ or $m > d_{n+1}$, let $\mu_m^{n+1} = \mu_m^n$, and for $j < m \leqslant d_{n+1}$, let $\mu_m^{n+1} = \mu_j^n 0^{m-j-1}1$. Then $S_{n+1} = \{\sigma_i : i \leqslant n +$

a version of it in Schnorr [266], namely Lemma 1 on page 380. In Chaitin [42], where the first proof, explicitly for prefix-free complexity, seems to appear, the key idea of that proof is attributed to Nick Pippinger. Thus perhaps we should refer to the theorem by the rather unwieldy name of Kraft-Levin-Schnorr-Pippinger-Chaitin Theorem. Instead, we will refer to it as the KC Theorem. Since it is an effectivization of Kraft's inequality, readers should feel free if they wish to regard the initials as coming from "Kraft's inequality (Computable version)".

$1\} \cup \{\mu_m^{n+1} : x_m^{n+1} = 1\}$ is the same as S_n except that μ_j^n is replaced by a pairwise incomparable set of superstrings of μ_j^n. This fact clearly ensures that S_{n+1} is prefix-free.

This completes the definition of the σ_i. Each finite subset of $\{\sigma_0, \sigma_1, \ldots\}$ is contained in some S_n, and is hence prefix-free. Thus the whole set is prefix-free. Since the σ_i are chosen effectively, we can define a prefix-free machine M by letting $M(\sigma_i) = \tau_i$ for each i. □

We call an effectively enumerated set of requests $\langle d_i, \tau_i \rangle_{i \in \mathbb{N}}$ such that $\sum_i 2^{-d_i} \leqslant 1$ a *KC set*. The *weight* of this set is $\sum_i 2^{-d_i}$. The beauty of using the KC Theorem to define a prefix-free machine is that we need only build a KC set, issuing requests and ensuring that the weight is kept less than or equal to 1; the prefix-free machine itself is built implicitly for us.

The following is a very useful immediate consequence of the KC Theorem.

Corollary 6.6.2. *Let $\langle d_i, \tau_i \rangle_{i \in \mathbb{N}}$ be a KC set. Then $K(\tau_i) \leqslant d_i + O(1)$.*

Proof. Let M be as in the KC Theorem. Then $K(\tau_i) \leqslant K_M(\tau_i) + O(1) \leqslant d_i + O(1)$. □

A simple application of the KC Theorem is the following. Let $\tau_0, \tau_1, \ldots$ be an effective enumeration of $2^{<\omega}$, with τ_0 being the empty string. Let $d_0 = 2$ and for $d > 0$ let $d_i = |\tau_i| + 2 \log |\tau_i| + 2$. Then

$$\sum_i 2^{-d_i} \leqslant \frac{1}{4} + \sum_{i>0} \frac{2^{-|\tau_i|}}{2|\tau_i|^2} = \frac{1}{4} + \sum_{n>0} 2^n \frac{2^{-n}}{2n^2} = \frac{1}{4} + \frac{1}{2} \sum_{n>0} \frac{1}{n^2} < 1.$$

Thus $\{\langle d_i, \tau_i \rangle\}_{i \in \mathbb{N}}$ is a KC set, so we have the upper bound

$$K(\tau) \leqslant |\tau| + 2 \log |\tau| + O(1).$$

We will improve on this estimate in Theorem 6.7.3 below.

The following is another useful application of the KC Theorem.

Proposition 6.6.3. *Let $\mathcal{A}_0, \mathcal{A}_1, \ldots \subset 2^\omega$ be pairwise disjoint uniformly Σ_1^0 classes. Then $K(n) \leqslant -\log \mu(A_n) + O(1)$.*

Proof. Recall from Observation 5.16.3 that there are uniformly c.e. sets $A_0, A_1, \ldots \subset 2^{<\omega}$ such that $\mathcal{A}_n = \bigcup \{[\sigma] : \sigma \in A_n\}$. Thus we can approximate $\mu(A_n)$ from below. That is, there is a computable function $f : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}^+$ such that $f(n, s+1) \geqslant f(n, s)$ for all n and s, and $\lim_s f(n, s) = \mu(A_n)$ for all n.

Build a KC set L as follows. For each n and k, if there is an s such that $f(n, s) \geqslant 2^{-k}$, then enumerate a request $\langle k+1, n \rangle$. The weight of L is bounded by $\sum_n \sum_{k \leqslant -\log \mu(A_n)} 2^{-(k+1)} \leqslant \sum_n \mu(A_n) \leqslant 1$, so L is indeed a KC set.

For each n, there is an s such that $f(n, s) \geqslant \frac{\mu(A_n)}{2}$, so L contains a request $\langle k+1, n \rangle$ with $k \leqslant -\log \mu(A_n) + 1$, which implies, by Corollary 6.6.2, that $K(n) \leqslant -\log \mu(A_n) + O(1)$. □

Corollary 6.6.4. *Let $A \subset 2^{<\omega}$ be prefix-free and let $A_0, A_1, \ldots \subset A$ be pairwise disjoint and uniformly c.e. Then $K(n) \leqslant -\log \sum_{\sigma \in A_n} 2^{-|\sigma|} + O(1)$.*

Corollary 6.6.5. *Let $B_0, B_1, \ldots \subseteq 2^{<\omega}$ be pairwise disjoint and c.e. Then $K(n) \leqslant -\log \sum_{\tau \in B_n} 2^{-K(\tau)} + O(1)$.*

Proof. Let $A_n = \{\sigma : U(\sigma) \in B_n\}$. The A_n are pairwise disjoint, uniformly c.e. subsets of the prefix-free set $\mathrm{dom}(U)$. Since $\tau^* \in A_n$ for all $\tau \in B_n$, Corollary 6.6.4 implies that

$$K(n) \leqslant -\log \sum_{\sigma \in A_n} 2^{-|\sigma|} \leqslant -\log \sum_{\tau \in B_n} 2^{-|\tau^*|} = -\log \sum_{\tau \in B_n} 2^{-K(\tau)}.$$

□

6.7 Basic properties of prefix-free complexity

Fix a universal prefix-free machine U. We begin by showing that the upper bound given at the end of the previous section cannot be improved too much. Note that, since $K = C_U$ and U is prefix-free, we have $\sum_{\sigma \in 2^{<\omega}} 2^{-K(\sigma)} \leqslant \sum_{U(\tau)\downarrow} 2^{-|\tau|} \leqslant 1$.

Lemma 6.7.1. *For any d, there are σ with $K(\sigma) > |\sigma| + \log|\sigma| + d$.*

Proof. Suppose that $K(\sigma) \leqslant |\sigma| + \log|\sigma| + d$ for all σ. Then

$$\sum_{\sigma \in 2^{<\omega}} 2^{-K(\sigma)} \geqslant \sum_{\sigma \in 2^{<\omega}} \frac{2^{-|\sigma|}}{(d+1)|\sigma|} = \sum_n 2^n \frac{2^{-n}}{(d+1)n} = \sum_n \frac{1}{(d+1)n} = \infty,$$

which is a contradiction. □

In fact, the argument in the above proof shows the following.

Corollary 6.7.2. *Let $f : 2^{<\omega} \to \mathbb{N}$ be such that $\sum_{\sigma \in 2^{<\omega}} 2^{-f(\sigma)} = \infty$. Then $K(\sigma) > |\sigma| + f(\sigma)$ for infinitely many σ.*

The most precise bound on $K(\sigma)$ is given in the following.

Theorem 6.7.3 (Chaitin [43]). $K(\sigma) \leqslant |\sigma| + K(|\sigma|) + O(1)$.

Proof. Consider the prefix-free machine M that, on input μ, looks for σ and τ such that $\mu = \tau\sigma$ and $U(\tau)\downarrow = |\sigma|$, and if such a pair is found, outputs σ. Note that, since U is prefix-free, there can be only one such pair σ, τ for any given μ. Furthermore, it is easy to check that M is prefix-free. So letting ν_σ be a minimal-length U-program for $|\sigma|$, we have

$$K(\sigma) \leqslant C_M(\sigma) + O(1) = |\nu_\sigma| + |\sigma| + O(1) = K(|\sigma|) + |\sigma| + O(1).$$

□

Chaitin [43] showed that the bound given in Theorem 6.7.3 is tight, as part of the following result.

Theorem 6.7.4 (Counting Theorem, Chaitin [43])**.**

(i) $\max\{K(\sigma) : |\sigma| = n\} = n + K(n) \pm O(1)$.

(ii) $|\{\sigma : |\sigma| = n \wedge K(\sigma) \leqslant n + K(n) - r\}| \leqslant 2^{n-r+O(1)}$, *where the constant* $O(1)$ *does not depend on* n *and* r.

Of course, part (i) of the Counting Theorem follows from part (ii). The most elegant way to prove part (ii) was given by Chaitin [43, 44], and works by way of the minimality of K as an information content measure.

Definition 6.7.5 (Chaitin, after Gács and Levin)**.** An *information content measure* is a partial function $F : 2^{<\omega} \to \mathbb{N}$ such that

1. $\sum_{F(\sigma)\downarrow} 2^{-F(\sigma)} \leqslant 1$ and
2. $\{\langle \sigma, k\rangle : F(\sigma) \leqslant k\}$ is c.e.

Notice that K is an information content measure.

As with prefix-free machines, we can enumerate the information content measures as $\{F_k : k \in \mathbb{N}\}$, in such a way that the corresponding sets given in item 2 of Definition 6.7.5 are uniformly c.e. We can then define a *minimal information content measure*

$$\widehat{K}(\sigma) := \min_{F_k(\sigma)\downarrow}\{F_k(\sigma) + k + 1\}.$$

It is easy to check that $\widehat{K}$ is indeed an information content measure, and that it is minimal in the sense that, for any information content measure F, we have $\widehat{K}(\sigma) \leqslant F(\sigma) + O(1)$. In particular, $\widehat{K}(\sigma) \leqslant K(\sigma) + O(1)$.

On the other hand, we can easily translate information content measures to prefix-free machines, using the KC Theorem. That is, given an information content measure F, the set $\{\langle \sigma, k\rangle : F(\sigma) \leqslant k\}$ is a KC set, so there is a prefix-free machine M such that for each σ there is a τ of length $F(\sigma)$ with $M(\tau) = \sigma$. In particular, this fact holds for $F = \widehat{K}$, so $K(\sigma) \leqslant \widehat{K}(\sigma) + O(1)$. Thus we see that $\widehat{K}$ and K are the same up to an additive constant, and so we will henceforth identify K and $\widehat{K}$ without further comment.

Chaitin found some clever proofs exploiting the minimality of K. Before turning to the proof of the Counting Theorem, here is an application of this idea. We give another proof, due to Chaitin, that $K(\sigma) \leqslant |\sigma| + K(|\sigma|) + O(1)$. We have

$$\sum_{\sigma \in 2^{<\omega}} 2^{-(|\sigma| + K(|\sigma|))} = \sum_n 2^n 2^{-(n+K(n))} = \sum_n 2^{-K(n)} \leqslant 1.$$

Thus the map $\sigma \mapsto |\sigma| + K(|\sigma|)$ is an information content measure, so $K(\sigma) \leqslant |\sigma| + K(|\sigma|) + O(1)$ by the minimality of K.

Proof of Theorem 6.7.4. As mentioned above, it is enough to prove part (ii). Let $F(n) = \lceil -\log(\sum_{|\sigma|=n} 2^{-K(\sigma)})\rceil$. Then

$$\sum_n 2^{-F(n)} \leqslant \sum_n \sum_{|\sigma|=n} 2^{-K(\sigma)} = \sum_{\sigma \in 2^{<\omega}} 2^{-K(\sigma)} \leqslant 1,$$

and $\{\langle n,k\rangle : F(n) \leqslant k\}$ is c.e. Thus F is an information content measure. (Here we are thinking of n as a binary string by identifying it with its binary representation.) So, by the minimality of K, there is a c such that

$$2^{-K(n)} \geqslant 2^{-F(n)-c} = O(\sum_{|\sigma|=n} 2^{-K(\sigma)}).$$

Now suppose that there are more than $2^{n-r+O(1)}$ strings of length n with $K(\sigma) < n + K(n) - r$. In other words, suppose that there exists an unbounded binary function f such that, for

$$S_{n,r} := \{\sigma : |\sigma| = n \wedge K(\sigma) < n + K(n) - r\},$$

we have $|S_{n,r}| \geqslant f(n,r)2^{n-r}$ for all n and r. Then for each r we have

$$2^{-K(n)} = O(\sum_{|\sigma|=n} 2^{-K(\sigma)}) \geqslant O(\sum_{\sigma \in F} 2^{-K(\sigma)}) \geqslant$$
$$O(f(n,r)2^{n-r}2^{-n-K(n)+r}) = O(f(n,r)2^{-K(n)}).$$

The constant in the O term does not depend on n and r, and $f(n,r)$ is unbounded, so we have a contradiction.[5] □

Thus we see that in certain aspects, such as upper bounds, prefix-free complexity is more difficult to deal with than plain complexity. There are trade-offs, however, in that prefix-free complexity simplifies certain results, such as the following. (Cf. the discussion beginning in the paragraph before Lemma 6.1.5.)

Lemma 6.7.6. $K(\sigma\tau) \leqslant K(\sigma, \tau) \leqslant K(\sigma) + K(\tau) + O(1)$.

We leave the proof as a useful exercise. For this and other similar facts about prefix-free complexity, see Li and Vitanyi [185] or Fortnow [107]. We will sharpen the estimate of Lemma 6.7.6 when we consider Symmetry of Information in Theorem 6.10.2.

[5]It would also be possible to prove this result using the KC Theorem. The key step in the above proof is showing that $2^{-K(n)} \geqslant O(\sum_{|\sigma|=n} 2^{-K(\sigma)})$. We can do this by monitoring $\lceil -\log(\sum_{|\sigma|=n} 2^{-K(\sigma)})\rceil$, and each time this value drops to a new low p, enumerating a KC request $\langle p, n\rangle$. The details are left as an exercise.

6.8 Computable bounds for $K(x)$

Ignoring additive constants, we have seen that for plain complexity, there is a computable upper bound on $C(\sigma)$, namely $|\sigma|$, that is achieved infinitely often. The analogous bound on $K(\sigma)$, however, is $|\sigma| + K(|\sigma|)$, which is not computable, and approximations to this bound, such as $|\sigma| + 2\log|\sigma|$ or $|\sigma| + \log|\sigma| + 2\log^{(2)}|\sigma|$, are not achieved infinitely often. It is tempting to conjecture that there is no computable upper bound on $K(\sigma)$ that is achieved infinitely often. This conjecture is false, however.

Theorem 6.8.1 (Solovay, see Gács [118]). *There is a computable function* $g : 2^{<\omega} \to \mathbb{N}$ *such that*

1. $K(\sigma) \leqslant g(\sigma) + O(1)$ *and*

2. $K(\sigma) = g(\sigma)$ *for infinitely many* σ.

Proof. It will be notationally convenient to build $g : \mathbb{N} \to \mathbb{N}$ so that $K(n) \leqslant g(n)$ for almost all n and $K(n) = g(n)$ for infinitely many n. Since we can identify binary strings with natural numbers via Gödel numbers, this construction will be enough to establish the theorem. We will in fact first build an auxiliary function $\widehat{g}$, then modify it slightly to obtain g.

We can construct a Turing machine M such that, for each τ and n, if $U(\tau)[s]\downarrow= n$, then $M(\tau)[2s]\downarrow= \langle n, s\rangle$. Since U is universal, there is a computable function f such that if $U(\tau)[s]\downarrow= n$ then $U(\mu)[f(s)]\downarrow= \langle n, s\rangle$ for some μ with $|\mu| \leqslant |\tau| + O(1)$, which implies that $K_{f(s)}(\langle n, s\rangle) \leqslant K_s(n) + O(1)$.

Now define $\widehat{g}$ as follows. Given x, find the unique n and s such that $x = \langle n, s\rangle$, and let $\widehat{g}(x) = K_{f(s)}(x)$. Clearly, $K(x) \leqslant \widehat{g}(x)$ for all x.

On the other hand, suppose that x is one of the infinitely many integers of the form $\langle n, s\rangle$ with $K(n) = K_s(n)$. Then $\widehat{g}(x) = K_{f(s)}(x) \leqslant K_s(n) + O(1) = K(n) + O(1) \leqslant K(x) + O(1)$, the last inequality coming from the fact that the function $\langle n, s\rangle \mapsto n$ is computable. Thus, for such x, we have $\widehat{g}(x) = K(x) \pm O(1)$. So there is a constant $c \in \mathbb{Z}$ such that $K(x) = \widehat{g}(x) + c$ for infinitely many x. Define $g(m) = \widehat{g}(m) + c$. □

Note that there is a critical difference between the computable bound $|\sigma|$ for $C(\sigma)$ and the computable bound $g(\sigma)$ of Theorem 6.8.1. We know that for each n there is some string σ of length n with $C(\sigma) = |\sigma| \pm O(1)$. For g, however, we only know that there are infinitely many σ with $K(\sigma) = g(\sigma)$, and apparently only $\emptyset'$ can figure out where these σ are. This difference will be important, especially in Chapter 15, where we construct a class of sets (the K-*trivial* sets) that show that Chaitin's Theorem 6.4.2 fails for K in place of C.

6.9 The Coding Theorem and discrete semimeasures

Chaitin's information content measures are more or less the same as the computably enumerable discrete semimeasures introduced by Levin [179] and in Gács [114](and in some sense Solomonoff [282]).

Definition 6.9.1. A *discrete semimeasure* is a function $m : 2^{<\omega} \to \mathbb{R}^+ \cup \{0\}$ such that

$$\sum_{\sigma \in 2^{<\omega}} m(\sigma) \leqslant 1.$$

An equivalent way to think of a discrete semimeasure is to identify $2^{<\omega}$ with $\mathbb{N}$ and think of $\mathbb{N}$ as being our measure space. Or that $2^{<\omega}$ is the measure space and not Cantor Space. Notice that under this identification, all strings are incompatible. Clearly, the standard Lebesgue measure we have looked at so far is *not* a discrete semimeasure. The standard Lebesgue measure is a *continuous* measure. The concept of a continuous semimeasure is examined when we look at the complexity of reals in Chapter 9. The discrete Lebesgue measure is $\lambda(\sigma) = 2^{-2|\sigma|-1}$. Again, all computably enumerable discrete semimeasures are enumerable, and hence there is a minimal one, this time up to multiplicative constant. Putting all of the above discussion together, we have the following identification first articulated in Levin's Thesis, with something akin to it in Solomonoff [282]:

Theorem 6.9.2 (Coding Theorem, Solomonoff [282], Levin [179, 114], Chaitin [43])**.** $K(\sigma) = -\log m(\sigma) + O(1)$.

Actually, the above version of the Coding Theorem is a shortened form. The full version considers the probability that an arbitary machine outputs σ. To wit, we define the following.

Definition 6.9.3. Given a prefix-free machine D, let $Q_D(\sigma) = \mu(D^{-1}(\sigma))$

That is, $Q_D(\sigma)$ is the probability that D outputs σ. If D is the fixed universal machine we will write $Q(\sigma)$ for $Q_D(\sigma)$.

Theorem 6.9.4 (Coding Theorem-strong form)**.** $-\log m(\sigma) = -\log Q(\sigma) + O(1) = K(\sigma) + O(1)$.

Proof. We note that $Q(\sigma) \geqslant 2^{-K(\sigma)} = 2^{-|\sigma^*|}$, since $D(\sigma^*) = \sigma$. Therefore $-\log Q(\sigma) \leqslant K(\sigma)$. But, $\sum 2^{-log Q(\sigma)} \leqslant \sum_\sigma Q(\sigma) \leqslant 1$. Hence, by minimality of K, $-\log Q(\sigma) \geqslant K(\sigma) + O(1)$ and hence $-\log Q(\sigma) = K(\sigma) + O(1)$, as required. $\square$

Notice that from this proof, $-\log Q(\sigma)$ is a measure of complexity, and hence, by the minimality of K among measures of complexity, we know that $2^{-K(\sigma)} \leqslant Q(\sigma)$. By Theorem 6.9.4, we know that for some constant d,

$$2^{-K(\sigma)} \leqslant Q(\sigma) \leqslant d2^{-K(\sigma)}.$$

Thus we can often replace usage of K by Q.

One informal interpretation of the Coding Theorem is that if a string has many long descriptions then it will have a short description as well.

6.10 Prefix-free symmetry of information

As with plain complexity, we can explore the notion of symmetry of information for prefix-free Kolmogorov complexity. In the same spirit as for C we may define the *prefix-free information content* of τ in σ as

$$I_K(\sigma : \tau) = K(\tau) - K(\tau \mid \sigma).$$

Theorem 6.10.1 (Symmetry of Information, Levin and Gács [114], Chaitin [42])**.** $I_K(\langle x, K(x)\rangle : y) = I(\langle y, K(y)\rangle : x) + O(1)$.

Note that given the relationship $K(z, K(z)) = K(z*) + O(1)$, the Symmetry of Information Theorem for prefix-free complexity may be neatly rewritten as

$$I(x^* : y) = I(y^* : x).$$

As with the C case, Levin's Symmetry of Information Theorem follows from a reformulation:

Theorem 6.10.2 (Symmetry of Information, Levin and Gács [114], Chaitin [42])**.** $K(x, y) = K(x) + K(y|x, K(x)) + O(1) = K(x) + K(y|x^*) + O(1)$.

Proof. To prove Theorem 6.10.1 from Theorem 6.10.2, by Theorem 6.10.2, we have $K(x, y) = K(x) + K(y|x, K(x)) = K(y) + K(x|y, K(y)) + O(1)$, and hence,

$$K(y) - K(y|x, K(x)) = K(x) - K(x|y, K(y)) + O(1),$$

and Theorem 6.10.1 follows.

Now we turn to the proof of Theorem 6.10.2.

First we prove that

$$K(x, y) \leqslant K(x) + K(y|x, K(x)) + O(1).$$

Given x^* and $z = K^*(y|x, K(x))$, we can construct a prefix-free machine M which, upon input x^*z, will compute x and $K(x) = |x^*|$. It will then compute y from x and $K(x)$ and z

To finish we need to prove that

$$K(x, y) \geqslant K(x) + K(y|x, K(x)) + O(1).$$

To achieve this, we prove that

$$K(y|x^*) \leqslant K(x,y) - K(x) + O(1).$$

We run the computation of U assuming that exactly one string halts at each stage. Call this p_s at stage s. Then, at each stage s, compute $\langle x_s, y_s\rangle$ with

$$U(p_s) = \langle x_s, y_s\rangle.$$

By the Coding Theorem, Theorem 6.9.4, there is a constant c such that

$$2^{K(x)-c}(\sum_y Q(\langle x, y\rangle)) \leqslant 1,$$

for all x. (To see this, imagine we are building a machine V which, each time we see $U(p)\downarrow = \langle x, y\rangle$, declares a Kraft-Chaitin axiom $\langle |p|, x\rangle$. Then, relative to V, $Q_V(x) = \sum_y Q_U(\langle x, y\rangle)$, meaning that $\sum_y Q(\langle x, y\rangle) \leqslant Q(x) + O(1)$.)

We will now define a new conditional machine M using Kraft-Chaitin. With z on the oracle tape, M tries to compute z' with $U(z) = z'$, and hence with x^* on the tape, computes x. M then simulates the machine M_x with the Kraft-Chaitin set

$$\langle |p_t| - |x^*| + c, y_t\rangle,$$

for each p_t of the form $\langle x, y_t\rangle$. Let W denote the computably enumerable collection of such requirements.

Notice that $\sum_{t\in W} 2^{-(|p_t|-|x^*|+c)} \leqslant 2^{K(x)-c}(\sum_y Q(\langle x, y\rangle)) \leqslant 1$, and hence Kraft-Chaitin can be applied to M_x.

For each p with $U(p) = \langle x, y\rangle$, there is a $\widehat{p}$ with $U(\widehat{p}|x^*) = M_x(\widehat{p}) = y$, and with $|\widehat{p}| = |p| - K(x) + c$. This shows that

$$K(y|x^*) \leqslant K(x,y) - K(x) + O(1).$$

□

Another way to express Theorem 6.10.2 is

Corollary 6.10.3. $K(x,y) = K(x) + K(y|x^*) + O(1)$.

6.11 Defining prefix-free randomness

At this stage we ought to be ready to define prefix-free randomness. But now there seem two interpretations.

First our intuition was that ν is random means that the shortest description of ν is as long as it can be. We will define ν to be *strongly* Chaitin (or K-) random if $K(\nu) \geqslant |\nu| + K(|\nu|) + O(1)$.

Second, we could regard ν as random if it essentially needed to be hard-wired, and hence $K(\nu) \geqslant |\nu| + O(1)$. Let us call this *weakly* Chaitin random[6].

We will see in the next two Chapters that there is a big difference between the two notions of prefix-free randomness.

Regarding the relationship between plain complexity and prefix complexity, certainly every (prefix-free) K-machine is a (plain) C-machine and hence every C-random string is weakly K-random. The converse is definitely not true. In fact, in Corollary 9.3.11, we will see that, *for infinitely many n, there are strings x of length n and*

(i) $K(x) \geqslant n$

(ii) $C(x) \leqslant n - \log n$.

On the other hand, in the next Chapter we will see that every *strongly* Chaitin random string is C-random. (Theorem 7.3.3).

As in the last section, we can summarize the two cornerstones of the theory of prefix-free complexity;

Corollary 6.11.1. *(i)* $K(x) \leqslant |x| + K(|x|) + c$

(ii) $|\{x : |x| = n \wedge K(x) \leqslant n + K(n) + c - j\}| = O(2^{n-j})$.

6.12 Some basic finite sets

In this section, we prove some results of Solovay [284] giving estimates of the sizes of some basic sets associated with a universal prefix-free machine U. Recall that σ^* is the first string τ of length $K(\sigma)$ such that $U(\tau)\downarrow = \sigma$. (A more precise definition is given in Section 6.5.) Recall also that we write $f \sim g$ to mean that $f(n) = O(g(n))$ and $g(n) = O(f(n))$. We also slighlty abuse notation by writing, for instance, $f(n) \sim 2^n$ to mean that $f \sim n \mapsto 2^n$. We begin with the following result.

Theorem 6.12.1 (Solovay [284])**.** *Let*

(i) $p(n) = |\{\sigma : |\sigma| = n \wedge U(\sigma)\downarrow\}|$,

(ii) $P(n) = |\{\sigma : |\sigma| \leqslant n \wedge U(\sigma)\downarrow\}|$,

(iii) $p'(n) = |\{\sigma : |\sigma^*| = n\}|$, *and*

(iv) $d(n) = |\{\sigma : K(\sigma) \leqslant n\}|$.

---

[6]The reader should beware that these notions are *nonstandard*, and indeed Chaitin has used the term “strongly Chaitin random” for an apparent variation of Martin-Löf randomness.

Then

$$p(n) \sim P(n) \sim d(n) \sim 2^{n-K(n)}.$$

Furthermore, although the definition of σ^* is machine dependent, there is a number k such that

$$\sum_{n \leqslant j \leqslant n+k} p'(j) \sim 2^{n-K(n)}.$$

Proof. We begin by establishing the upper bound on $p(n)$. Let $A_n = \{\sigma : |\sigma| = n \wedge U(\sigma)\downarrow\}$. The A_n are pairwise disjoint, uniformly c.e. subsets of the prefix-free set $\mathrm{dom}(U)$. By Corollary 6.6.4,

$$K(n) \leqslant -\log \sum_{\sigma \in A_n} 2^{-|\sigma|} + O(1) = -\log(p(n)2^{-n}) + O(1),$$

so $p(n) = O(2^{n-K(n)})$.

We now turn to the lower bound on $p(n)$.

Lemma 6.12.2. *There is a set of strings S_n such that*

(i) $\#S_n \geqslant c2^{n-K(n)}$, *and*

(ii) $U(x) = x$ *if* $x \in S_n$.

This lemma will establish that $p_n \geqslant c2^{n-K(n)}$ and the lower bound $d_n \geqslant c2^{n-K(n)}$.

The proof of the lemma uses the Recursion Theorem. The machine M works as follows. On input x, M first tries to parse $x = yz$ with $y \in \mathrm{dom}(U)$. M then computes $U(y)$ ($= n$, say) and then reads $n - (\Pi_M + |y|)$ many additional bits getting z_1, say. If it reads to the end of z, it halts without output. If not, it halts with output $\Pi_M y z_1$.

Now, given n,

$$S_n = \{\Pi_M n^* w : |w| = n - K(n) - |\Pi_M|\}.$$

Then $U(\Pi_M n^* w) = M(n^* w) = \Pi_M n^* w$. Also, $\#S_n = 2^{n-K(n)-|\Pi_M|}$. This establishes the Lemma, and hence $p_n \sim 2^{n-K(n)}$, as required.

Since $p_n \leqslant P_n$, the lower bound $c2^{n-K(n)} \leqslant P_n$ is immediate. For the upper bound, we estimate:

$$P_n \leqslant \sum_{j=0}^{n} p_j \leqslant \sum_{j=0}^{n} C2^{j-K(j)} = \sum_{j=0}^{n} C2^{n-j-K(n-j)}.$$

Now, $K(n-j)+K(j)+O(1) \geqslant K(n)$. Thus $-K(n-j) \leqslant -K(n)+2\log j + O(1)$. Hence $P_n \leqslant \sum_{j=0}^{n} C2^{n-K(n)-j+2\log j+O(1)}$ and this is

$$\leqslant C_1 2^{n-K(n)}[\sum_{j=0}^{\infty} j^2 2^{-j}] \leqslant C_2 2^{n-K(n)},$$

as the series is convergent. Thus $P_n \leqslant C_2 2^{n-K(n)}$.

Trivially, $d_n \leqslant P_n$, since we can map each x with $K(x) \leqslant n$ to a minimal program, we also get the desired upper estimate on d_n, and hence $d_n \sim P_n 2^{n-K(n)}$.

We next consider p'_n. We can choose k such that $d_{n+k} \geqslant 2P_n d_n$. Indeed, $d_{n+k} \geqslant c2^{n+k+K(n+k)} \geqslant c_1 2^{n+k-K(n)-O(\log k)}$. Also, $d_n \leqslant C2^{n-K(n)}$. So it suffices to choose k so large that $c_1 2^{k-O(\log k)} \geqslant 2C$. But then there are at least $C2^{n-K(n)}$ many words x for which $n \leqslant K(x) \leqslant n+k$. This proves

$$\sum_{n \leqslant j \leqslant n+k} p'_j \geqslant C2^{n-K(n)}.$$

Note that the upper estimate is trivial from that for P_n. □

Solovay remarks that there is another way to state these results. We say that a y with $U(y) = x$ is a p-minimal program for x if $|y| \leqslant K(x) + q$. There is a q such that the number of q-minimal programs of length n is $\sim 2^{n-K(n)}$. To see this, by Theorem 6.4.2, we have seen that the number of q minimal programmes is $O(2^q)$, uniformly in x. Now let $a_1 q$ be such that

$$d_{n+q_1} \geqslant 2d_n.$$

Then we need the following lemma.

Lemma 6.12.3. *There is a q_2 such that if $n \geqslant K(x) + q$, then there is a y with $|y| = n$ and $U(y) = n$.*

Granted Lemma 6.12.3, if $n + q \geqslant K(x) \geqslant n$, then there is a program for x of length $n + q + q_2$ that is $q + q_2$ minimal. Hence the number of $q_1 + q_2$ minimal programs of length n is $\geqslant C2^{n-q_1-q_2-K(n-q_1-q_2)} \geqslant c2^{n-K(n)}$. Again the upper bound is trivial, and so, for q sufficiently large, the number of q minimal programs of length n is $\sim 2^{n-K(n)}$.

Proof. (of Lemma 6.12.3) We construct a mahine M that proceeds as follows. On input x M tries to parse it as $x = x_1 x_2 y$ with $x_1, x_2 \in \mathrm{dom}(U)$.

M then computes $w = U(x_1)$ and $n = U(x_2)$. Then M tries to read $n - |x_2| - |\Pi_M|$ many bits (if possible), and outputs w. Now we choose q_2 so that $n \geqslant q_2$ entails $n \geqslant K(n) + |\Pi_M|$. Then if $n \geqslant q_2$, and $|z| = n - K(n) - |\Pi_M|$,

$$U(\Pi_M x^* n^* z) = x, \text{ and,}$$

$$|\Pi_M x^* n^* z| = K(x) + n,$$

giving the lemma. □

Solovay remarks that the definition of p'_n is machine dependent, and he does not know of a natural prefix pree universal machine with $p'_n \sim 2^{n-K(n)}$. It is easy to give an example of a universal machine where $K(x)$ is always odd.

Next we turn to another group of results of Solovay [284], this time concerning

$$D_n = \{x : |x| \leqslant n \wedge U(x)\downarrow\}.$$

To begin with, note that given P_n and n we can compute D_n by simply waiting for all the programs of length $\leqslant n$ to be listed in the enumeration of the domain of U. Thus

$$K(D_n) \leqslant K(\langle n, P_n\rangle) + O(1) \leqslant K(n) + K(P_n|n) + O(1).$$

Now, $K(P_n|n) = K(P_n|n^*) + O(1)$. But $K(n - K(n)|n^*)$ is $O(1)$ and hence

$$K(P_n|n^*) \leqslant n - K(n) + K(n - K(n)|n^*) + O(1),$$

and hence

$$K(P_n|n^*) \leqslant n - K(n) + O(1).$$

The conclusion is that

$$K(D_n) \leqslant K(n) + (n - K(n)) + O(1) = n + O(1).$$

This estimate is sharp as we now see.

Theorem 6.12.4 (Solovay [284])**.** $K(D_n) = n + O(1)$.

Proof. Conside the following machine M. On input x it first parses $x = x_1 y$ with $x_1 \in \text{dom}(U)$ and computes $n = U(x_1)$. Then let E be the finite set of strings with Gödel number n. Let m be the length of the longest string in E. If $\ell = m - |\Pi_M| - |x_1| \geqslant 0$, M reads the next ℓ bits of x, giving y_1, say. Then if M sees that $\Pi_M x_1 y_1 \in E$, and outputs 0 if it is not, remaining undefined if it is. Thus, if $K(D_n) \leqslant n - |\Pi_M|$, and $x_1 = D_n^*$,

$$\Pi_M x_1 y_1 \in D_n \text{ iff } \Pi_M x_1 y_1 \notin D_n,$$

if $|y| = n - (|x_1| + |\Pi_M|)$. Consequently, $|K(D_n)| > n - |\Pi_M|$. By the estimate before the statement of the Theorem, we see that $K(D_n) = n + O(1)$. □

We can state one final result about D_n.

Theorem 6.12.5 (Solovay [284])**.** $K(D_n^*|D_n) = O(1)$.

Proof. We know that $K(x^*|x) = K(K(x)|x)$. But $K(D_n) = n + O(1)$. For all but finitely many n, by the proof of the previous lemma, the length of the longest string in D_n. Thus $K(n|D_n) = O(1)$, proving the result. □

6.13 The conditional complexity of σ^* given σ

In this section, we investigate how difficult it is to produce σ^* given σ. This question is equivalent to asking how difficult it is to produce $K(\sigma)$

given σ, as shown by the following lemma, which has the same proof as the analogous Lemma 6.2.1.

Lemma 6.13.1. $K(\sigma^* \mid \sigma) = K(K(\sigma) \mid \sigma) \pm O(1)$.

It follows from this result that $K(\sigma^* \mid \sigma) \leqslant \log|\sigma| + O(\log^{(2)}|\sigma|)$. The following result shows that this bound is not too far from optimal.

Theorem 6.13.2 (Gács [114], Solovay [284])**.** *For all sufficiently large n, there is a $\sigma \in 2^n$ such that*

$$K(\sigma^* \mid \sigma) \geqslant \log n - \log^{(2)} n - 3.$$

Before proving this theorem, we need a combinatorial lemma. To simplify notation, for the remainder of this section, we let $k(n) = \log n - \log^{(2)} n - 3$, and assume that n is sufficiently large for $k(n)$ to be positive.

Lemma 6.13.3. $\sum_{i<2^{k(n)}} n^{4i} < 2^n$.

Proof. Let $M = \sum_{i<2^{k(n)}} n^{4i}$. Then $M < \sum_{j<2^{k(n)+2}} n^j < n^{2^{k(n)+2}}$, so $\log M \leqslant 2^{k(n)+2} \log n$, and hence $\log^{(2)} M \leqslant k(n) + 2 + \log^{(2)} n = \log n - 1$. Thus $M < 2^n$. □

Corollary 6.13.4. *Suppose that 2^n is divided into $2^{k(n)}$ disjoint pieces $A_0, \ldots, A_{2^{k(n)}-1}$. Then there is an $i < 2^{k(n)}$ such that $|A_i| > n^{4i}$.*

For the least such i, we have $\sum_{j<i} |A_j| = O(n^{4(i-1)})$.

Proof. If $|A_i| \leqslant n^{4i}$ for all i, then

$$2^n = \sum_{i<2^{k(n)}} |A_i| \leqslant \sum_{i<2^{k(n)}} n^{4i} < 2^n.$$

Now let i be least such that $|A_i| > n^{4i}$. Then

$$\sum_{j<i} |A_j| \leqslant \sum_{j<i} n^{4j} = O(n^{4(i-1)}.$$

□

Proof of Theorem 6.13.2. Assume for a contradiction that $K(\sigma^* \mid \sigma) < k(n)$ for all $\sigma \in 2^n$.

We say that $\tau \in 2^{<k(n)}$ is *active for* σ if there is a μ such that $U^\sigma(\tau) = \mu$ and $U(\mu) = \sigma$. Note that for each $\sigma \in 2^n$, there is a τ that is active for σ such that $U^\sigma(\tau) = \sigma^*$.

Since the active strings all have length less than $k(n)$, the number of strings that are active for a given σ is in $[1, 2^{k(n)})$. Let

$$A_i = \{\sigma \in 2^n : \text{there are exactly } 2^{k(n)-i} \text{ many strings active for } \sigma\},$$

and let

$$S_i = \bigcup_{j<i} A_j.$$

Note that the S_i are uniformly c.e. (More precisely, we have a different collection of S_i for each n. All of these collections together are uniformly c.e.) Let $S_{i,s}$ be the stage s approximation to S_i.

By Corollary 6.13.4, there is a least i such that $|A_i| > n^{4i}$, and for this i we have $|S_i| = O(n^{4(i-1)})$.

Since $|A_i| > n^{4i}$, there must be a $\sigma \in A_i$ such that $K(\sigma) \geqslant 4i \log n$. Now suppose that we know the parameters i, $|S_i|$, and n. Then we can wait until a stage s such that $S_{i,s} = S_i$, and hence compute S_i. Every string entering S_{i+1} that is not in S_i is in A_i, so we can computably enumerate A_i. But once we know that $\sigma \in A_i$, we can find the $2^{k(n)-i}$ many strings τ active for σ, and compute $U^\sigma(\tau)$ for all such τ. One of these computations will produce σ^*, and hence we can determine $K(\sigma)$. Let σ be the first string to enter our enumeration of A_i with $K(\sigma) \geqslant 4i \log n$. We have just argued that we can compute σ given i, $|S_i|$, and n. Thus $K(\sigma) \leqslant K(i) + K(|S_i|) + K(n) + O(1)$, so to arrive at a contradiction, it is enough to show that $4i \log n \nleqslant K(i) + K(|S_i|) + K(n) + O(1)$ if n is sufficiently large.

Now, $K(n) \leqslant \log n + O(\log^2 n)$, and $i < 2^{k(n)}$, so

$$K(i) \leqslant k(n) + O(\log k(n)) = \log n - \log^2 n - 3 + O(\log(\log n - \log^2 n - 3)) \leqslant \log n + O(\log^2 n).$$

Finally, $|S_i| = O(n^{4(i-1)})$, so

$$K(|S_i|) \leqslant \log(|S_i|) + O(\log^2 |S_i|) \leqslant 4(i-1)\log n + O(\log^2 n).$$

Thus $K(i) + K(|S_i|) + K(n) = (4i-2)\log n + O(\log^2 n)$, and hence $4i \log n \nleqslant K(i) + K(|S_i|) + K(n) + O(1)$ for sufficiently large n, as desired. $\square$

The same proof also gives the analogous result for plain complexity.

Theorem 6.13.5 (Gács [114], Solovay [284]). *For all sufficiently large n, there is a $\sigma \in 2^n$ such that*

$$C(C(\sigma) \mid \sigma) \geqslant \log n - \log^{(2)} n - 3.$$

Note that this lower bound is close to the obvious upper bound $C(C(\sigma) \mid \sigma) \leqslant C(C(\sigma)) + O(1) \leqslant \log|\sigma| + O(1)$.

Chaitin conjectured the following which implies the above and is conjectured correct by Solovay:

Conjecture 6.13.6. *There is an infinite sequence σ_n of strings such that*

(i) $|\sigma_n| = n$

(ii) $K(\sigma_n) \sim n$

(iii) $K(\sigma^* \mid \sigma) \sim \log n$.

Solovay points out that the σ's we construct might well satisfy $K(\sigma_n) \leqslant \frac{n}{\log \log n}$, for example.

7
Some Theorems Relating K and C

In this chapter, we look at some of the fundamental relationships between prefix-free complexity and plain complexity. Most of this material, although of independent interest, will not be used directly in the rest of this book. Exceptions are Corollaries 7.3.3 and 7.3.8, which will be relevant in the discussion of Kolmogorov randomness and strong Chaitin randomness in Chapter 9.

Throughout this chapter, we fix a universal prefix-free machine U and a universal Turing machine V, and define K and C using these machines.

7.1 Levin's Theorem

Our first results relating K and C are due to Levin. They appeared in his dissertation [177]; the first published proof was in Gács [115].

Theorem 7.1.1 (Levin; see Gács [115])**.**

(i) $C(\sigma) = \min\{n : K(\sigma \mid n) \leqslant n\} \pm O(1)$.

(ii) $C(\sigma) = K(\sigma \mid C(\sigma)) \pm O(1)$.

Proof. Let $m_\sigma = \min\{n : K(\sigma \mid n) \leqslant n\}$.

Part (i). Consider a prefix-free oracle machine M, with coding constant c given by the Recursion Theorem, such that $M^n(\tau) = \mu$ iff $|\tau| = n - c$ and $V(\tau) = \mu$. Given σ, let τ be a minimal-length V-program for σ. On input τ and oracle $C(\sigma) + c$, this machine will output σ. Thus $K(\sigma \mid C(\sigma) + c) \leqslant C(\sigma) + c$, and hence $C(\sigma) \geqslant m_\sigma - O(1)$.

Now consider a plain machine N that, on input τ, searches for $\mu \preccurlyeq \tau$ such that $U^{|\tau|}(\mu)\downarrow$, and if such a μ is found, outputs $U^{|\tau|}(\mu)$. Note that, by the prefix-freeness of U, such a μ, if it exists, is unique. Suppose that $K(\sigma \mid n) \leqslant n$, and let μ be a minimal-length U-program for σ given n. Note that $|\mu| \leqslant n$. Let $\tau = \mu 0^{n-|\mu|}$. Then $N(\tau) = U^n(\mu) = \sigma$, so $C(\sigma) \leqslant |\tau| + O(1) = n + O(1)$. Thus $C(\sigma) \leqslant m_\sigma + O(1)$.

Part (ii). We have $K(\sigma \mid m_\sigma - 1) \geqslant m_\sigma - 1$, so $K(\sigma \mid m_\sigma) \geqslant K(\sigma \mid m_\sigma - 1) - O(1) \geqslant m_\sigma - O(1)$. But also $K(\sigma \mid m_\sigma) \leqslant m_\sigma$, so in fact $K(\sigma \mid m_\sigma) = m_\sigma \pm O(1)$. Now part (i) gives us

$$K(\sigma \mid C(\sigma)) = K(\sigma \mid m_\sigma) \pm O(1) = m_\sigma \pm O(1) = C(\sigma) \pm O(1).$$

$\square$

7.2 Solovay's Theorems relating K and C

In the remainder of this chapter, we will look at the beautiful unpublished material of Solovay relating K to C. The positive results involve simulations of Turing machines by prefix-free machines and vice-versa. The negative ones involve the construction of an infinite sequence of strings whose plain and prefix-free complexities behave very differently in the limit. These results also have a bearing on the relationship between C-randomness and strong K-randomness (as defined in Sections 6.1 and 6.11, respectively). All of the results and proofs in this and the following sections are due to Solovay [284]. In the final section of this chapter, we give applications of Solovay's results due to J. Miller to obtain a result of An. A. Muchnik and an improvement thereof.

As the following lemma shows, it is not hard to give upper bounds on $K(\sigma)$ in terms of $C(\sigma)$. Recall that, for a function f, we write $f^{(n)}$ for the result of composing f with itself n many times.

Lemma 7.2.1. *$K(\sigma) \leqslant C(\sigma) + C^{(2)}(\sigma) + \cdots + C^{(n)}(\sigma) + O(C^{(n+1)}(\sigma))$ for any n.*

Proof. This lemma follows from Lemma 7.2.2 below by an easy induction, but we give a direct proof here.

For a string $\sigma = a_0 a_1 \dots a_m$, let $\overline{\sigma} = a_0 a_0 a_1 a_1 \dots a_m a_m 01$. That is, $\overline{\sigma}$ is the result of doubling each bit of σ and adding 01 to the end. The key fact about this operation is that if $\sigma \neq \tau$ then $\overline{\sigma}$ and $\overline{\tau}$ are incompatible.

Thus there is a prefix-free machine M such that $M(\overline{\tau}) = V(\tau)$ for each τ. Taking τ to be a minimal-length V-program for σ, this fact shows that $K(\sigma) \leqslant |\overline{\tau}| + O(1) = O(C(\sigma))$, thus proving the lemma for $n = 0$.

But we can also build a prefix-free machine M so that for each τ we have $M(\overline{\tau'}\tau) = V(\tau)$, where τ' is a minimal-length V-program for $|\tau|$. On input μ, the machine M simply looks for a splitting $\mu = \mu_1 \mu_2$ such that

$\mu_1 = \overline{\nu}$ for some ν, then computes $V(\nu)$, and if that halts and equals $|\mu_2|$, computes $V(\mu_2)$ and outputs the result if any. Taking τ to be a minimal-length V-program for σ, this construction shows that $K(\sigma) \leqslant |\overline{\tau'}\tau| = C(\sigma) + O(C^{(2)}(\sigma))$, thus proving the lemma for $n = 1$.

It should now be clear how to build a prefix-free machine M so that for each τ we have $M(\overline{\tau''}\tau'\tau) = V(\tau)$, where τ'' is a minimal-length V-program for $|\tau'|$, which proves the lemma for $n = 2$, and how to iterate this process to prove the lemma in general. □

The more precise relationships between C and K, as given by Solovay [284], are

$$K(\sigma) = C(\sigma) + C^{(2)}(\sigma) \pm O(C^{(3)}(\sigma)). \tag{7.1}$$

and

$$C(\sigma) = K(\sigma) - K^{(2)}(\sigma) \pm O(K^{(3)}(\sigma)). \tag{7.2}$$

The goal of this section is to establish these equalities.

It is useful to recall what the O-notation means here. For example, (7.1) means that there is a d such that

$$C(\sigma) + C^{(2)}(\sigma) - dC^{(3)}(\sigma) \leqslant K(\sigma) \leqslant C(\sigma) + C^{(2)}(\sigma) + dC^{(3)}(\sigma)$$

for all σ. We have already seen that the second inequality holds for some d. The difficulty comes in proving the first inequality.

One might expect (7.1) and (7.2) to be part of an infinite sequence of approximations involving increasing numbers of iterations of C and K, respectively, as in Lemma 7.2.1. In the next section we will see that, remarkably, this is not the case.

As Solovay observed, (7.1) and (7.2) are in fact equivalent. Indeed, we will also prove that

$$K^{(2)}(\sigma) - C^{(2)}(\sigma) = O(K^{(3)}(\sigma)) \tag{7.3}$$

and

$$K^{(3)}(\sigma) \text{ is asymptotically equal to } C^{(3)}(\sigma). \tag{7.4}$$

Granted these two facts, (7.1) and (7.2) are clearly equivalent.

The proof of the above equations proceeds in two stages. First we prove that

$$K(\sigma) \leqslant C(\sigma) + K(C(\sigma)) + O(1), \tag{7.5}$$

which is not too difficult. Then we prove the more difficult inequality

$$C(\sigma) \leqslant K(\sigma) - K^{(2)}(\sigma) + K^{(3)}(\sigma) + O(1). \tag{7.6}$$

Note that (7.5) and (7.6) are close to (7.2), the problem being that (7.5) has the term KC rather than $K^{(2)}$. It turns out that we can use the estimate on $K - C$ to get one on $K^{(2)} - KC$ to establish (7.2). After doing so, it will remain to prove (7.3) and (7.4) to get (7.1).

We begin by establishing (7.5).

Lemma 7.2.2. $K(\sigma) \leqslant C(\sigma) + K(C(\sigma)) + O(1)$.

Proof. Define a prefix-free machine M as follows. On input μ, first attempt to simulate U by searching for $\mu_1 \preccurlyeq \mu$ such that $U(\mu_1)\downarrow$. If such a string is found then let μ_2 be such that $\mu = \mu_1\mu_2$ and check whether $|\mu_2| = U(\mu_1)$. If so, output $V(\mu_2)$ (if this value is defined).

Notice that M is prefix-free, since, firstly, U is prefix-free, and, secondly, if M halts on μ then $\mu = \mu_1\mu_2$ with $U(\mu_1)\downarrow$ and $|\mu| = |\mu_1| + |U(\mu_1)|$, which means that all extensions of μ_1 on which M halts have the same length, and hence are pairwise incompatible.

Given σ, let τ_2 be a minimal-length V-program for σ and let τ_1 be a minimal-length U-program for $C(\sigma) = |\tau_2|$. Then $M(\tau_1\tau_2) = V(\tau_2) = \sigma$, and hence $K(\sigma) \leqslant |\tau_2| + |\tau_1| + O(1) = C(\sigma) + K(C(\sigma)) + O(1)$. ☐

We now establish (7.6). The idea of the proof is the following. Fix a computable enumeration of $\mathrm{dom}(U)$ and let L_n be the list of strings $\tau \in \mathrm{dom}(U)$ such that $|\tau| = n$, ordered according to this enumeration. We will show that there is a c such that $|L_{K(x)}| \leqslant 2^{K(x)-K^{(2)}(x)+c}$. We will then argue as follows (in the proof of Lemma 7.2.5). Given σ, let τ be a minimal-length U-program for σ. Let μ_1 be a minimal-length U-program for $K^{(2)}(\sigma)$ and let μ_2 be a string of length $K(\sigma) - K^{(2)}(\sigma) + c$ encoding the position of τ on the list $L_{K(\sigma)}$. The strings μ_1 and μ_2 together allow us to compute both $K(\sigma) = U(\mu_1) + |\mu_2| - c$ and the position of τ on the list $L_{K(\sigma)}$, and hence allow us to compute σ. Since U is prefix-free, the string $\mu_1\mu_2$ is enough to allow us to compute σ, whence $C(\sigma) \leqslant |\mu_1\mu_2| + O(1) = K(\sigma) - K^{(2)}(\sigma) + K^{(3)}(\sigma) + O(1)$.

We now give the details of this argument.

Lemma 7.2.3. $|L_n| \leqslant 2^{n-K(n)+O(1)}$.

Proof. We wish to apply the KC Theorem (Theorem 6.6.1) to build a prefix-free machine M as follows. We enumerate the L_n simultaneously. Whenever we first see that $|L_n| \geqslant 2^k$, we enumerate a request $\langle n-k+1, n\rangle$. We claim that these requests form a KC set. Assume this claim for now, and let k_n be the largest k such that $|L_n| \geqslant 2^k$. Then $K(n) \leqslant n - k_n + O(1)$, so $2^{K(n)} \leqslant 2^{n-k_n+O(1)}$. Thus $|L_n| < 2^{k_n+1} \leqslant 2^{n-K(n)+O(1)}$.

So we are left with establishing the claim. Defining k_n as above, this task amounts to showing that

$$\sum_{n=0}^{\infty}\sum_{i=0}^{k_n} 2^{-(n-i+1)} \leqslant 1.$$

To show that the above inequality holds, first note that

$$\sum_{i=0}^{k_n} 2^{-(n-i+1)} = \sum_{j=n-k_n+1}^{n+1} 2^{-j} \leqslant \sum_{j=n-k_n+1}^{\infty} 2^{-j} = 2^{k_n-n}.$$

Since $2^{k_n} \leqslant |L_n|$, we now have

$$\sum_{i=0}^{k_n} 2^{-(n-i+1)} \leqslant 2^{-n}|L_n|,$$

and hence

$$\sum_{n=0}^{\infty}\sum_{i=0}^{k_n} 2^{-(n-i+1)} \leqslant \sum_{n=0}^{\infty} 2^{-n}|L_n| =$$

$$\sum_{n=0}^{\infty}\sum\{2^{-|x|} : x \in \mathrm{dom}(U) \wedge |x| = n\} = \sum\{2^{-|x|} : x \in \mathrm{dom}(U)\} \leqslant 1.$$

This fact establishes the claim and completes the proof. □

Corollary 7.2.4. $|L_{K(\sigma)}| \leqslant 2^{K(\sigma)-K^{(2)}(\sigma)+O(1)}$.

We now establish (7.6).

Lemma 7.2.5. $C(\sigma) \leqslant K(\sigma) - K^{(2)}(\sigma) + K^{(3)}(\sigma) + O(1)$.

Proof. Let c be such that $|L_{K(\sigma)}| \leqslant 2^{K(\sigma)-K^{(2)}(\sigma)+c}$ for all σ. We define a machine M as follows.

On input μ, first attempt to simulate U by searching for $\mu_1 \preccurlyeq \mu$ such that $U(\mu_1)\downarrow$. If such a string is found then let μ_2 be such that $\mu = \mu_1\mu_2$. Let $n = |\mu_2| + U(\mu_1) - c$ and interpret μ_2 as a number j in the interval $[1, 2^{|\mu_2|}]$ in the natural way. Enumerate L_n until its jth element appears, if ever. If such an element τ appears, output $U(\tau)$ (if this value is defined).

Given σ, let τ be a minimal-length U-program for σ and let ν_2 be a string of length $K(\sigma) - K^{(2)}(\sigma) + c$ encoding the position j of τ on the list $L_{K(\sigma)}$. Let ν_1 be a minimal-length U-program for $K^{(2)}(\sigma)$. If we run M on input $\nu_1\nu_2$ then M will set $n = |\nu_2| + U(\nu_1) - c = K(\sigma) - K^{(2)}(\sigma) + c + K^{(2)}(\sigma) - c = K(\sigma)$ and will proceed to search the list L_n for its jth element, namely τ. It will then output $U(\tau) = \sigma$.

Since $|\nu_1\nu_2| = K^{(3)}(\sigma) + K(\sigma) - K^{(2)}(\sigma) + c$, the result follows. □

We are now ready to show that (7.2) holds. Let m and n range over the integers. It is easy to check that

$$K(|m+n|) \leqslant K(|m|) + K(|n|) + O(1).$$

(Here $|\cdot|$ denotes absolute value.) Also,

$$K(|m|) \leqslant K(|m-n|) + K(|n|) + O(1)$$

and similarly with m and n interchanged, from which it follows that

$$|K(|m|) - K(|n|)| \leqslant K(|m-n|) + O(1).$$

Lemma 7.2.6. $C(\sigma) = K(\sigma) - K^{(2)}(\sigma) \pm O(K^{(3)}(\sigma))$.

Proof. Let $D(\sigma) = K(\sigma) - C(\sigma) - K^{(2)}(\sigma)$. We need to show that $|D(\sigma)| = O(K^{(3)}(\sigma))$.

By Lemma 7.2.5, $D(\sigma) \geqslant -K^{(3)}(\sigma) - O(1)$.

By Lemma 7.2.2, $D(\sigma) \leqslant K(C(\sigma)) - K^{(2)}(\sigma) + O(1)$. By the facts about the relationship between K-complexity and addition and subtraction of integers mentioned above,

$$|K(C(\sigma)) - K^{(2)}(\sigma)| \leqslant K(|C(\sigma) - K(\sigma)|) + O(1) = \\ K(|D(\sigma) + K^{(2)}(\sigma)|) + O(1) \leqslant K(|D(\sigma)|) + K^{(3)}(\sigma) + O(1).$$

Thus $D(\sigma) \leqslant K(|D(\sigma)|) + K^{(3)}(\sigma) + O(1)$.

Putting the two previous paragraphs together, we see that

$$|D(\sigma)| \leqslant K^{(3)}(\sigma) + K(|D(\sigma)|) + O(1).$$

For k ranging over the natural numbers, $K(k) = o(k)$, so

$$|D(\sigma)| - o(|D(\sigma)|) \leqslant K^{(3)}(\sigma),$$

whence $|D(\sigma)| = O(K^{(3)}(\sigma))$. □

We now turn to (7.3), from which (7.4) and (7.1) will follow easily.

Lemma 7.2.7. $|K^{(2)}(\sigma) - C^{(2)}(\sigma)| = O(K^{(3)}(\sigma))$.

Proof. By Lemma 7.2.6, $C(\sigma)$ and $K(\sigma)$ differ by $O(K^{(2)}(\sigma))$, and hence their K-complexities differ by $O(K^{(3)}(\sigma))$. That is,

$$K(C(\sigma)) = K^{(2)}(\sigma) \pm O(K^{(3)}(\sigma)). \tag{7.7}$$

Likewise, (7.7) shows that $K^{(2)}(C(\sigma))$ and $K^{(3)}(\sigma)$ differ by $o(K^{(3)}(\sigma))$, whence

$$K^{(2)}(C(\sigma)) = O(K^{(3)}(\sigma)). \tag{7.8}$$

Replacing σ by $C(\sigma)$ in (7.2), we have

$$C^{(2)}(\sigma) = K(C(\sigma)) - K^{(2)}(C(\sigma)) \pm O(K^{(3)}(C(\sigma))).$$

By (7.8), this equation gives us

$$C^{(2)}(\sigma) = K(C(\sigma)) \pm O(K^{(3)}(\sigma)). \tag{7.9}$$

Combining this equation with (7.7) establishes the lemma. □

Corollary 7.2.8. $K^{(3)}(\sigma)$ *is asymptotically equal to* $C^{(3)}(\sigma)$.

Proof. By Lemma 7.2.7, $K^{(2)}(\sigma)$ and $C^{(2)}(\sigma)$ differ by $O(K^{(3)}(\sigma))$, so $|K^{(3)}(\sigma) - K(C^{(2)}(\sigma))| = o(K^{(3)}(\sigma))$, and hence $K^{(3)}$ and $KC^{(2)}$ are asymptotically equal. But K and C are asymptotically equal, so $KC^{(2)}$ and $C^{(3)}$ are asymptotically equal. □

Corollary 7.2.9. $K(\sigma) = C(\sigma) + C^{(2)}(\sigma) \pm O(C^{(3)}(\sigma))$.

Proof. From Lemma 7.2.6 we have $K(\sigma) = C(\sigma) + K^{(2)}(\sigma) \pm O(K^{(3)}(\sigma))$. Using Corollary 7.2.7 we then have $K(\sigma) = C(\sigma) + C^{(2)}(\sigma) \pm O(K^{(3)}(\sigma))$. By Lemma 7.2.8, it follows that $K(\sigma) = C(\sigma) + C^{(2)}(\sigma) \pm O(C^{(3)}(\sigma))$. □

The following corollary of the previous results will be useful below.

Corollary 7.2.10. $K(\sigma) = C(\sigma) + K(C(\sigma)) \pm O(C^{(3)}(\sigma))$.

Proof. Combine Corollaries 7.2.8 and 7.2.9 with (7.9). □

The results above all have relativized forms. For instance, we have the following.

Corollary 7.2.11.

$$K(\sigma \mid \tau) = C(\sigma \mid \tau) + C^{(2)}(\sigma \mid \tau) \pm O(C^{(3)}(\sigma \mid \tau)),$$

$$C(\sigma \mid \tau) = K(\sigma \mid \tau) - K^{(2)}(\sigma \mid \tau) \pm O(K^{(3)}(\sigma \mid \tau)),$$

and

$$C(\sigma \mid \tau) \leqslant K(\sigma \mid \tau) + O(1) \leqslant C(\sigma \mid \tau) + K(C(\sigma \mid \tau)) + O(1).$$

Another point worth noticing is that the upper bound on the size of $L_{K(\sigma)}$ given in Corollary 7.2.4 is strict, as shown by the following result.

Lemma 7.2.12. $|L_{K(\sigma)}| \geqslant 2^{K(\sigma)-K^{(2)}(\sigma)+O(1)}$.

Proof. Suppose otherwise. Define the prefix-free machine M as follows. By the Recursion Theorem, we can assume we know the coding constant c of M. On input μ, search for a splitting $\mu = \mu_1\mu_2$ such that $U(\mu_1)\downarrow$ and $|\mu_2| = U(\mu_1) - |\mu_1| - (c+1)$. If such a splitting is found then let k be the element of $[1, 2^{|\mu_2|}]$ coded by μ_2, and search for the kth element enumerated into $L_{U(\mu_1)}$. If such an element τ is found, compute $U(\tau)$ and output the result, if any.

By assumption, there is a σ such that $|L_{K(\sigma)}| \leqslant 2^{K(\sigma)-K^{(2)}(\sigma)-(c+1)}$. Let τ be a minimal-length U-program for σ and let μ_1 be a minimal-length U-program for $K(\sigma) = |\tau|$. Let μ_2 be a string of length $K(\sigma) - K^{(2)}(\sigma) - (c+1)$ coding the position of τ in $L_{K(\sigma)}$. Then $M(\mu_1\mu_2)\downarrow = \sigma$. But $|\mu_1\mu_2| = K^{(2)}(\sigma) + K(\sigma) - K^{(2)}(\sigma) - (c+1) = K(\sigma) - (c+1)$, so $K_M(\sigma) = K(\sigma) - (c+1)$. But c is the coding constant of M, so $K(\sigma) \leqslant K(\sigma) - 1$, which is a contradiction. □

7.3 Solovay's results on strong K-randomness and C-randomness, and limitations on the results of the previous section

In this section, we will present Solovay's theorem that, roughly speaking, every strongly K-random finite string is C-random, but the converse is not true. One intuitive explanation for this difference between the two notions of randomness is the following. Suppose we know a string σ. If σ is C-random then also knowing $C(\sigma)$ gives us no additional information, but if σ is strongly K-random then also knowing $K(\sigma)$ gives us $K(|\sigma|)$. Thus there is more information in the prefix-free complexity of a strongly K-random string than in the plain complexity of a C-random string. Indeed, this extra amount of information is exploited in the proof of Theorem 7.3.7 below, which has as an immediate corollary that not every C-random string is strongly K-random.

The existence of C-random but not strongly K-random strings will also allow us to show that the main results of the previous section cannot be improved. For instance, in the previous section we established identity (7.1): $K(\sigma) = C(\sigma)+C^{(2)}(\sigma)\pm O(C^{(3)}(\sigma))$. At first glance, one would expect this identity to be part of an infinite sequence of identities with decreasing O-terms (as in Lemma 7.2.1). However, the methods of this section will show that the identity $K(\sigma) = C(\sigma)+C^{(2)}(\sigma)+C^{(3)}(\sigma)\pm O(C^{(4)}(\sigma))$ is *not* true, and hence (7.1) is as sharp as possible.

7.3.1 Positive results

The following proposition combines the information about C- and K-randomness from Corollaries 6.1.4 and 6.11.1.

Proposition 7.3.1. *There are constants c_C and c_K such that the following hold.*

(i) $C(\sigma) \leqslant |\sigma| + c_C$.

(ii) $|\{\sigma : |\sigma| = n \wedge C(\sigma) \leqslant n + c_C - j\}| = O(2^{n-j})$.

(iii) $K(\sigma) \leqslant |\sigma| + K(|\sigma|) + c_K$.

(iv) $|\{\sigma : |\sigma| = n \wedge K(\sigma) \leqslant n + K(n) + c_K - j\}| = O(2^{n-j})$.

Let

$$m_C(\sigma) = |\sigma| + c_C - C(\sigma)$$

and

$$m_K(\sigma) = |\sigma| + K(|\sigma|) + c_K - K(\sigma).$$

In light of Proposition 7.3.1, we see that these values measure how far the complexity of σ falls from its potential maximum. For either version of randomness, the random strings are those strings σ for which the corresponding $m(\sigma)$ is small.

For fixed constants c and d, let us say that σ is *essentially C-random* if $C(\sigma) \geqslant |\sigma|-c$ and *essentially strongly K-random* if $K(\sigma) \geqslant |\sigma|+K(|\sigma|)-d$. The results about these notions proved below will be independent of the choice of c and d.

As we will see, the following theorem implies that every essentially strongly K-random string is essentially C-random.

Theorem 7.3.2. $m_K(\sigma) \geqslant m_C(\sigma) - O(\log m_C(\sigma))$.

Proof. Since $C(\sigma) = |\sigma| - m_C(\sigma) + O(1)$,

$$K(C(\sigma)) = K(|\sigma| - m_C(\sigma) + O(1)) \leqslant K(|\sigma|) + K(m_C(\sigma) + O(1)) \leqslant K(|\sigma|) + O(\log m_C(\sigma)).$$

By Lemma 7.2.2, $K(\sigma) \leqslant C(\sigma) + K(C(\sigma)) + O(1)$. Consequently,

$$K(\sigma) \leqslant |\sigma| - m_C(\sigma) + K(|\sigma|) + O(\log m_C(\sigma)).$$

Rearranging this inequality, we get

$$|\sigma| + K(|\sigma|) - K(\sigma) \geqslant m_C(\sigma) - O(\log m_C(\sigma).$$

Since the left-hand side of this inequality is $m_K(\sigma)$ (up to the constant c_K), it follows that

$$m_K(\sigma) \geqslant m_C(\sigma) - O(\log m_C(\sigma)).$$

□

Corollary 7.3.3. *Every essentially strongly K-random string is essentially C-random.*

Proof. If σ is essentially strongly K-random then $m_K(\sigma) \leqslant c$ for some fixed c (independent of σ). By Theorem 7.3.2, $m_C(\sigma) - d \log m_C(\sigma) \leqslant c$ for some fixed d, which clearly implies that $m_C(\sigma) \leqslant c'$ for some fixed c'. □

7.3.2 Counterexamples

We now turn to counterexamples. We will present Solovay's construction of an infinite sequence of strings whose plain complexities behave "as differently as possible" from their prefix-free complexities, and examine the consequences of the existence of such sequences.

We begin with a few lemmas that will be useful below. The first says that if τ is essentially C-random given n, then there are many μ's of length n such that $\tau\mu$ is essentially C-random.

Lemma 7.3.4. *For each c there is a d such that, for all n and τ, if*

$$C(\tau \mid n) \geqslant |\tau| - c \tag{7.10}$$

then

$$|\{\mu : |\mu| = n \wedge C(\tau\mu) \leqslant |\tau| + n - d\}| \leqslant 2^{n-c}. \tag{7.11}$$

Proof. Fix c. We build a machine that, for each d, τ, and n not satisfying (7.11), provides a short C-program for τ given n. The length of this program will depend on d, and we will show that, for large enough d, this program is shorter than $|\tau| - c$, so that any τ and n not satisfying (7.11) for any d also fail to satisfy (7.10).

For each d, m, and n, let $S_{d,m,n}$ be a list of all σ such that

$$|\sigma| = m \quad \text{and} \quad |\{\mu : |\mu| = n \wedge C(\sigma\mu) \leqslant m + n - d\}| > 2^{n-c}.$$

Notice that the $S_{d,m,n}$ are uniformly c.e. Furthermore, there are at most $2^{m+n-d+1}$ many pairs $\langle \sigma, \mu \rangle$ such that $|\sigma| = m$ and $C(\sigma\mu) \leqslant m + n - d$, whence $|S_{d,m,n}| \leqslant 2^{m+c-d+1}$.

Define the (plain) machine M as follows. On input σ, n, search for $\sigma_1, \sigma_2, \sigma_3$ such that $\sigma_1, \sigma_2 \in \operatorname{dom}(U)$ and $\sigma = \sigma_1\sigma_2\sigma_3$. If such σ_i exist, then let $d = U(\sigma_1)$ and $m = U(\sigma_2) + |\sigma_3|$. Interpret σ_3 as a number j in the interval $[1, 2^{|\sigma_3|}]$ and output the jth element of $S_{d,m,n}$, if any.

Consider d, n, and τ not satisfying (7.11), and let $m = |\tau|$. Then $\tau \in S_{d,m,n}$. Let $\sigma_1, \sigma_2, \sigma_3$ be strings of minimal length such that $U(\sigma_1) = d$ and $U(\sigma_2) = d - c + 1$, and σ_3 has length $m + c - d + 1$ and codes the position of τ in $S_{d,m,n}$. Then $M(\sigma_1\sigma_2\sigma_3, n) = \tau$, so

$$\begin{aligned} C(\tau \mid n) \leqslant K(d) + K(d-c+1) + m + c - d + 1 + O(1) \leqslant \\ m - d + O(\log d) = |\tau| - d + O(\log d), \end{aligned}$$

where the O constant does not depend on τ or n. Choosing d large enough, we see that if n and τ do not satisfy (7.11) then $C(\tau \mid n) \leqslant |\tau| - c$, so n and τ do not satisfy (7.10). □

Combining Lemma 7.3.4 with part (iv) of Lemma 7.3.1, we have the following corollary, which says that if τ is essentially C-random given n, then, since there are many μ's of length n such that $\tau\mu$ is essentially C-random, there is such a μ that is essentially strongly K-random.

Corollary 7.3.5. *For each c there is a d such that, for all n and τ, if $C(\tau \mid n) \geqslant |\tau| - c$ then there is a μ with*

1. $|\mu| = n$,
2. $C(\tau\mu) \geqslant |\tau| + n - d$, *and*
3. $K(\mu) \geqslant n + K(n) - d$.

The next lemma says that, given any n, we can find an m such that the prefix-free complexity of the strongly K-random strings of length m is close to n.

Lemma 7.3.6. *There is a c such that for each n there is an m with $|m + K(m) - n| \leqslant c$.*

Proof. If d is large enough then $|K(m+d) - K(m)| \leqslant K(d) + O(1) < d$. Let $f(m) = m + K(m)$. Then $f(m+d) - f(m) = m + d + K(m+d) - m - K(m) = d + K(m+d) - K(m)$, so $0 < f(m+d) - f(m) < 2d$. Let $c = 2d$ and, given n, choose m such that $|f(m) - n|$ is minimal. Then $|f(m) - n| \leqslant c$, since otherwise one of $f(m+d)$ or $f(m-d)$ would be closer to n than $f(m)$. □

We are now ready to prove the main theorem of this section.

Theorem 7.3.7. *There is an infinite sequence of strings ν_i such that*

1. $\lim_i |\nu_i| = \infty$,
2. $C(\nu_i) = |\nu_i| \pm O(1)$, *and*
3. $\lim_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} = 1$.

Before proceeding with the proof, we note the following consequence. By item 2 in Theorem 7.3.7, each ν_i is essentially C-random. By item 3, ν_i is not essentially strongly K-random for large enough i. Hence the converse to Corollary 7.3.3 does not hold.

Corollary 7.3.8. *There are infinitely many strings that are essentially C-random but not essentially strongly K-random.*

Proof of Theorem 7.3.7. The idea of the proof is to find τ_i and n_i such that τ_i is hard to describe given n_i, but easy given $K(n_i)$. Then Corollary 7.3.5 implies that there is a μ_i of length n_i that is random enough so that from $K(\mu_i)$ we can compute $K(n_i)$ (so τ_i is easy to describe given $K(\mu_i)$), but at the same time $\tau_i\mu_i$ is essentially C-random. Letting $\nu_i = \tau_i\mu_i$, we have that $K(\nu_i)$ is not much larger than $K(\mu_i)$, but $C(\nu_i)$ is as large as possible. By carefully choosing τ_i and n_i so that ν_i is sufficiently longer than μ_i, we will be able to satisfy the requirements of the theorem.

We now proceed with the details of the proof. Recall that σ^* is the first U-program of length $K(\sigma)$ to converge with output σ. Recall also that by $\log n$ we mean the base 2 logarithm of n, rounded up to the nearest integer.

By Theorem 6.13.2, we can select $n_0, n_1, \ldots$ such that

1. $\log n_i = 2^{2^i}$ and
2. $K(n_i^* \mid n_i) \geqslant 2^i - O(i)$.

By Lemma 6.13.1,

$$K(n_i^* \mid n_i) = K(K(n_i) \mid n_i) \pm O(1) \leqslant K^{(2)}(n_i) + O(1) \leqslant \log 2^{2^i} + O(\log^{(2)} 2^{2^i}) = 2^i + O(i).$$

Combining this inequality with item 2 above, we have

$$K(n_i^* \mid n_i) = 2^i \pm O(i).$$

Thus $K^{(2)}(n_i^* \mid n_i) = O(i)$ (and hence, of course, $K^{(3)}(n_i^* \mid n_i) = o(i)$), so by Corollary 7.2.11,

$$C(n_i^* \mid n_i) = K(n_i^* \mid n_i) - K^{(2)}(n_i^* \mid n_i) \pm O(K^{(3)}(n_i^* \mid n_i)) = 2^i \pm O(i).$$

Let τ_i be the first to halt among the minimal-length V-programs for n_i^* given n_i. (That is, τ_i is $(n_i^*)^*_C$, where the second star is defined with respect to V^{n_i}.) Then

$$|\tau_i| = C(n_i^* \mid n_i) = 2^i \pm O(i).$$

Also, by the minimality of τ_i,

$$C(\tau_i \mid n_i) = |\tau_i| \pm O(1).$$

(That is, $|\tau_i| = C(n_i^* \mid n_i) \leqslant C(\tau_i \mid n_i) + O(1) \leqslant |\tau_i| + O(1)$. Thus it follows that τ_i is hard to describe given n_i, but as we will see below, τ_i is relatively easy to describe given n_i^*, and hence given n_i and $K(n_i)$.)

By Corollary 7.3.5, there is a μ_i such that

1. $|\mu_i| = n_i$,
2. $C(\tau_i\mu_i) = |\tau_i| + n_i \pm O(1)$, and
3. $K(\mu_i) = n_i + K(n_i) \pm O(1)$.

Let $\nu_i = \tau_i\mu_i$. Then $\lim_i |\nu_i| = \infty$, and each ν_i is C-random, since $C(\nu_i) = |\tau_i| + n_i \pm O(1) = |\nu_i| \pm O(1)$. So we are left with showing that $\lim_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} = 1$.

We begin by computing $K(|\nu_i|)$. First,

$$K(|\nu_i|) = K(|\tau_i| + n_i \pm O(1)) = K(n_i) \pm O(K(|\tau_i|)).$$

Now,

$$K(|\tau_i|) = K(2^i \pm O(i)) = K(2^i) \pm O(\log i) = O(\log i),$$

since $K(2^i) = K(i) \pm O(1) = O(\log i)$. Therefore,

$$K(|\nu_i|) = K(n_i) \pm O(\log i).$$

We now need an upper bound on $K(\nu_i)$. If we have a U-program σ for μ_i and a U-program σ' for τ_i given σ, then we can easily obtain a U-program for $\nu_i = \tau_i\mu_i$ of length roughly $|\sigma\sigma'|$. Thus,

$$K(\nu_i) \leqslant K(\mu_i) + K(\tau_i \mid \mu_i^*) + O(1) = n_i + K(n_i) + K(\tau_i \mid \mu_i^*) \pm O(1).$$

So we will obtain our bound on $K(\nu_i)$ by bounding $K(\tau_i \mid \mu_i^*)$.

Now, $n_i = |\mu_i|$, so $K(n_i \mid \mu_i^*) = O(1)$. Since $|\mu_i^*| = K(\mu_i) = n_i + K(n_i) \pm O(1)$, we have $K(K(n_i) \mid \mu_i^*) = O(1)$. By Lemma 6.13.1, $K(n_i^* \mid n_i, K(n_i)) = O(1)$. Thus,

$$K(n_i^* \mid \mu_i^*) \leqslant K(K(n_i) \mid \mu_i^*) + K(n_i \mid \mu_i^*) + K(n_i^* \mid n_i, K(n_i)) + O(1) = O(1).$$

So $K(\tau_i \mid \mu_i^*) \leqslant K(\tau_i \mid n_i^*) + K(n_i^* \mid \mu_i^*) + O(1) = K(\tau_i \mid n_i^*) + O(1)$.

Recall that τ_i is the first to halt among the minimal-length V-programs for n_i^* given n_i. By the relativized version of Lemma 6.2.1,

$$K(\tau_i \mid n_i^*) \leqslant K(|\tau_i| \mid n_i^*) + O(1) = O(\log |\tau_i|) = O(i),$$

so $K(\tau_i \mid \mu_i^*) = O(i)$.

Putting the last three paragraphs together, we have

$$K(\nu_i) \leqslant n_i + K(n_i) + O(i).$$

Together with the computation of $K(|\nu_i|)$ above, this bound implies that

$$\begin{aligned} m_K(\nu_i) = |\nu_i| + K(|\nu_i|) - K(\nu_i) \geqslant \\ 2^i + n_i + K(n_i) - O(i) - [n_i + K(n_i) + O(i)] = \\ 2^i - O(i) = \log^{(2)} |\nu_i| - O(\log^{(3)} |\nu_i|). \end{aligned}$$

(The last equality holds because $|\nu_i| = |\mu_i| + |\tau_i| = n_i + 2^i \pm O(i)$, and $\log n_i = 2^{2^i}$.) Thus

$$\liminf_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} \geqslant 1. \tag{7.12}$$

On the other hand, it is easy to get an upper bound on $m_K(\nu_i)$ using the results of the Section 7.2. Recall that $C(\nu_i) = |\nu_i| \pm O(1)$. Also, by Corollary 7.2.10, $K(\nu_i) = C(\nu_i) + K(C(\nu_i)) \pm O(C^{(3)}(\nu_i))$, so

$$\begin{aligned} m_K(\nu_i) = |\nu_i| + K(|\nu_i|) - K(\nu_i) = \\ C(\nu_i) + K(C(\nu_i)) \pm O(1) - [C(\nu_i) + K(C(\nu_i)) \pm O(C^{(3)}(\nu_i))] = \\ O(C^{(3)}(\nu_i)) = O(\log^{(2)} |\nu_i|). \end{aligned}$$

Thus $m_K(\nu_i) = O(\log^{(2)} |\nu_i|)$, but we are not quite done because of the multiplicative constant that might be hidden in the O-term. The following method of finishing the proof was suggested by Joe Miller.

Lemma 7.3.9. $K(|\nu_i|) \leqslant K^{(2)}(\nu_i) + K(m_K(\nu_i)) + O(1)$.

Proof. Assume that $c_i := K(|\nu_i|) - K^{(2)}(\nu_i) - K(m_K(\nu_i))$ is positive, since otherwise we are done. From minimal-length U-programs for $K(\nu_i)$, for $m_K(\nu_i)$, and for c_i, we can compute $K(|\nu_i|)$, and hence can compute $|\nu_i| = m_K(\nu_i) + K(\nu_i) - K(|\nu_i|)$. Thus,

$$\begin{aligned} K(|\nu_i|) &\leqslant K^{(2)}(\nu_i) + K(m_K(\nu_i)) + K(c_i) + O(1) \\ &= K(|\nu_i|) - c_i + K(c_i) + O(1). \end{aligned}$$

Thus $c_i \leqslant K(c_i) + O(1)$, whence $c_i = O(1)$. □

To finish the proof of the theorem given this lemma, we have the following inequality.

$$\begin{aligned} m_K(\nu_i) &= |\nu_i| + K(|\nu_1|) - K(\nu_i) \\ &= C(\nu_i) + K(|\nu_i|) - K(\nu_i) \pm O(1) \\ &\leqslant K(|\nu_i|) - K^{(2)}(\nu_i) + K^{(3)}(\nu_i) + O(1) && \text{by Lemma 7.2.5} \\ &\leqslant K(m_K(\nu_i)) + K^{(3)}(\nu_i) + O(1) && \text{by Lemma 7.3.9} \\ &\leqslant K(m_K(\nu_i)) + \log^{(2)} |\nu_i| + o(\log^{(2)} |\nu_i|). \end{aligned}$$

By the crude bound $m_K(\nu_i) = O(\log^{(2)} |\nu_i|)$ established above, $K(m_K(\nu_i)) = o(\log^{(2)} |\nu_i|)$, so

$$m_K(\nu_i) \leqslant \log^{(2)} |\nu_i| + o(\log^{(2)} |\nu_i|),$$

whence

$$\limsup_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} \leqslant 1.$$

Together with (7.12), this inequality implies that

$$\lim_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} = 1,$$

completing the proof of the theorem. □

The following theorem, whose proof is based on Theorem 7.3.7, has as a corollary that the identities (7.1) and (7.2) cannot be improved, and the inequality (7.5) cannot be reversed.

Theorem 7.3.10. *There are infinite sequences of strings ν_i, τ_i, and μ_i such that*

1. $\lim_i |\nu_i| = \infty$,
2. $C(\tau_i) = C(\nu_i) + O(1)$,
3. $\lim_i \frac{K(\tau_i) - K(\nu_i)}{\log^{(2)} |\nu_i|} = 1$,
4. $K(\mu_i) = K(\nu_i) + O(1)$, *and*
5. $\lim_i \frac{C(\nu_i) - C(\mu_i)}{\log^{(2)} |\nu_i|} = 1$.

Proof. Let ν_i be as in Theorem 7.3.7.

For each i, let τ_i be a strongly K-random string of length $|\nu_i|$. By Corollary 7.3.3, $C(\tau_i) = |\nu_i| \pm O(1) = C(\nu_i) \pm O(1)$. The choice of τ_i implies that $K(\tau_i) = |\nu_i| + K(|\nu_i|) \pm O(1)$, so

$$K(\tau_i) - K(\nu_i) = |\nu_i| + K(|\nu_i|) - [|\nu_i| + K(|\nu_i|) - m_K(\nu_i)] \pm O(1) = m_K(\nu_i) \pm O(1).$$

By Theorem 7.3.7,

$$\lim_i \frac{K(\tau_i) - K(\nu_i)}{\log^{(2)} |\nu_i|} = \lim_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} = 1.$$

By Lemma 7.3.6, there exist m_i such that $|m_i + K(m_i) - K(\nu_i)| = O(1)$. For each i, let μ_i be a strongly K-random string of length m_i. Then $K(\mu_i) = m_i + K(m_i) \pm O(1) = K(\nu_i) \pm O(1)$. Furthermore, by Lemma 7.3.3, $C(\mu_i) = m_i \pm O(1)$, and by Theorem 7.3.7, $C(\nu_i) = |\nu_i| \pm O(1)$. Thus to establish item 5, it is enough to show that $\lim_i \frac{|\nu_i| - m_i}{\log^{(2)} |\nu_i|} = 1$.

Let $r_i = |\nu_i| - m_i$. Then $K(m_i) = K(|\nu_i|) \pm O(\log r_i)$, so

$$r_i \pm O(\log(r_i)) = |\nu_i| + K(|\nu_i|) - (m_i + K(m_i)) = \\ |\nu_i| + K(|\nu_i|) - K(\nu_i) \pm O(1) = m_K(\nu_i) \pm O(1).$$

By Theorem 7.3.7,

$$\lim_i \frac{r_i}{\log^{(2)} |\nu_i|} = \lim_i \frac{m_K(\nu_i)}{\log^{(2)} |\nu_i|} = 1.$$

□

Corollary 7.3.11. (i) *The relation*

$$K(\sigma) = C(\sigma) + C^{(2)}(\sigma) + C^{(3)}(\sigma) \pm O(C^{(4)}(\sigma))$$

does not hold in general.

(ii) *The relation*

$$C(\sigma) = K(\sigma) - K^{(2)}(\sigma) + K^{(3)}(\sigma) \pm O(K^{(4)}(\sigma))$$

does not hold in general.

(iii) *The relation*

$$K(\sigma) = C(\sigma) + K(C(\sigma)) \pm O(1)$$

does not hold in general.

Proof. Let ν_i, τ_i, and μ_i be as in Theorem 7.3.10.

If the equation in part (i) of the corollary were true then, since $C(\nu_i) = C(\tau_i) \pm O(1)$, we would have

$$K(\nu_i) - K(\tau_i) = O(C^{(4)}(\nu_i)) = O(\log^{(3)} |\nu_i|).$$

But then

$$\lim_i \frac{K(\nu_i) - K(\tau_i)}{\log^{(2)} |\nu_i|} = 0,$$

contradicting Theorem 7.3.10.

The argument for part (ii) is the same, with μ_i in place of τ_i and K and C interchanged.

Similarly, if the equation in part (iii) were true then we would have $K(\nu_i) - K(\tau_i) = O(1)$, which would lead to the same contradiction as before. □

7.4 Muchnik's Theorem

In [221], Muchnik proved the following very interesting result relating prefix-free complexity and plain complexity. The proof we give, which is based on Solovay's results discussed above, is due to Joe Miller [208].

Theorem 7.4.1 (Muchnik [221]). *For each d there are strings σ and τ such that*

$$K(\sigma) > K(\tau) + d$$

and

$$C(\tau) > C(\sigma) + d.$$

Proof. Let $\{\nu_i\}_{i\in\mathbb{N}}$ be the sequence constructed in Theorem 7.3.7. For each i, take ρ_i to be a strongly K-random string of length $|\nu_i| - \frac{\log^{(2)}|\nu_i|}{2}$. (Here we slightly abuse notation by writing $\frac{\log^{(2)}|\nu_i|}{2}$ to mean $\lfloor\frac{\log^{(2)}|\nu_i|}{2}\rfloor$.) First note that

$$\begin{aligned}
K(\rho_i) - K(\nu_i) &= |\rho_i| + K(|\rho_i|) - K(\nu_i) \pm O(1)\\
&= |\nu_i| - \frac{\log^{(2)}|\nu_i|}{2} + K(|\nu_i| - \frac{\log^{(2)}|\nu_i|}{2} - K(\nu_i) \pm O(1)\\
&= |\nu_i| - \frac{\log^{(2)}|\nu_i|}{2} + K(|\nu_i|) - K(\nu_i) \pm O(\log^{(3)}|\nu_i|)\\
&= m_K(\nu_i) - \frac{\log^{(2)}|\nu_i|}{2} \pm O(\log^{(3)}|\nu_i|).
\end{aligned}$$

Therefore,

$$\lim_i \frac{K(\rho_i) - K(\nu_i)}{\log^{(2)}|\nu_i|} = \lim_i \frac{m_K(\nu_i) - \frac{\log^{(2)}|\nu_i|}{2}}{\log^{(2)}|\nu_i|} = 1 - \frac{1}{2} = \frac{1}{2}.$$

By Theorem 7.3.3, ρ_i is essentially C-random, so $C(\nu_i) - C(\rho_i) = |\nu_i| - |\rho_i| \pm O(1) = \frac{\log^{(2)}|\nu_i|}{2} \pm O(1)$, and hence

$$\lim_i \frac{C(\nu_i) - C(\rho_i)}{\log^{(2)}|\nu_i|} = \lim_i \frac{\frac{\log^{(2)}|\nu_i|}{2} \pm O(1)}{\log^{(2)}|\nu_i|} = \frac{1}{2}.$$

So for any d we can take i large enough so that $K(\rho_i) - K(\nu_i) \geqslant d$ and $C(\nu_i) - C(\rho_i) \geqslant d$. $\square$

Miller's methods allow an additional improvement to the previous result, namely that we can choose σ and τ to have the same length.

Theorem 7.4.2 (Miller [208]). *For each d there are strings σ and τ such that $|\sigma| = |\tau|$,*

$$K(\sigma) \geqslant K(\tau) + d,$$

and

$$C(\tau) \geqslant C(\sigma) + d.$$

Proof. We extend the proof above. For each i, define $\tau_i = \rho_i 0^{\frac{\log^{(2)}|\nu_i|}{2}}$. Then $|\tau_i| = |\nu_i|$. Furthermore, $K(\tau_i) = K(\rho_i) \pm O(\log^{(3)}|\nu_i|)$ and $C(\tau_i) = C(\rho_i) \pm O(\log^{(3)}|\nu_i|)$. Thus

$$\lim_i \frac{K(\tau_i) - K(\nu_i)}{\log^{(2)}|\nu_i|} = \frac{1}{2} = \lim_i \frac{C(\nu_i) - C(\tau_i)}{\log^{(2)}|\nu_i|}.$$

As above, for any d we can take i large enough so that $K(\tau_i) - K(\nu_i) \geqslant d$ and $C(\nu_i) - C(\tau_i) \geqslant d$. $\square$

Miller observed that the estimates used in both proofs can be improved; it is not hard to show that $K(|\nu_i| - \frac{\log^{(2)}|\nu_i|}{2}) = K(|\nu_i|) \pm O(1)$, and from this, that $K(\tau_i) = K(\rho_i) \pm O(1)$ and $C(\tau_i) = C(\rho_i) \pm O(1)$. The weaker estimates above were used because they are sufficient for our purposes and require no explanation.

8
Effective Reals

In this section we study the left computably enumerable reals, and discuss some other classes of effectively approximable reals. Left computably enumerable reals occupy a similar place in the study of relative randomness to that of computably enumerable sets in the study of relative computational complexity.

Recall that all the reals we consider are in $[0,1]$, and are identified with elements of 2^ω and with subsets of $\mathbb{N}$.

8.1 Representing reals

Let α be a real. By $L(\alpha)$ we mean the *left cut of* α, that is, $\{q \in \mathbb{Q} : q < \alpha\}$. It is well-known that such cuts can be used to define the reals from the rationals, as can Cauchy sequences. As we will, see both of these approaches can be effectivized.

Since we identify a real α with the set of natural numbers A such that nth bit of α is 1 iff $n \in A$, it is natural to define α to be a *computable real* if A is a computable set. Equivalently, α is computable if there is an algorithm that, on input n, returns the nth bit of α. (Notice that we could have used a base other than base 2 in this definition; it is easy to check that the computability of the binary expansion of α is equivalent to the computability of the n-ary expansion of α for any $n > 1$.) Another natural definition, based on the Dedekind cut approach, is to say that α is computable if $L(\alpha)$ is computable. Fortunately, these two definitions agree.

Proposition 8.1.1. *A real α is computable iff $L(\alpha)$ is computable.*

Proof. If $L(\alpha)$ is computable then we can compute $A \upharpoonright n$, since it is the lexicographically largest string $\sigma \in 2^n$ such that $0.\sigma \in L(\alpha)$.

If α is computable, then there is an algorithm for computing $L(\alpha)$: Let $q_n = 0.(A \upharpoonright n+1)$ (i.e., q_n is the rational whose binary expansion is given by the first $n+1$ bits of the characteristic function of A). Given $q \in \mathbb{Q}$, wait until either $q \leqslant q_n$ for some n, in which case $q \in L(\alpha)$, or $q - q_n \geqslant 2^{-n}$ for some n, in which case $q \notin L(\alpha)$, since $\alpha - q_n < 2^{-n}$. It is clear that one of the two cases must occur. □

Turning to the Cauchy sequence approach, a natural effectivization is to consider those reals α that are limits of *computable* sequences of rationals. As we will see, though, this class is much larger than that of the computable reals, even if we insist that the sequences of rationals be monotonic. The reason is that, to fully effectivize the notion of Cauchy sequence, we should require not only that the sequence be computable, but also that it have a computable *rate of convergence*. The following result is implicit in Turing's original paper [306].

Theorem 8.1.2 (Turing [306])**.** *A real is computable if and only if it is the limit of a computable sequence of rationals $q_0, q_1, \ldots$ for which there is a computable function f such that, for all n,*

$$|\alpha - q_{f(n)}| < 2^{-n}.$$

Proof. If $\alpha = 0.A$ for a computable set A then let $q_n = 0.(A \upharpoonright n+1)$. Then $q_0, q_1, \ldots \to \alpha$ and $|\alpha - q_n| < 2^{-n}$ for all n.

For the converse, suppose that α is the limit of a computable sequence of rationals as in the statement of the theorem. For ease of notation, let $r_n = q_{f(n)}$. If α is rational then it is computable, so assume α is irrational. For each k there must be an n such that the first $k+1$ bits of the binary expansions of $r_n - 2^{-n}$ and $r_n + 2^{-n}$ agree, since any convergent sequence of rationals not having this property must converge to a rational. Given k, search for an n with the above property and define $A(k)$ to be the kth bit of r_n. Since $\alpha \in (r_n - 2^{-n}, r_n + 2^{-n})$, the kth bit of α must be the same as that of the binary expansion of r_n, whence $\alpha = 0.A$. □

As the above proof shows, the function f in Theorem 8.1.2 can be required to be the identity function without altering the result. It is known from folklore that the computable reals form a *real closed field.* The methods used to prove this are along the lines used in the proof of Theorem 8.5.8, and we delay intrducing them until then.

Having defined computable reals, it is natural to look for a definition of computably enumerable reals. The following is the natural definition based on the Dedekind cut approach.

Definition 8.1.3. A real α is *left computably enumerable* (*left-c.e.*) if $L(\alpha)$ is computably enumerable.

Left-c.e. reals are often referred to simply as "c.e. reals", but we wish to avoid any confusion between c.e. *reals* and c.e. *sets*. Left-c.e. reals have also been called *recursively enumerable*, *left computable*, *left semicomputable*, and *lower semicomputable*.

The following result shows, among other equivalences, that the left-c.e. reals are exactly those that are approximable from below by a computable sequence of rationals.

Theorem 8.1.4 (Soare [276]; Calude, Hertling, Khoussainov, and Wang [36]). *The following are equivalent for a real α.*

1. *α is the limit of a computable increasing sequence of rationals.*
2. *α is the limit of a computably enumerable increasing sequence of rationals.*
3. *α is left computably enumerable.*
4. *There is a computably enumerable prefix-free set $A \subset 2^\omega$ such that $\alpha = \sum_{\sigma \in A} 2^{-|\sigma|}$.*
5. *There is a computable prefix-free set $A \subset 2^\omega$ such that $\alpha = \sum_{\sigma \in A} 2^{-|\sigma|}$.*
6. *There is a computable binary function f such that*
   (a) *for all k and s, if $f(k,s) = 1$ and $f(k,s+1) = 0$, then there is some $j < k$ such that $f(j,s) = 0$ and $f(j,s+1) = 1$,*
   (b) *$a_k := \lim_s f(k,s)$ exists for all k, and $\alpha = 0.a_0a_1 \ldots$.*

Remark The reader should be aware of the two orderings at work here. In (i) the rationals are coded and the sequence of codes computably enumerable. It is possible to have the sequence "increasing" as a sequence of rationals in the real ordering yet as codes they could be decreasing. For (v) we mean that there is a computable function $g : \omega \mapsto \mathbb{Q}$ with $\alpha = \lim_s g(s)$ and the range of g a computable set of (codes of) rationals.

Proof. None of the proofs are difficult and we leave most as exercises for the reader. The most interesting is that (ii) implies (vi). We need to replace $q_0, q_1, \cdots$ with a computable enumeration with the same limit. Let $<_\mathbb{R}$ denote the real ordering. We simply find a sequence of rationals with $q_n <_\mathbb{R} r_n <_\mathbb{R} q_{n+1}$ and such that the code of r_{n+1} exceeds that of r_n, which is possible by the density of the rationals. The sequence r_n so obtained has the same limit as the q_i and is increasing in Gödel number.

It is an interesting exercise to see how to use the Kraft-Chaitin inequality in, for example, the proof of the Calude, et al. result. For instance, if α is left-c.e. then it can be constructed as a computable sequence of dyadic rationals α_s so that $\alpha_{s+1} - \alpha_s$ is of the form $\sum_{j \in B_s} 2^{-j}$ and we can have that

$\sigma \in B_s$ implies that σ has length less than or equal to s. Thus $\alpha_{s+1} - \alpha_s = p(s)2^{-(s+1)}$. Hence by the Kraft-Chaitin inequality we can find a A_s of strings of length $s+1$ with $\alpha_{s+1} - \alpha_s = \sum_{\sigma \in A_s} 2^{-|\sigma|}$, and so that $A = \cup_s A_s$ is prefix free. (Specifically, we would enumerate $p(s)$ many requirements $\langle s+1, \lambda \rangle$.) $\square$

The above shows our first possible connection with randomness. We see that (iii) and (iv) say that left-c.e. reals correspond to the domains of prefix-free machines, via Kraft-Chaitin. In the same way, computably enumerable sets correspond to domains of Turing machines, and form the basis for much of classical computability theory.

Before we turn to randomness considerations, we use the above to point out that left-c.e. reals and c.e. sets are quite different. It is important that the reader realize that we are *not* defining a left-c.e. real to be $.A$ for some c.e. set A. Define a real α to be *strongly c.e.* if there is a c.e. set A such that $\alpha = .A$. It is easy to use the characterization above (specifically (iv)) to construct a left-c.e. real that is not strongly c.e. (a theorem of Soare [276]).

Specifically, we need to satisfy the requirement

$$R_j : \alpha \neq .W_e.$$

The idea is very simple. Devote positions $2e$ and $2e+1$ to R_e. We initially set $A(2e+1) = 1, A(2e) = 0$. If ever $2e+1 \in W_e$, make $A(2e+1) = 0$ and $A(2e) = 1$.

Notice that every strongly c.e. real is left-c.e. but that if A is c.e. and not computable, then $\alpha = .A$ is left-c.e. and cannot be computable.

8.2 Computably enumerable reals and randomness

The classic set in computability theory is the halting problem $K = \{e : \varphi_e(e)\downarrow\}$. This is the "natural" undecidable set. In the context of randomness, we saw in Theorem 9.2.5 that there is equally natural set, as we see below. Recall that for a fixed universal prefix-free machine M, we defined the *halting probability with respect to M* as

$$\Omega_M = \sum_{M(\sigma\downarrow)} 2^{-|\sigma|}.$$

In the last Chapter we observed that Ω was a Δ_2^0 real. But since $\Omega = \lim \Omega_s$ where $\Omega_s = \sum_{M(\sigma\downarrow)[s]} 2^{-|\sigma|}$, Ω is a left-c.e. real. This real can be thought of as an analog of the halting set for effective randomness. As we have already seen Ω is random. Furthermore as we see in the next Chapter, it is random, and amongst left-c.e. reals, in some sense the *only* random real.

8.3 Degree-theoretical aspects of representations

8.3.1 Degrees of representations

We say that a c.e. sequence of rationals $\{q_i : i \in \omega\}$ with monotonic limit α *represents* α.

Representations were first effectively analyzed by Calude, Coles, Hertling and Khoussainov [35]. We have seen that if a real is left-c.e. then it has a computable representation. If a real is computable then every representation must be computable (exercise). Suppose that a left-c.e. real is noncomputable. What else can be said about its representations? For instance, the natural degree of a left-c.e. real is the degree of its left cut: $\deg(L(\alpha))$. Does α always have a representation of degree $\deg(L(\alpha))$? of other degrees?

Theorem 8.3.1. *(i)* (**Calude, Coles, Hertling, and Khoussainov [35]**) *α has a representation of degree* $\deg(L(\alpha))$.

(ii) (**Soare [276]**) *If $B = \{q_i\}$ is a representation of α then $B \leqslant_T L(\alpha)$ and in fact $B \leqslant_{wtt} L(\alpha)$.*

(iii) (**Calude, et al. [35]**)*Every representation of α is half of a c.e. splitting of $L(\alpha)$.*

The theorem above extends earlier work of Soare who examined, in particular, the relationship between $L(a)$ and $\deg(B)$ for $a = \sum_{n \in B} 2^{-n}$. In [276], Soare observed that $L(a) \leqslant_T B$ and $B \leqslant_{tt} L(a)$. However, he also proved that there are strongly c.e. a, as above, with $L(a) \not\leqslant_{tt} B$.

Evidently (iii) implies (ii). Clearly, if A represents α then A must be an infinite c.e. subset of $L(\alpha)$. The thing to note is that $L(\alpha) - A$ is also c.e.. Given rational q, if q occurs in $L(\alpha)$, we need only wait till either q occurs in A or some rational bigger than q does.

Note that this means that if α is computable then every representation of α is computable. Also note that the proof actually gives that if A represents α, $A \leqslant_{wtt} L(\alpha)$. (It is interesting to note that strong reducibilities often play a large role in effective mathematics since reducibilities that occur naturally tend to be stronger than $\leqslant_T$. For instance, in a finitely presented group, the word problem tt-reduces to the conjugacy problem ([132]), algebraic closure is related to Q-reducibility ([28, 87, 196, 331]) and wtt-degrees characterize the degrees of bases of a c.e. vector space (Downey-Remmel [94]).)

We would like to prove that if A is half of a splitting of $L(\alpha)$ then A represents α. But it is not difficult to prove that this is not true. We know that if A represents α then there needs to be a computable function g with range A so that, as reals, $g(i) < g(i+1)$. It is easy to construct splittings of some α where no such g exists by a simple diagonalization argument. Calude et al. did find that the converse of (iii) did happen in some cases.

Theorem 8.3.2 (Calude et al. [35])**.** *Let A be a representation of α. For subsets B of A, the following are equivalent.*

- *B represents α.*
- *B is half of a splitting of A.*

The proof of this result is straightforward and is left to the reader. Calude et al. [35] also obtained a partial degree theoretical converse to (iii). Namely, they showed that *(i) α has a representation of degree* $\deg(L(\alpha))$*, and (ii) every representation can be extended to one of degree* $\deg(L(\alpha))$. In Downey [69], Downey improved the Calude et al. [35] result, and obtained a complete characterization of the representations of a real x in terms of the m-degrees of splittings of $L(x)$.

Theorem 8.3.3 (Downey [69])**.** *The following are equivalent*

- $\mathbf{b}$ *is the m-degree of a splitting of $L(x)$.*
- $\mathbf{b}$ *is the weak truth table degree of a representation of x.*

Proof. To prove Theorem 8.3.3, we need only show that if $L(x) = C \sqcup D$ is any c.e. splitting of $L(x)$ then there is a representation $\widehat{C} = \{c_i\}$ of x of wtt degree that of C. (Without loss of generality, we suppose that C is noncomputable.) We do this in stages. At each stage s, we assume that we have enumerated C_s and D_s so that $L(a)_s = C_s \sqcup D_s$, where $L(a)_s$ is the collection of rationals in $L(a)$ by stage s, including all those of Gödel number $\leqslant s$. Additionally we will have a parameter $m(s)$. At stage $s+1$ compute C_{s+1} and D_{s+1}. Find the least rational, $q \in C_{s+1}$, by Gödel number, if any, such that $q > m(s)$.

If no such q exists, set $m(s+1) = m(s)$, and do nothing else.

If one exists, put all rationals with Gödel number below $s+1$, in increasing real order, into $\widehat{C}_{s+1}$ and reset $m(s+1)$ to be the maximum rational (as a real) in $L(x)_{s+1}$.

To verify the construction, first note that $\widehat{C}$ is an increasing sequence of rationals. Its limit will be a provided that it is infinite, because of the use of $m(s)$.

First we claim that $m(s) \to \infty$. Suppose not, so that there is an s such that, for all $t \geqslant s$, $m(s) = m(t)$. Then we claim that C is computable, this being a contradiction. To decide if $z \in C$, go first to stage $s' = s + g(z)$, where $g(z)$ denotes the Gödel number of z. If $z \notin C_{s'}$, then either $z > m(s)$, or $z \in D_{s'}$. In either case, $z \notin C$. Hence $m(s) \to \infty$.

Note that $\widehat{C} \leqslant_m C$. Only numbers entering C enter $\widehat{C}$ and can do so only at the same stage. Given q go to a stage s bigger than the Gödel number of q. If q is below $m(s)$ then, as before, we can decide computably if $q \in C$. Else, note that $q \in C$ iff $q \in \widehat{C}$. The same argument shows that $C \leqslant_m \widehat{C}$. □

We remark that many of the theorems of Calude et al. [35] now come out as corollaries to the characterization above, and known results on splittings and *wtt* degrees. Notice that by Sacks splitting theorem every noncomputable left-c.e. real x has representations in infinitely many degrees. From known theorems we get the following.

Corollary 8.3.4. *There exist computably enumerable reals a_i such that the collection of T-degrees of representations $R(a_i)$ have the following properties.*

(i) $R(a_1)$ *consists of every c.e. (m-) degree*

(ii) $R(a_2)$ *forms an atomless boolean algebra, which is nowhere dense in the c.e. degrees.*

For the proofs see Downey and Stob [96].

We also remark that the above has a number of other consequences regarding known limits to splittings. For instance;

Corollary 8.3.5. *If a left-c.e. real a has representations in each T-degree below that of $L(a)$ then either $L(a)$ is Turing complete or low$_2$.*

This follows since Downey [68] demonstrated that a c.e. degree contains a set with splittings in each c.e. degree below it iff it was complete or low$_2$. It is not clear if every nonzero c.e. degree contains a left-c.e. real that cannot be represented in every c.e. degree below that of $L(\alpha)$.

8.4 Presentations of reals

The Calude et al. theorem gave many possible ways of representing reals, not just with Cauchy sequences. We explore the other methods with the following definition.

Definition 8.4.1 (Downey and LaForte [86])**.** Let $A \subset \{0,1\}^*$. We say that A is a *presentation* of a left-c.e. real x if A is a prefix free c.e. set with

$$x = \mu(A) = \sum_{n \in A} 2^{-|n|}.$$

8.4.1 Ideals and presentations

Previously we have seen that every real x has a representation of degree $L(x)$. However, presentations can behave quite differently.

Theorem 8.4.2 (Downey and LaForte [86])**.** *There is a left-c.e. real α that is not computable, but such that if A presents α then A is computable.*

Theorem 8.4.2 follows from the Downey-Terwijn theorem below (Theorem 8.4.9). We remark that Downey and LaForte demonstrated that degrees containing such "only computably presentable" reals can be high, but if a degree is promptly simple then every left-c.e. real of that degree must have a noncomputable c.e. presentation. Using a $\mathbf{0}'''$ argument, Wu [320] has constructed a c.e. degree $\mathbf{a} \neq \mathbf{0}$ such that if α is any left-c.e. noncomputable real of degree below $\mathbf{a}$, then α has a noncomputable presentation.

As with many structures of computable algebra and the like, the classification of the degrees realized as presentations seems to depend on a stronger reducibility than Turing reducibility. In this case, the relevant reducibility seems to be weak truth table reducibility. The following result is easy to prove.

Theorem 8.4.3 (Downey and LaForte [86])**.** *Let α be a left-c.e. real and let A be such that $\alpha = .A$. Let B be a presentation of α. Then $B \leqslant_{wtt} A$ with use function the identity.*

The proof is left as an exercise. What is interesting is that there is a sort of converse to this result.

Theorem 8.4.4 (Downey and LaForte [86])**.** *If A is a presentation of a left-c.e. real α and $C \leqslant_{wtt} A$ is c.e., then there is a presentation B of α with $B \equiv_{wtt} C$.*

Proof. Suppose $\Gamma(X)$ is a computable functional with a computable use function γ such that $\Gamma(A) = C$. We can assume γ is monotonically increasing. Let $\langle n, m\rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be a computable one-to-one function such that for all n, m, $\max\{n, m\} < \langle n, m\rangle$. (Adding 1 to the usual pairing function gives such a function.) Notice that, since A presents α, using the Chaitin-Kraft theorem we can enumerate strings of any length we wish into $B[s]$ at as long as we ensure

$$\sum_{\sigma \in B[s]} 2^{-|\sigma|} \leqslant \sum_{\sigma \in A[s]} 2^{-|\sigma|}.$$

We fix enumerations of Γ, C and A so that at each stage s, exactly one element enters C and for every $x < s$, $\Gamma_s(A_s; x) = C_s(x)$. We may assume A is infinite, since there is nothing to prove if A is computable. We construct B in stages, using the function $\langle n, m\rangle$ as follows.

At stage 0, let $B[0] = \emptyset$.

At stage $s + 1$, we first find the unique number n_s entering C and all strings σ that enter A at stage $s + 1$. For each $|\sigma| < \gamma(n_s)$, we enumerate $2^{\langle|\sigma|, n_s\rangle - |\sigma|}$ strings of length $\langle|\sigma|, n_s\rangle$ into $B[s + 1]$. For each $|\sigma| \geqslant \gamma(n_s)$, we enumerate $2^{\langle|\sigma|, |\sigma|+s\rangle - |\sigma|}$ strings of length $\langle|\sigma|, |\sigma| + s\rangle$ into $B[s]$.

This ends the construction of B.

Notice that all of the actions taken at stage $s + 1$ serve to ensure that

$$\sum_{\sigma \in B[s+1]} 2^{-|\sigma|} = \sum_{\sigma \in A[s+1]} 2^{-|\sigma|};$$

hence, we always have enough strings available to keep B prefix-free.

Suppose $n \in \mathbb{N}$. Let $s(n)$ be least so that $B[s(n)]$ agrees with B on all strings less than or equal to length $\langle \gamma(n), n \rangle$. Now, suppose there exists $t > s(n)$ such that $n \in C[t]-C[t-1]$. In this case, because for every s and $x < s$, $C(x)[s] = \Gamma(A; x)[s]$, there must be some σ with $|\sigma| < \gamma(n)$ which enters A at t. By construction, then, since $n = n_t$, we have $2^{\langle |\sigma|, n_t \rangle - |\sigma|} > 1$ strings of length $\langle |\sigma|, n_t \rangle$ entering B at stage $t > s(n)$, which is a contradiction. Hence we can compute $C(n)$ from $B(n)$ with a use bounded by the number of strings of length less than or equal to $\langle \gamma(n), n \rangle$, which is a computable function. This gives $C \leqslant_{wtt} B$.

Next consider any binary string τ. Using the computability of $\langle i, n \rangle$ and the fact that $\max\{i, n\} < \langle i, n \rangle$ we can ask whether there exist i and n such that $|\tau| = \langle i, n \rangle$. If not, then $\tau \notin B$. In this case, let $t(n) = 0$. Otherwise, suppose $|\tau| = \langle i, n \rangle$. If $i \geqslant \gamma(n)$, then τ can only enter B at stage s if $s = n - i$. If, on the other hand, $i < \gamma(n)$. Then if τ enters B at stage $s + 1$, this can only be because $|\tau| = \langle |\sigma|, n_s \rangle$ for some σ entering A at s, and we enumerate $2^{\langle |\sigma|, n_s \rangle - |\sigma|}$ strings of length $\langle |\sigma|, n_s \rangle$ into $B[s + 1]$. In either case, if we let $t(n)$ be the least number greater than $n - i$ so that $C[t(n)] \upharpoonright_{n+1} = C \upharpoonright_{n+1}$, we have $B(\tau) = B(\tau)[t(n)]$. Since n is computable from $|\tau|$, $B \leqslant_{wtt} C$, as required. $\square$

Note that one corollary is that a strongly c.e. real $\alpha = .A$ with the degree of A *wtt*-topped[1], has the property that it has presentations in every T degree below that of A.

We remark that the Theorem above does not hold for *tt*-reducibility.

Proposition 8.4.5. *There exist a left-c.e. real α and a presentation B of α such that $B \nleqslant_{tt} \alpha$.*

Proof. We construct a left-c.e. real α and a prefix-free c.e. domain B presenting α such that for every e, if φ_e is a tt-reduction, then there is a string σ such that

$R_e: \quad \sigma \in B \iff \alpha \not\models \varphi_e(\sigma).$

Let $D = \{0^n 1 : n \in \omega\}$, so that D is a recursive prefix-free domain.

Stage 0. Let $\alpha[0] = 0$, $B[0] = \emptyset$, $\sigma_{e,0} = 0^e 1$.

Stage $s > 0$. Look at the smallest e for which R_e has not yet been satisfied and for which $\varphi_e(\sigma_e) \downarrow [s]$. If $\alpha[s] + 2^{-|\sigma_e|} \models \varphi_e(\sigma_e)$ then instead of putting σ_e into B we put the extensions $\sigma_e 0$ and $\sigma_e 1$ into B. If $\alpha[s] + 2^{-|\sigma_e|} \not\models \varphi_e(\sigma_e)$ then put σ_e into B. In both cases add $2^{-|\sigma_e|}$ to α, and initialize all R_i with $i > e$ by redefining σ_i to be fresh strings from D.

Clearly B is c.e., and B is prefix-free because D is. Furthermore, $\mu(B) = \alpha$ because every time we add measure to B we add the same amount numerically to α. Finally, if φ_e is a tt-reduction, then $\sigma_e \in B \Leftrightarrow \alpha \not\models \varphi_e(\sigma_e)$

[1] That is, for all c.e. $B \leqslant_T A$, $B \leqslant_{wtt} A$

because σ_e is kept out of B precisely when $\alpha \models \varphi_e(\sigma_e)$. Because after every diagonalization the lower priority σ_i, $i > e$, are picked fresh, they do not interfere with the action taken for R_e. Hence the construction is finite injury. $\square$

Note the following.

Lemma 8.4.6. *Suppose that A and B present α. Then there is a presentation of α of wtt degree $A \oplus B$.*

Proof. Note that $C = \{0\sigma : \sigma \in A\} \cup \{1\sigma : \sigma \in B\}$ is prefix free, as both A and B are, and presents α. $\square$

Corollary 8.4.7 (Downey and LaForte [86]). *The wtt-degrees of c.e. sets presenting α forms a Σ^0_3 ideal.*

Proof. Note that

$$\mathcal{I}(\alpha) = \{W_e : \exists A \text{ presentation of } \alpha.\ W_e \equiv_{wtt} A\}$$

forms an ideal by Theorem 8.4.4 and Lemma 8.4.6. Let us determine the complexity of $\mathcal{I}(\alpha)$. The statement "$\mu(W_e) = \alpha$" is Π^0_2 ("for all diameters ε there is a stage s such that $\mu(W_e)[s]$ and $\alpha[s]$ are closer than ε"). Saying that W_e is prefix-free is Π^0_1 ($\forall \sigma, \tau \in W_e.\ \sigma \not\sqsubset \tau$). For a given c.e. set A the set $\{W_e : W_e \equiv_{wtt} A\}$ is Σ^0_3 (see Odifreddi [234, p627]; roughly, we have to say "there exists a wtt-reduction such that $\forall x \forall y \leqslant x \exists s > x$ such that at stage s the reduction gives the right answers on y"). All in all, $W_e \in \mathcal{I}(\alpha)$ if and only if there exists d such that a Σ^0_3 statement holds true of W_d. So we see that $\mathcal{I}(\alpha)$ is a Σ^0_3-ideal. $\square$

To see that this is optimal, note that for α computable we have by Theorem 8.4.4 that $\mathcal{I}(\alpha) = \{W_e : W_e \text{ computable}\}$, and this set is Σ^0_3-complete. $\mathcal{I}(\alpha)$ is not always Σ^0_3-complete: For $\alpha = \chi_K$ we already saw that $\mathcal{I}(\alpha) = \{W_e : e \in \omega\}$ is trivial (as an index set). The following result, in the spirit of Rice's Theorem, saying that this is in fact the *only* case where $\mathcal{I}(\alpha)$ is not Σ^0_3-complete.

Theorem 8.4.8 (Downey and Terwijn [97]). *$\mathcal{I}(\alpha)$ is either $\{W_e : e \in \omega\}$ or Σ^0_3-complete.*

Proof. Let $\alpha = \chi_A$ be a left-c.e. real. It is easy to see that $\mathcal{I}(\alpha) = \omega$ iff A is wtt-complete. Suppose that A is not wtt-complete. We prove that $\mathcal{I}(\alpha)$ is Σ^0_3-complete. This can be proved using the methods of Rogers and Kallibekov, see Odifreddi [234, p625-627]. We sketch the proof and leave the details to the reader. Let $\mathrm{Inf} = \{e : W_e \text{ is infinite}\}$. We use that the weak jump $\{x : W_x \cap \mathrm{Inf} \neq \emptyset\}$ of Inf is Σ^0_3-complete. It suffices to build sets B_x uniformly in x such that

$$\begin{aligned} W_x \cap \mathrm{Inf} \neq \emptyset &\implies B_x \text{ computable} \\ W_x \cap \mathrm{Inf} = \emptyset &\implies B_x \not\leqslant_{wtt} A. \end{aligned}$$

In the first case clearly the wtt-degree of B_x contains a presentation of α, while in the second case it follows from Theorem 8.4.4 that this is not the case.

We have requirements

P_e : $e \in W_x \wedge W_e$ infinite $\Longrightarrow (\forall i \geqslant e)\ [\omega^{[i]} \subseteq B_x]$

that try to make B_x computable, and

R_e : (Γ_e, γ_e) is total wtt-reduction $\Longrightarrow \exists z[B_x(z) \neq \Gamma_e^A(z)]$.

for making $B_x \not\leqslant_{wtt} A$, and give them the priority ordering $P_0 < R_0 < P_1 < R_1 < \ldots$

R_e is handled by Sacks's coding strategy (see Theorem 5.12.1): We maintain a length of agreement function $l(e,s)$ monitoring agreement between B_x and Γ_e^A. We code $K \restriction l(e,s)$ into $\omega^{[2e+1]}$. Then, provided that the higher priority requirements are finitary, R_e is also finitary (and hence satisfied), since otherwise the whole of K would be coded into B_x and still we would have $B_x \leqslant_{wtt} A$, contradicting the incompleteness of A.

P_e is handled directly by filling the rows above $\omega^{[2e]}$ up to $\max W_e[s]$ at every stage s whenever e is found to be in W_x.

If W_x contains no code of an infinite c.e. set then all P_e are finitary, hence every R_e succeeds and $B_x \not\leqslant_{wtt} A$. If on the other hand $e \in W_x$ is a minimal code of an infinite c.e. set then $(\forall i \geqslant e)[\omega^{[i]} \subseteq B_x]$. Since all higher priority requirements are finitary, $B_x \cap \omega^{[i]}$ is finite for every $i < e$. Hence B_x is computable. □

We have seen that $\mathcal{I}(\alpha)$ is a Σ_3^0-ideal. Theorem 8.4.9 says that conversely every Σ_3^0-ideal in the c.e. wtt-degrees is of the form $\mathcal{I}(\alpha)$ for some left-c.e. real α.

Theorem 8.4.9 (Downey and Terwijn [97]). *Suppose that $\mathcal{I}$ is any Σ_3^0 ideal in the computably enumerable wtt degree s. Then there is a left-c.e. (noncomputable) real α whose degrees of presentations are exactly the members of $\mathcal{I}$.*

Note that $\mathcal{I}$ consisting of a single element $\{\mathbf{0}\}$ is an example of an ideal, and hence we get the existence of a noncomputable left-c.e. real with only computable presentations (Theorem 8.4.2) as a corollary.

The proof will make use of the following lemma, which states that every Σ_3^0 collection of c.e. sets that contains all the finite sets is identical to a uniformly c.e. collection of c.e. sets, and implies that every Σ_3^0-ideal is generated by a uniform collection of c.e. sets.

Lemma 8.4.10 (Yates [324]). *Let $I \in \Sigma_3^0$ and let $\mathcal{C} = \{W_i : i \in I\}$ be a collection of c.e. sets containing all the finite sets. Then there is a uniformly c.e. collection $\{V_e : e \in \omega\}$ such that $\mathcal{C} = \{V_e : e \in \omega\}$.*

Proof. Let R be a computable predicate such that $i \in I \Leftrightarrow \exists e \forall n \exists m R(i,e,n,m)$. For each e, construct a c.e. set V_e as follows. For each successive n, look for an m such that $R(i,e,n,m)$, and if such an m is found then copy W_i

by setting $V_{e,n} = W_{i,n}$. If $\forall n \exists m R(i,e,n,m)$ then V_e equals W_i. Otherwise, for some n, there is no m such that $R(i,e,n,m)$, so V_e is finite. In either case, $V_e \in \mathcal{C}$, so $\{V_e : e \in \omega\} \subseteq \mathcal{C}$. Conversely, for each $i \in I$ there is an e such that $\forall n \exists m R(i,e,n,m)$, whence $V_e = W_i$, so $\mathcal{C} \subseteq \{V_e : e \in \omega\}$. □

Proof. (of Theorem 8.4.9) **Outline of the proof.** By Lemma 8.4.10 we may suppose that the Σ_3^0-ideal is given to us by a uniform collection of c.e. sets $U_0, U_1, U_2, \ldots$ We want to construct α such that for all e:

$$C_e : \text{ code } U_e \text{ into } \mathcal{I}(\alpha).$$

We satisfy C_e by constructing $A_e \equiv_{wtt} U_e$ with $\alpha = \mu(A_e)$, We remark that if we desired to make α noncomputable, the some of the C_e could be $\alpha \neq .\overline{W_e}$. These would be also positive requirements, and would be satisfied by waiting for a n to occur in $W_{e,s}$ and putting 2^{-n} into α. This is also a coding action and is met for the same reasons as the coding requirements are. This is, as we see below, we will need to code some quantity 2^{-n} into A for the sake of coding. This will be drip fed into A because of the action of the N_e, as we see below. There will be no difference in this action which ever type of $C - e$ we chose to use. We will not mention this further.

$$N_e : W_e \text{ presents } \alpha \Longrightarrow W_e \leqslant_{wtt} \bigoplus_{i \leqslant e} A_i.$$

First we describe the strategies for meeting these requirements in isolation, and then we describe how we will combine the strategies (using a tree of strategies).

We will try to satisfy $U_e \leqslant_{wtt} A_e$ by permitting: Along with A_e we define a use function ψ_e such that whenever a number x enters U_e we put (or at least try to put) a string $\psi_e(x)$ into A_e.

We will try to ensure $A_e \leqslant_{wtt} U_e$ by allowing a small string to enter A_e only for the sake of coding U_e (or making α noncomputable). So, assuming that $\psi_e(x) \geqslant x$ we will have $A_e \leqslant_{wtt} U_e$ with the identity as use function.

Along with the construction we will define α by enumerating rational values in it (see Theorem 8.1.4). $\alpha[s]$ will be the approximation of α determined by the numbers put into it by stage s. The second part of C_e will be satisfied by ensuring that there are infinitely many stages s with $(\alpha = \mu(A_e))[s]$, so that indeed all the A_e will present the same α.

For N_e, if $\alpha[s]$ and $\mu(W_e)[s]$ grow close we will try to make W_e computable by restraining $\alpha[s]$. We will monitor how close the two get by defining a monotone unbounded sequence of numbers $m(e)[s]$, and every time we see that $|\alpha - \mu(W_e)| < 2^{-m(e)}[s]$ we will try to keep $\alpha[s]$ from changing on short strings, thus allowing only minor changes. Were we to completely succeed in this, then W_e would be computable as follows: When asked if $\gamma \in W_e$, run the construction until $|\alpha - \mu(W_e)| < 2^{-m(e)}[s]$, with $m(e)[s] >> |\gamma|$. Then $\gamma \in W_e$ if and only if $\gamma \in W_e[s]$.

A coding strategy C_e can easily live with the action of a higher priority coding strategy C_i simply by picking different coding locations. We describe how the other strategies can be combined.

First we look at how N_e can deal with the outcome of a higher priority C_i. As described above, when at stage s it holds that $|\alpha - \mu(W_e)| < 2^{-m(e)}[s]$, N_e tries to restrain $\alpha[s]$ by trying to keep it from changing more than $2^{-m(e)}$. (It will allow minor changes in α to give lower priority requirements a chance of succeeding.) However, the coding action of C_i may spoil this. Suppose that, despite the injuries of C_i, at the end of the construction W_e presents α. Although we cannot argue anymore that W_e is computable, we can still argue that it is computable in A_i, which is good enough for us. To compute whether $\gamma \in W_e$, A_i compute s so large that $\mu(A_i)$ changes no more than $2^{-|\gamma|+1}$ after s by the coding of U_i. Then, using that the construction is computable, compute a stage s such that $|\alpha - \mu(W_e)| < 2^{-m(e)}[s]$, where $2^{-m(e)}[s] < 2^{-|\gamma|+2}$. Then $\alpha[s]$ is not changed more than $2^{-m(e)}[s]$ by N_e, and $\alpha[s]$ is not changed more than $2^{-|\gamma|+1}$ because of the coding of C_i, so $\mu(W_e)[s]$ cannot change more than $2 \cdot 2^{-m(e)}[s] + 2^{-|\gamma|+1} < 2^{-|\gamma|}$. So $\gamma \in W_e$ if and only if $\gamma \in W_e[s]$. Hence W_e is computable in A_i.

Second we look at how C_i can deal with the outcome of a higher priority N_e. There are two relevant outcomes of N_e: The infinitary outcome is when at infinitely many stages (which we will call e-expansionary stages) $\mu(W_e)$ grows closer to α. The finitary outcome is when from a certain stage onwards, $\mu(W_e)[s]$ is bounded away from $\alpha[s]$. Suppose that x enters U_e at stage s. Then A_e wants to code this event by enumerating a string δ. Suppose further that $|\alpha - \mu(W_e)|[s] < 2^{-m(e)}[s] < 2^{-|\delta|}$. Then A_e is not allowed to enumerate a string as short as δ, since this would cause $\alpha[s]$ to change $2^{-|\delta|}$, which is more than N_e allows. To get around this we use the trick of Downey and LaForte [86, Theorem 8]. Namely, in the above situation we let A_e announce that it wishes to enumerate δ, without actually doing so. Furthermore, we make α slightly bigger, so little that the computability of W_e as described above is not affected, namely that if $\mu(W_e)$ is to stay close to α then W_e cannot enumerate a short string. Then there are two possibilities for W_e: Either it does not respond, remaining forever more than $2^{-m(e)}[s]$ apart from α, in which case it does not present α and the outcome of N_e will be finitary. Or it responds by growing closer than $2^{-m(e)}[s]$ to α again, in which case we repeat the procedure. If W_e keeps responding to the small changes we make in α, by repeating enough times we will be able to create enough space between A_e and α for δ to enter A_e. Note that it is important that we do not allow N_e to let its value $m(e)$ grow during this procedure. We will refer to this strategy as the "drip feed strategy", since we think of C_i succeeding by feeding α changes small enough to be allowed by N_e, and doing this often enough to be able to finally make its move. (In Downey [71] the analogy is drawn with a stock market trader that wants to dump a large number of shares without disrupting the market. Every time the trader sells a small amount of shares he waits until the price recovers before doing so again.)

The strategy for C_e becomes a little more complicated when it has to deal with the outcome of more than one N-strategy. Suppose that C_e is below N_i, which in its turn is below N_j. Suppose that we try to put δ into A_e using the drip feed strategy described above. Then C_e will try to change α by an amount of 2^{-n} in $2^{-|\delta|+n}$ steps, where $n = m(i)$, the maximum change in α that N_i allows for. Now while waiting for N_i to respond to the first change, N_j may let its value $m(j)$ grow, since it does not know (or care) whether N_i is going to respond. When N_i finally does respond, $m(j)$ may have become so big that N_j does not allow a change of 2^{-n} in α, thus frustrating the drip feed strategy of C_e. The solution is to let N_i in turn use a drip feed strategy to let N_j allow for a change of 2^{-n}. If both N_i and N_j are infinitary, in the end all the changes in α requested by C_e will be allowed for. After every successfully completed drip feeding strategy, the N-strategies are allowed to let their m-value grow. This works in general for any finite number of N-strategies above C_i, by recursively nesting the drip feed strategies.

The tree of strategies. In general, of course, a requirement has to deal with the outcome of *all* the higher priority requirements, not just one. We handle the combinatorics of this using the usual infinite injury framework of putting all the strategies on a tree. We use $2^{<\omega}$ as a tree of strategies, assigning both C_e and N_e to every string of length e. Define

$$g(e) = \begin{cases} 0 & \text{if } W_e \text{ presents } \alpha \\ 1 & \text{otherwise.} \end{cases}$$

The path defined by g is called the *true path* of the construction. At every stage s we will have a finite approximation g_s of g of length at most s such that $g = \liminf_{s\to\infty} g_s$. For any string σ, a stage s is a *σ-stage* if $s = 0$ or $s > 0$ and $\sigma \sqsubseteq g_s$. A σ-strategy is *initialized* if all its parameters are set to being undefined. For any two strings σ and τ, $\sigma <_L \tau$ if and only if there is a string ρ such that $\rho 0 \sqsubseteq \sigma$ and $\rho 1 \sqsubseteq \tau$. A σ-strategy has *higher priority* than a τ-strategy if $\sigma \sqsubset \tau$ or $\sigma <_L \tau$. For the two σ-strategies, C_σ has priority over N_σ.

To coordinate the drip feed strategies, we equip every node σ with a *counter* $c(\sigma)$ and let σ only act at stages where $c(\sigma) = 0$. The counter $c(\sigma)$ will indicate how many steps a drip feed strategy needs to be successfully completed. So, at the start of a drip feed strategy initiated by some low priority coding requirement that wants to put a short string δ into α, the counter $c(\sigma)$ is set to $2^{-|\delta|+n}$ for some number n determined by the restraint of the first infinitary N-requirement above it. Every time a package of size 2^{-n} passes $N_{|\sigma|}$ the counter $c(\sigma)$ is decreased by 1. Every time the counter reaches 0 we allow σ to act.

In order to enable the infinitary coding actions of low priority requirements to interleave with the coding of high priority requirements, we equip the strategies with *lists* Λ for bookkeeping which strings wish to enter α using a drip feed strategy. After a drip feed strategy is successfully completed, the top element of Λ is removed. Every string on Λ has to wait until it is on the top of the list before it can start a drip feed strategy. (Such lists were not needed in the proof of Theorem 8.4.2 since there the positive requirements were finitary.)

Construction. Every σ will build its own copy A_σ and try to satisfy $C_{|\sigma|}$ by building a wtt-reduction ψ_σ from $U_{|\sigma|}$ to A_σ. The construction will feature several auxiliary parameters and functions: For every σ we have lists $\Lambda(C_\sigma)$ and $\Lambda(N_\sigma)$, a counter $c(\sigma)$, functions $l(\sigma)$ and $m(\sigma)$ monitoring the length of agreement, and a restraint function $r(\sigma)$. The construction proceeds in stages.

Stage $s = 0$. Set $\alpha[0] = 0$, $g[0] = \lambda$. For all σ, let $A_\sigma[0] = \emptyset$, and initialize all σ-strategies, i.e. set all parameters to be $\uparrow$ (undefined).

Stage $s > 0$. The finite approximation $g[s]$ (of length at most s) to the constructions true path will be defined by the σ that are active at stage s. These σ are defined by recursion, in increasing order. Given an active σ, the next (if any) active node is determined by N_σ.

Action for the positive requirement C_σ. Let $e = |\sigma|$. First we pick suitable coding locations for U_e in A_σ: For every $x \leqslant s$, if $\psi_\sigma(x)[s-1]\downarrow$ then let $\psi_\sigma(x)[s] = \psi_\sigma(x)[s-1]$. If $\psi_\sigma(x)[s-1]\uparrow$ then pick for $\psi_\sigma(x)[s]$ a fresh string of length bigger than x and not extending any string previously enumerated into A_σ. For every x that enters U_e at s do the following: Let ρ be the longest initial segment of σ such that $\rho 0 \sqsubseteq \sigma$ and $r(\rho) > \psi_\sigma(x)[s]$. (This means that ρ is the largest initial such that $\psi_\sigma(x)$ has to use the drip feed strategy to get into A_σ.) Add $\psi_\sigma(x)[s]$ to the list $\Lambda(C_\sigma)[s]$ and add $|\psi_\sigma(x)[s]|$ to $\Lambda(N_\rho)[s]$. ($\Lambda(C_\sigma)$ is the list C_σ uses to keep track of the strings it wants to put into A_δ using a drip feed strategy. $\Lambda(N_\sigma)$ is a list of numbers that N_σ uses to bookkeep for which (sizes of) strings a request has been made by a lower priority requirement to enter α.) If ρ does not exist, put $\psi_\sigma(x)[s]$ into A_σ straightaway.

Let δ be the string on top of the list $\Lambda(C_\sigma)[s]$. Let ρ be the longest initial segment of σ such that $\rho 0 \sqsubseteq \sigma$ and $r(\rho)[s] > |\delta|$. (The existence of such ρ is guaranteed by the fact that δ is on $\Lambda(C_\sigma)$.) See if $|\delta|$ is on the list $\Lambda(N_\rho)[s]$. If so, do nothing. If not, this means that δ has been successfully processed by N_ρ, and hence, by recursion, by all relevant N-strategies above σ. So in this case we put δ into A_σ and remove it from $\Lambda(C_\sigma)$, and we initialize all *negative* strategies $\tau \geqslant \sigma$ (meaning that the restraint value of these strategies becomes undefined).

We make $\alpha = \mu(A_\sigma)[s]$ by putting, if necessary, either fresh strings (not extending previous ones) into A_σ, or numbers into α.

Action for the negative requirement N_σ. Let $e = |\sigma|$. Define the following length of agreement functions,

$$l(\sigma)[s] = \begin{cases} s & \text{if } \mu(W_e) = \alpha[s] \\ \min\{n : |\alpha - \mu(W_e)|[s] > 2^{-n}\} - 1 & \text{otherwise.} \end{cases}$$

$$m(\sigma)[s] = \max\{l(\sigma)[t] : t < s\}.$$

so that we always have $|\alpha - \mu(W_e)| \leqslant 2^{-l(\sigma)}[s]$. A σ-stage s is *σ-expansionary* if $l(\sigma) > m(\sigma)[s]$.

If s is not σ-expansionary we let $\sigma 1$ act at s, and we initialize all τ-strategies with $\sigma <_L \tau$.

If s is σ-expansionary we initialize all τ-strategies, $\tau \geqslant \sigma 1$. Whether $\sigma 0$ is allowed to act depends on the value of σ's counter $c(\sigma)$ (see two cases below).

If $\Lambda(N_\sigma)[s]$ is empty we set $r(\sigma) = l(\sigma)[s]$ and let $\sigma 0$ act. Otherwise, let n be the number on top of the list $\Lambda(N_\sigma)[s]$. If $c(\sigma) \uparrow [s-1]$ then set $c(\sigma) = 2^{-n+r(\sigma)}[s]$ and let $\sigma 0$ act. If $c(\sigma) \downarrow [s-1]$ there are two cases:

I. $c(\sigma)[s-1] = 0$. This means that σ's drip feed strategy for n has been successfully completed. Remove n from $\Lambda(N_\sigma)[s]$, set $r(\sigma) = l(\sigma)[s]$, and let $\sigma 0$ act.

II. $c(\sigma)[s-1] > 0$. This means that at a previous stage t this counter was set to some number $2^{-n+r(\sigma)}[t]$, and $r(\sigma)$ has not changed since. Let ρ be maximal with $\rho 0 \sqsubseteq \sigma$ and $r(\rho) > r(\sigma)[s]$. Since ρ acts its counter $c(\rho)$ must be 0 at s. See if $r(\sigma)[s]$ is on the list $\Lambda(N_\rho)[s]$. If so, do nothing. (In this case $r(\sigma)[s]$ is still waiting for its turn to start a drip feed strategy.) If not, $r(\sigma)[s]$ was removed from $\Lambda(N_\rho)$ when its counter $c(\rho)$ became 0 at some stage $\leqslant s$. So we let $c(\sigma)[s] = c(\sigma)[s-1]-1$, and we put the next copy of $r(\sigma)[s]$ on the list $\Lambda(N_\rho)[s]$. If there is no such ρ we add $2^{-r(\sigma)}[s]$ to α.

This completes the construction.

Verification. Because the construction is recursive, α is a left-c.e. real, and every A_σ is a c.e. prefix-free domain.

We prove by induction along the true path g that for σ on g we have $A_\sigma \equiv_{wtt} U_e$, $\mu(A_\sigma) = \alpha$, and if W_e presents α then $W_e \leqslant_{wtt} \bigoplus_{\tau \sqsubseteq \sigma} A_\tau$, where $e = |\sigma|$.

First note that always $A_\sigma \leqslant_{wtt} U_e$: A string δ can only enter A_σ after $\psi_e(x)$ has been picked when $|\delta| > |\psi_e(x)|$ or $\delta = \psi_e(y)$ for some y. Since $|\psi_e(x)| > x$, if U_e does not change below $|\delta|$ then also A_σ doesn't. So $A_\sigma \leqslant_{wtt} U_e$ with use the identity function.

Claim For $\rho 0 \sqsubset g$, every number put on $\Lambda(N_\rho)$ is eventually removed from $\Lambda(N_\rho)$ again.
Namely, since there are infinitely many stages at which ρ acts, and ρ can

only act when $c(\rho) = 0$ or when $\Lambda(N_\rho)$ is empty, infinitely often $\Lambda(N_\rho)$ is empty or its top element is removed. Since any element on the list has only finitely many predecessors on the list, every element is eventually removed.

Suppose that C_σ is never initialized after stage s, i.e. $g[s]$ is never to the left of σ. Note that $\tau \sqsubset \sigma$ can then only initialize σ's negative strategy. We prove that $U_e \leqslant_m A_\sigma$ via ψ_σ. Now suppose x enters U_e at σ-stage $t > s$. Let $\rho \sqsubseteq \sigma$ be maximal with $\rho 0 \sqsubseteq \sigma$ and $r(\rho) > r(\sigma)[t]$. If no such ρ exists $\psi_\sigma(x)$ enters A_σ immediately. Otherwise, $\psi_\sigma(x)$ is added to the list $\Lambda(C_\sigma)$ and $|\psi_\sigma(x)|$ is added to $\Lambda(N_\rho)$. By the above claim it is eventually removed from $\Lambda(N_\rho)$. But then it is also eventually removed from $\Lambda(C_\sigma)$, and put into A_σ at the same stage. Since no other strategy can put $\psi_\sigma(x)$ into A_σ we have $x \in U_e \Leftrightarrow \psi_\sigma(x) \in A_\sigma$.

Now we also have that $\mu(A_\sigma) = \alpha$ because $\alpha = \mu(A_\sigma)[s]$ at the end of C_σ's action at every σ-stage s.

We verify that N_σ is satisfied. Suppose that $W_e \subseteq 2^{<\omega}$ is prefix-free and $\mu(W_e) = \alpha$. Suppose that s is such that the σ-strategies are never initialized after stage s. Let $\gamma \in 2^{<\omega}$. We compute whether $\gamma \in W_e$ as follows. Determine a σ-stage $t \geqslant s, |\gamma| + 1$ such that $A_\tau[s] \upharpoonright |\gamma| + 1 = A_\tau \upharpoonright |\gamma| + 1$ for all $\tau \sqsubseteq \sigma$, and such that $l(\sigma)[t] > |\gamma| + 1$ and $r(\sigma) = l(\sigma)[t]$. We can compute the latter because the construction is recursive and $\lim_s l(\sigma)[s] = \infty$ by the assumption that $\mu(W_e) = \alpha$. Furthermore, we may choose t such that $r(\sigma) = l(\sigma)[t]$, because $\sigma 0$ will act infinitely often. By the definition of l we have $|\alpha - \mu(W_e)| \leqslant 2^{-l(\sigma)}[t]$. Which things can change $\alpha[t]$, and what is the effect on $\mu(W_e)$?

- The coding strategies C_τ with $\tau \sqsubseteq \sigma$ are done below $|\gamma| + 1$ by choice of t.
- Since all $\tau \geqslant \sigma 1$ are initialized at every σ-expansionary stage, the coding strategies C_τ with $\sigma <_L \tau$ cannot change α more than $2^{-t} \leqslant 2^{-|\gamma|-1}$ in total.
- The coding strategies C_τ with $\tau \sqsupseteq \sigma 0$ may wish to change $\alpha[t]$ below $r(\sigma)$. Note that $r(\sigma) = l(\sigma)[t]$. But they have to use the drip feed strategy in order to do this, meaning that they cannot disrupt the inequality $|\alpha - \mu(W_e)| < 2^{-r(\sigma)}[t]$ by more than $2^{-r(\sigma)}[t]$ ever, after which they have to wait until $\mu(W_e)$ grows closer to α than $2^{-r(\sigma)}[t]$ again. This means that $\mu(W_e)$ cannot change on any string of length smaller than $r(\sigma)[t]$, and in particular not on γ.

Summing up, $\mu(W_e)[t]$ cannot change more than

$$2^{-|\gamma|-1} + 2^{-r(\sigma)}[t] < 2 \cdot 2^{-|\gamma|-1} = 2^{-|\gamma|}.$$

So $\gamma \in W_e \Leftrightarrow \gamma \in W_e[t]$.

This completes the verification, and the proof of Theorem 8.4.9. $\square$

Theorem 8.4.11 (Downey and LaForte [86]). *There is a noncomputable left-c.e. real α such that if V presents α then V is computable.*

Proof. The computable sets form a Σ_3^0 ideal. □

We also remark that there would be no problems in making α high via standard highness requirements. The "drip feed" strategy makes it hard to get elements into α. Can we say anything about degrees that realize ideals? What about the Downey-LaForte reals with only computable presentations? The answer is "sometimes." This is the topic of the next subsection.

8.4.2 Promptly simple presentations of reals

We recall from Definition 5.19.4, that a coinfinite computably enumerable set D is called promptly simple if there is a computable function p such that for every infinite c.e. set W, there exists a stage $s > 0$ and number x such that $x \in (W[s] - W[s-1]) \cap D[p(s)]$. It is clear that no generality is lost by also requiring that for every s, $p(s) > s$: the intuitive meaning is therefore that p enables D to eventually guess correctly about some immediate change in W. This notion was introduced by Maass in [195], and general technical methods for working with promptly simple sets were developed in Ambos-Spies, Jockusch, Shore, and Soare [6], as we have seen in, for example, Theorem 5.19.6. We now show that if a left-c.e. real a has promptly simple degree, it has a noncomputable presentation.

Constructions involving promptly simple sets are simplified by the use of the following technical result due to Ambos-Spies, Jockusch, Shore, and Soare, [6]:

Theorem 8.4.12 (Slowdown Lemma). *Let $U_e[s]$ be a computable sequence of finite sets such that for all e $U_e[s] \subseteq U_e[s+1]$ and $U_e = \bigcup_{s=0}^{\infty} U_e[s]$. Then there exists a computable function g such that for all e,*

(i.) $W_{g(e)} = U_e$, *and*

(ii.) *if* $x \in U_e[s] - U_e[s-1]$, *then* $x \notin W_{g(e)}[s]$.

The condition (*ii.*) on $W_{g(e)}$ means that every element enumerated into U_e appears strictly later in $W_{g(e)}$.

Theorem 8.4.13 (Downey and LaForte [86]). *Suppose α has promptly simple degree. Then there is a presentation, A, of α that is noncomputable.*

Proof. Let $\alpha \equiv_T D$, where D is promptly simple, and suppose α is given to us with an almost-c.e. approximation — that is, there is a computable function $\alpha(i, s)$ such that

(i.) for all i, $\alpha(i) = \lim_{s\to\infty} \alpha(i, s)$

(ii.) for all i and s, if $\alpha(i,s) = 1$ and $\alpha(i,s+1) = 0$, then there exists some $j < i$ such that $\alpha(j,s) = 0$ and $\alpha(j,s+1) = 1$.

In particular, let $D = \Gamma(\alpha)$. We must give an algorithm to enumerate a prefix-free set of strings A so that $\sum_{x \in A} 2^{-|x|} = \alpha$, and for every program index e,

$$'P_e : W_e \text{ infinite} \Longrightarrow \overline{A} \neq W_e.$$

This ensures A is noncomputable, since $\overline{A}$ is not c.e.

The following result shows that we can always assume we have some strings of the proper length available to add to our set. It is worth noting that this result does not enable us to choose the particular strings we intend to add, however: this is one feature that makes working with presentations of a left-c.e. real different from working with the Dedekind-cut type representations.

We will build A in stages, enumerating at most one binary string at each stage, in such a way that there are infinitely many stages s such that $(\sum_{x \in A} 2^{-|x|} = \alpha)[s]$.

In order to diagonalize against all c.e. sets, we need to add strings to A at various stages which have relatively short lengths. The strategy for satisfying $'P_e$ will involve a finite sequence of attempts to enumerate a short string from W_e into A, at least one of which will work if W_e is infinite. Very briefly the idea is the following: Because α must compute the promptly simple set D, we can use the function p giving the prompt simplicity of D to search for a stage at which α must change on some relatively small value, enabling us to enumerate a string from W_e into A. The key fact is that the search can be computably bounded by p and is guaranteed to eventually succeed by the prompt simplicity which p witnesses for D.

Construction:

At stage 0, $A[0] = \emptyset$.

We now specify the actions at stage $s > 0$: Notice that by Kraft-Chaitin, since $\alpha \leqslant 1$, we can enumerate a string of any length we wish into A at a given stage and simultaneously preserve prefix-freeness as long as we guarantee that $(\sum_{x \in A} 2^{-|x|} \leqslant \alpha)[s]$ At any stage s where we are not in the process of making some attempt on a requirement $'P_e$, we simply add enough strings of the proper length into A to ensure $(\sum_{x \in A} 2^{-|x|} = \alpha)[s]$.

First, suppose there is some least $e < s$ for which $'P_e$ has an active witness $x = x(e,i)[s]$ such that $D(x)[s] = \Gamma(\alpha; x)[s] = 0$ with use y_0 and $\overline{A}[s]$ and $W_e[s]$ are equal on all strings of length less than or equal to y_0. If α changes below this use, we would like to enumerate a diagonalizing witness of length y_0 into A. However, a change in α below 2^{-y_0} may be caused by the enumeration of a string of length longer than y_0 simply because the y_0th position of α is followed by a sequence of 1s. We therefore wait for a stage $t_0 > s$ and y of length greater than y_0 to appear such that $\overline{A}[s]$ and $W_e[s]$ are equal on all strings of length less than or equal to y and there is some j such that $y_0 < j < y$ such that $\alpha(j, t_0) = 0$. Note that such a stage must appear since α is noncomputable, hence irrational. Then we enumerate x into the auxiliary set U_e. Using the appropriate function g giving an index for a $W_{g(e)}$ which slows down the enumeration into U_e, we let $t > t_0$ be least such that $x \in W_{g(e)}[t]$. We now freeze all action for our construction until stage $p(t)$ is reached, and then check to see whether $x \in D[p(t)]$. If so, then $\alpha[p(t)] - \alpha[t] > 2^{-y}$, since α changed below 2^{-y_0}, so that we may add a new string of length less than or equal to y into A at stage $p(t)$ and satisfy $'P_e$ permanently. Otherwise we release A, enumerate sufficient strings of the proper length to restore $(\sum\limits_{x \in A} 2^{-|x|} = \alpha)[p(t)]$, and declare attempt i on $'P_e$ to have ended in failure.

Finally, let $j < s$ be least such that $'P_j$ is unsatisfied and there is no active witness defined for requirement $'P_j$ at s. Choose a new witness $x(j, k)[s]$, where k is the least number for which all previous attempts at satisfying P_j have ended in failure.

This ends the construction.

Verification

We just need to show that every requirement $'P_e$ is eventually satisfied and we only freeze A finitely often for the associated strategy.

Since D is coinfinite, if W_e is infinite and all our attempts at satisfying $'P_e$ were to end in failure, then the set $U_e = W_{g(e)}$ would be infinite. But then, since D is promptly simple, this means there would be some x in $(W_{g(e)}[t] - W_{g(e)}[t-1]) \cap D[p(t)]$. By definition of $W_{g(e)} = U_e$, this means D must have changed value on x between the stage $t_0 < t$ at which $\overline{A}[t_0]$ appeared to equal $W_e[t_0]$ and $p(t)$. But then some element from $W_e[t_0]$ is enumerated into A at stage $p(t)$ by construction, satisfying the requirement. Since each $'P_e$ can be satisfied after a finite number of attempts, it is a straightforward induction to show that each associated strategy only freezes A finitely often. Thus all requirements can be satisfied, and

$$\lim_{s \to \infty} \sum_{x \in A[s]} 2^{-|x|} = \alpha,$$

as required.

□

As we see in Chapter 15, in Theorem 15.10.1, there are other conditions on natural classes of reals that guarantee noncomputable presentations. There Stephan and Wu show that “K-triviality” implies having a noncomputable presentations for noncomputable reals.

8.5 Other classes of reals.

8.5.1 D.c.e. reals

One potentially important and basically unexplored class of reals is that of the d.c.e. reals. These are defined as follows.

Definition 8.5.1. We say that a real α is a *d.c.e. real* iff there exist left-c.e. reals β and γ such that $\alpha = \beta - \gamma$.

The d.c.e. reals are interesting since they are the field generated by the left-c.e. reals.

Lemma 8.5.2 (Ambos-Spies, Weihrauch, Zheng [14])**.** *A real x is d.c.e. if and only if there exist a constant M and a computable sequence of rationals $q_0, q_1, \dots$ with limit x such that*

$$\sum_{j=0}^{\infty} |q_{j+1} - q_j| < M.$$

Proof. If such a sequence exists, then define the left-c.e. reals y and z by adding $q_{j+1} - q_j$ to y whenever $q_{j+1} - q_j > 0$, and adding $q_j - q_{j+1}$ to z whenever $q_{j+1} - q_j < 0$. The fact that $\sum_{j=0}^{\infty} |q_{j+1} - q_j|$ is bounded ensures that y and z are finite, and clearly $x = y - z$.

For the converse, let $x = y - z$ for left-c.e. reals y and z. Let $y_0, y_1, \dots$ and $z_0, z_1, \dots$ be increasing sequences of rationals converging to y and z, respectively, and let $x_n = y_n - z_n$. Then

$$\sum_{n=0}^{\infty} |x_{n+1} - x_n| = \sum_{n=0}^{\infty} |(y_{n+1} - z_{n+1}) - (y_n - z_n)| \leqslant \sum_{n=0}^{\infty} (y_{n+1} - y_n) + \sum_{n=0}^{\infty} (z_{n+1} - z_n) = y - y_0 + z - z_0.$$

□

Question 8.5.3. *Characterize the computable g such that α is d.c.e. iff $\alpha(i) = \lim_s g(i,s)$ in the sense of (iv) of the Calude et al. theorem above.*

Theorem 8.5.4 (Ambos-Spies, Weihrauch, and Zheng [14])**.** *The d.c.e. reals form a field.*

Proof. Rearranging terms shows closure under addition, subtraction, and multiplication (e.g., $(x-y)(p-q) = (xp+yq)-(yp+xq)$). We show closure under division. Let x and y be d.c.e. reals. By Lemma 8.5.2, there are a constant M and sequences of rationals $x_0, x_1, \dots$ and $y_0, y_1, \dots$ converging to x and y, respectively, such that $\sum_{n=0}^{\infty} |x_{n+1} - x_n| < M$ and $\sum_{n=0}^{\infty} |y_{n+1} - y_n| < M$. We may assume we have chosen M large enough so that for all n, each of $|x_n|$, $|y_n|$, and $\frac{1}{|y_n|}$ is less than M. Now the sequence of rationals

$\frac{x_0}{y_0}, \frac{x_1}{y_1}, \ldots$ converges to $\frac{x}{y}$, and

$$\sum_{n=0}^{\infty}\left|\frac{x_{n+1}}{y_{n+1}} - \frac{x_n}{y_n}\right| = \sum_{n=0}^{\infty}\left|\frac{y_n x_{n+1} - y_{n+1}x_n}{y_n y_{n+1}}\right| =$$

$$\sum_{n=0}^{\infty}\left|\frac{y_n x_{n+1} - y_n x_n + y_n x_n - y_{n+1}x_n}{y_n y_{n+1}}\right| =$$

$$\sum_{n=0}^{\infty}\left|\frac{x_{n+1} - x_n}{y_{n+1}}\right| + \sum_{n=0}^{\infty}\left|\frac{x_n}{y_n y_{n+1}}(y_n - y_{n+1})\right| <$$

$$M\sum_{n=0}^{\infty}|x_{n+1} - x_n| + M^3\sum_{n=0}^{\infty}|y_n - y_{n+1}| < M^2 + M^4.$$

Thus, by Lemma 8.5.2, $\frac{x}{y}$ is a d.c.e. real. □

One could also ask what about other classes of reals. For instance, we have seen that if we have a monotone increasing computable sequence of reals we get a left-c.e. real. What happens if we weaken the condition that the sequence be monotone as reals? As we have seen if the jumps are bounded, then we get d.c.e. reals. In general, we get the following.

Theorem 8.5.5 (Demuth [62], Ho [127])**.** *A real α is of the form $.A$ for A a Δ^0_2 set iff α is the limit of a computable set of rationals.*

Proof. This uses another padding+density argument, as in the Calude, et al. result, and is left as an exercise. □

We remark that it is not difficult to show that there are d.c.e. reals that are not left-c.e. Here is a proof. Notice that if D is a d.c.e. set (that is $D = A - B$ for c.e. sets A and B) then $.D$ is a d.c.e. real.

Theorem 8.5.6 (Ambos-Spies, Weihrauch, and Zheng [14])**.** *There is a d.c.e. set B such that $.B$ is not a left nor right computable real.*

Proof. Let C and D be c.e. Turing incomparable sets. Define the d.c. e. set B as follows.

$$B = \{4n : n \in \overline{C}\} \cup \{4n+1 : n \in C\} \cup \{4n+2 : n \in D\} \cup \{4n+3 : n \in \overline{D}\}.$$

Using the Calude et al. characterization of left-c.e. reals, because of the $4x, 4x+1$ part, $\cdot\chi_B$ cannot be left computable, lest $C \leqslant_T D$, and similarly by the obvious modification to part (iva) above (reversing), the same shows that $\cdot\chi_B$ is not right computable lest $D \leqslant_T C$. For instance, if $\cdot\chi_B$ is left computable, let f be the strongly Δ^0_2 approximation given in (iv) of the Theorem. Note that we can run the approximations to C and D and f so that at each stage we can have things looking correct. That is, we can speed the enumeration so that for all s and all $n \leqslant s$, $n \in C_s$ iff $f(4n+1, s) = 1$ and $f(4n, s) = 0$, $n \notin C_s$ iff $f(4n, s) = 1$ and $f(4n+1, s) = 0$, and similarly for D_s, since this must be true for C, D and f in the limit. Assume that we have such enumerations. We claim that $C \leqslant_T D$ contrary to hypothesis. Suppose inductively we have computed C up to $n-1$. Let $s > n$ be a stage where the current approximation $f(i,s) : i \leqslant 4n+3$ correctly computes $C \upharpoonright n-1$, Note this is okay by the induction hypothesis, and means that $C_s \upharpoonright n-1 = C \upharpoonright n-1$, $D \upharpoonright n = D_s \upharpoonright n$ using the D-oracle, and as above, f appears correct for C_s and D_s up to n. Then it can only be that $n \in C$ iff $n \in C_s$. (The point is that if n later enters C at $t > s$ then since $f(4n,t)$ must become 0 something must enter B smaller than $4n+1$. But this is impossible since we have $C \upharpoonright n-1 = C_s \upharpoonright n-1$, and $D \upharpoonright n = D_s \upharpoonright n$.) The non-right computability is entirely analogous. □

It is not difficult to see that that the Δ^0_2 reals form a field.

8.5.2 Rettinger's Theorem and d.c.e. random reals

A very interesting fact about d.c.e. reals is that they give no more random reals than the left c.e. and right c.e. reals.

Theorem 8.5.7 (Rettinger unpubl.)**.** *Suppose that x is a d.c.e. real and that x is random. Then x is either left or right c.e..*

Proof. Let A be a d. c. e. real that is neither c. e. nor co-c. e. By the characterization of d. c. e. reals from Lemma 8.5.2, there exists a computable sequence $\{q_i : i \in \mathbb{N}\}$ of rational numbers so that $\lim_i q_i = x$ and $\sum_i |q_{i+1} - q_i| < 1$. Then there are infinitely many indices i so that $q_i > x$, because otherwise let i_0 be the largest index so that $q_i > x$ and define the sequence $\{p_i : i \in \mathbb{N}\}$ by $p_i = \max\{q_j : i_0 < j \leqslant i_0 + 1 + i\}$. Obviously $\{p_i : i \in \mathbb{N}\}$ converges to x and is monotone, i.e. x is c. e., being a contradiction to the assumption. Analogously there are infinitely many indices i so that $q_i < x$, we get a contradiction.

But this means that $x \in (q_i; q_{i+1})$ for infinitely many i with $q_i < q_{i+1}$. Thus $\{(q_i; q_{i+1}) : q_i < q_{i+1}\}$ and analogously $\{(q_{i+1}; q_i) : q_i > q_{i+1}\}$ are Solovay tests for x, i.e. x is not random. □

8.5.3 The field of d.c.e. reals

The d.c.e. reals form the smallest field containg the left-c.e. reals. Additionally, this field is real closed.

Theorem 8.5.8 (Ng Keng Meng [224], Raichev [241, 242])**.** *The field of d.c.e. reals is real closed.*

Proof. The following proof is due to Ng Keng Meng [224]. We will let $\mathbb{R}_2$ denote the d.c.e. reals.

• In the subsequent lemmas, we will consider f to be a real function which is analytic at some $u_0 \in \mathbb{R}_2$. Let its Taylor series converge in the open interval E centered at u_0, and assume that there is a uniform recursive sequence of rationals $\langle a_{k,n}\rangle_{k<\omega} \to f^{(n)}(u_0)$, such that $\sup\limits_{n\in\mathbb{N}}\sum\limits_{k=0}^{\infty}|a_{k+1,n} - a_{k,n}| < \infty$. Suppose further that $f(x)$ has a root $r \in E$, and form the sequence of functions $\langle f_k \in \mathbb{Q}[x]\rangle_{k<\omega}$ by : $f_k(x) = a_{k,0} + a_{k,1}(x-u_0) + \cdots + a_{k,k}\frac{(x-u_0)^k}{k!}$.

Lemma 8.5.9. *There are rationals α, β, M, m, m' such that :*

- *$[\alpha, \beta]$ is an interval in E containing r,*
- *$\forall x \in [\alpha, \beta]$, $|f''(x)| < M$, and $0 < m < |f'(x)| < m'$.*

Proof. Let $r \in E$ be a real root of $f(x)$ of multiplicity $k+1$. So, $f(x) = (x-r)^{k+1} \cdot g(x)$ where $g(r) \neq 0$. Now if the multiplicity of $r > 1$, then instead of working with $f(x)$, we work with $f^{(k)}(x)$ since $f^{(k)}(r) = 0$ as well, and $f^{(k)}(x)$ satisfies the conditions $\bullet$. In any case, we may assume that r is a simple root of $f(x)$. So, $f'(r) \neq 0$ and hence there is some interval $[r-\delta, r+\delta]$ such that $\forall x \in [r-\delta, r+\delta], f'(x) \neq 0$. We can just let $[\alpha, \beta]$ be $[r-\delta, r+\delta]$ (we may assume that $\delta \in \mathbb{Q}$), and thus we have what we want, for some appropriately chosen m, m' and M. $\square$

We may assume, without the loss of generality that $u_0 = 0$, since if $u_0 \neq 0$, we can work with the function $f(x+u_0)$ and observe that it satisfies the conditions $\bullet$, and $r - u_0$ is the root of $f(x+u_0)$.

Lemma 8.5.10. *Let $\langle x_i\rangle_{i<\omega}$ be any sequence of points in a closed bounded interval $I \subseteq E$. Then, we have $\sum\limits_{k=0}^{\infty}|f_{k+1}(x_k) - f_k(x_k)| < \infty$.*

Proof. Let $T = \sup\limits_{n\in\mathbb{N}}\sum\limits_{k=0}^{\infty}|a_{k+1,n} - a_{k,n}|$, and $U = \sup\limits_{x\in I}|x|$.

We have

$$\begin{aligned}
&\sum_{k=0}^{\infty}|f_{k+1}(x_k) - f_k(x_k)| \\
&\leqslant \sum_{k=0}^{\infty}|a_{k+1,0} - a_{k,0}| + \cdots + |a_{k+1,k} - a_{k,k}|\frac{|x_k|^k}{k!} + |a_{k+1,k+1}|\frac{|x_k|^{k+1}}{(k+1)!} \\
&\leqslant \sum_{k=0}^{\infty}\sum_{n=0}^{\infty}|a_{k+1,n} - a_{k,n}|\frac{U^n}{n!} + \sum_{k=0}^{\infty}|a_{k,k}|\frac{U^k}{k!} \\
&\leqslant T \cdot \sum_{n=0}^{\infty}\frac{U^n}{n!} + \sum_{k=0}^{\infty}|a_{k,k}|\frac{U^k}{k!} \text{ (by Fubini's Theorem)} \\
&\leqslant 2Te^U + \sum_{k=0}^{\infty}|f^{(k)}(0)|\frac{U^k}{k!},
\end{aligned}$$

and the last inequality follows because $|a_{k,k}-f^{(k)}(0)| \leqslant \sum_{j=k}^{\infty}|a_{j+1,k}-a_{j,k}| \leqslant T$, for all k. Since I is closed, it must contain U, and the lemma is proved because the power series converges absolutely on I. □

It follows that for each $m \in \mathbb{N}$, $\sum_{k=0}^{\infty}|f_{k+1}^{(m)}(x_k) - f_k^{(m)}(x_k)| < \infty$ as well.

Lemma 8.5.11. *$f_k \to f$ (similarly $f_k' \to f'$ and $f_k'' \to f''$) uniformly on $[\alpha, \beta]$.*

Proof. We let $I \subseteq E$ be any closed bounded interval containing $[\alpha, \beta]$, and 0. Since for each $k \in \mathbb{N}$, the $\sup_{x \in I}|f_{k+1}(x) - f_k(x)|$ is attainable, we let the sequence of points $\langle x_k \in I\rangle_{k<\omega}$ be such that $|f_{k+1}(x_k) - f_k(x_k)| = \max_{x \in I}|f_{k+1}(x) - f_k(x)|$. Then, it follows from Lemma 8.5.10 that given any $\varepsilon > 0$, we can find $N_\varepsilon \in \mathbb{N}$ such that whenever $u > v \geqslant N_\varepsilon$, we have $|f_u(x) - f_v(x)| \leqslant \sum_{k=N}^{\infty}|f_{k+1}(x_k) - f_k(x_k)| < \varepsilon$ for all $x \in I$.

It now remains to show that $f_k(x) \to f(x)$ for every $x \in E$. Let $T_n = \sum_{j=0}^{\infty}|a_{j+1,n} - a_{j,n}|$, and we will have that

$$\begin{aligned}
&\left|\sum_{n=0}^{k+1} f^{(n)}(0)\frac{x^n}{n!} - f_{k+1}(x)\right| \\
\leqslant &\sum_{n=0}^{k+1}|f^{(n)}(0) - a_{k+1,n}|\frac{x^n}{n!} \\
\leqslant &\sum_{n=0}^{k+1}\left(T_n - \sum_{j=0}^{k}|a_{j+1,n} - a_{j,n}|\right)\frac{x^n}{n!} \\
\leqslant &\sum_{n=0}^{k+1}T_n\frac{x^n}{n!} - \sum_{n=0}^{k+1}\sum_{j=0}^{k}|a_{j+1,n} - a_{j,n}|\frac{x^n}{n!}
\end{aligned}$$

holds for all k. Now, since $\sum_{n=0}^{\infty}T_n\frac{x^n}{n!} \leqslant \sum_{n=0}^{\infty}\sup_{n \in \mathbb{N}}T_n\frac{x^n}{n!} < \infty$, it follows by Fubini's Theorem that $\sum_{n=0}^{k+1}\sum_{j=0}^{k}|a_{j+1,n} - a_{j,n}|\frac{x^n}{n!} \to \sum_{n=0}^{\infty}T_n\frac{x^n}{n!}$ (as $k \to \infty$). Since f is represented by it's Taylor series in E, hence $f_k(x) \to f(x)$. □

Lemma 8.5.12. *There is an integer s_0 such that whenever $s \geqslant s_0$,*

- *$[\alpha, \beta]$ will contain a simple root r_s of f_s,*
- *$\forall x \in [\alpha, \beta]$, $|f_s''(x)| < M$, and $0 < m < |f_s'(x)| < m'$.*

Proof. We may suppose that $f(\alpha) < 0 < f(\beta)$. Since $f_s(\alpha) \to f(\alpha)$ and $f_s(\beta) \to f(\beta)$, we choose a large enough $s_0 \in \mathbb{N}$ so that whenever $s \geqslant s_0$, we have $f_s(\alpha) < 0 < f_s(\beta)$, and hence there is a (simple) root $r_s \in [\alpha, \beta]$. Let $\varepsilon > 0$ be such that $\min\limits_{x \in [\alpha,\beta]} |f'(x)| - \varepsilon > m$. Since $f'_s \to f'$ uniformly on $[\alpha, \beta]$, thus we can assume that s_0 is large enough such that $|f'_s(x) - f'(x)| < \varepsilon$ for every $s \geqslant s_0$, and $x \in [\alpha, \beta]$. Hence, we have $|f'(x)| - |f'_s(x)| < \varepsilon$. (The case for m' and M is treated similarly). $\square$

Note that for each $s \geqslant s_0$, we have $f'_s(x) > 0$ inside $[\alpha, \beta]$ and hence r_s is the only root. Thus, the labelling is well-defined, and in fact we have :

Lemma 8.5.13. *$r_s \to r$ as $s \to \infty$.*

Proof. Let $\varepsilon > 0$ be given, and we may assume $\varepsilon < \delta$ (from Lemma 8.5.9). Due to pointwise convergence again, we can choose N_ε such that whenever $s \geqslant N_\varepsilon$, we have $f_s(r - \varepsilon) \cdot f_s(r + \varepsilon) < 0$ and hence there is a root of f_s between $[r - \varepsilon, r + \varepsilon] \subseteq [\alpha, \beta]$. Since r_s is the only root there, hence we must have $|r_s - r| < \varepsilon$. $\square$

Let $K = \frac{M}{2m}$, and fix η to be any number such that $\eta < \min\{\dfrac{1}{2K}, \dfrac{\delta}{2}\}$. For all $s \geqslant s_0$, we also want $|r_s - r| < \frac{\eta}{2}$, so we may have to adjust s_0 to be large enough to suit this requirement. For simplicity, we will assume from now on that the sequence $\langle f_s \rangle_{s<\omega}$ starts with the index s_0. Let y_0 to be any rational chosen such that $|y_0 - r| < \frac{\eta}{2}$. For each $s \in \mathbb{N}$, define the recursive sequence of rationals $\langle x_{s,n} \rangle_{n<\omega}$ by :

$$x_{s,0} = y_0,$$
$$x_{s,n+1} = x_{s,n} - \frac{f_s(x_{s,n})}{f'_s(x_{s,n})}.$$

Lemma 8.5.14 (Newton's Method of Locating Roots)**.** *For each s, if $x' \in [\alpha, \beta]$, and $x'' = x' - \dfrac{f_s(x')}{f'_s(x')}$, then we have $|x'' - r_s| \leqslant K|x' - r_s|^2$.*

Proof. By Taylor expansion of $f_s(x)$ about the point x', we have

$$-f_s(x') = f'_s(x')(r_s - x') + \frac{f''_s(c)(r_s - x')^2}{2}$$

for some $r_s \leqslant c \leqslant x'$ (or the other way round). Hence we have

$$x'' = x' - \frac{f_s(x')}{f'_s(x')} = r_s + \frac{f''_s(c)(r_s - x')^2}{2f'_s(x')}$$

Since $c \in [\alpha, \beta]$, we have

$$|x'' - r_s| = |\frac{f''_s(c)(r_s - x')^2}{2f'_s(x')}| \leqslant K|x' - r_s|^2. \qquad \square$$

Lemma 8.5.15. *For all s and n, we have*

$$|x_{s,n+1} - x_{s,n}| \leqslant \Big((K\eta)^{n+1} + (K\eta)^n\Big)|y_0 - r_s|.$$

Proof. Fix an s, and we first prove that $|x_{s,n} - r_s| < \eta$ for all n. First note that $|x_{s,0} - r_s| \leqslant |x_{s,0} - r| + |r_s - r| < \eta$. Suppose that $|x_{s,n} - r_s| < \eta$, and now, $|x_{s,n} - r| \leqslant |x_{s,n} - r_s| + |r_s - r| < 2\eta < \delta$ so that $x_{s,n} \in [\alpha, \beta]$. So by Lemma 8.5.14, we have $|x_{s,n+1} - r_s| \leqslant K\eta^2 < \eta$.

Next, we prove that $|x_{s,n} - r_s| \leqslant (K\eta)^n |y_0 - r_s|$ by induction. We have $|x_{s,0} - r_s| = |y_0 - r_s|$. Now assume that the case for n holds, and by the Lemma 8.5.14 above, we have $|x_{s,n+1} - r_s| \leqslant K|x_{s,n} - r_s|^2 < (K\eta)|x_{s,n} - r_s| \leqslant (K\eta)^{n+1}|y_0 - r_s|$.

Lastly, the result will follow by triangle inequality. □

From now on we will assume that f and f_s are all increasing on $[\alpha, \beta]$. For each s, we can construct partitions $I_0, I_1, \cdots$ of $[\alpha, \beta]$ where the width is $\frac{1}{2^{s+1}m'}$. Then apply the sign test on f_s using $I_0, I_1, \cdots$ and let r'_s be the right end-point of the partition catching the root r_s.

Lemma 8.5.16. *For each s, we can find a $N_s \in \mathbb{N}$ such that*

$$|x_{s+1,N_s} - x_{s,N_s}| < \frac{1}{m}|f_s(r'_{s+1})|.$$

Proof. From the proof of Lemma 8.5.15, it is clear that $x_{s,n} \to r_s$ as $n \to \infty$. Thus, for each s, we have $|x_{s+1,n} - x_{s,n}| \to |r_{s+1} - r_s|$. However, $|r_{s+1} - r_s| = \dfrac{|f_s(r_{s+1})|}{f'_s(c)}$ for some $r_s \leqslant c \leqslant r_{s+1}$, so that $|r_{s+1} - r_s| \leqslant \dfrac{|f_s(r_{s+1})|}{m} < \dfrac{|f_s(r'_{s+1})|}{m}$. □

Define the recursive sequence $\langle y_n \rangle_{n<\omega}$ by the following :
y_0 is as defined before,
If $|x_{1,1} - x_{0,1}| \geqslant \frac{1}{m}|f_0(r'_1)|$, then let $y_1 = x_{0,1}$, and continue until we find some N_0 such that $|x_{1,N_0} - x_{0,N_0}| < \frac{1}{m}|f_0(r'_1)|$ (There is one, by Lemma 8.5.16). Then, let $y_{N_0} = x_{0,N_0}$, $y_{N_0+1} = x_{1,N_0}$, and $y_{N_0+2} = x_{1,N_0+1}$. The idea is that we wait until $|x_{s+1,n} - x_{s,n}|$ is small enough before we jump to the next level.

Lemma 8.5.17. *$y_n \to r$ as $n \to \infty$.*

Proof. We first observe that for any s and n, there is a c such that $y_c = x_{s',n'}$ with $s' > s$ and $n' > n$. So now let $\varepsilon > 0$ be given. Then, there is some t_0 such that $\forall a \geqslant t_0$, $|r_a - r| < \frac{\varepsilon}{4}$, and let $u_0 = \dfrac{\log(\frac{\varepsilon}{\eta})}{\log(k\eta)}$. Let c be obtained from t_0 and u_0 as described above. Then, for all $n \geqslant c$, we have $y_n = x_{a,b}$ for some $a > t_0$ and $b > u_0$. In that case, $|y_n - r| \leqslant |x_{a,b} - r_a| + |r_a - r| < (K\eta)^b|y_0 - r_a| + \frac{\varepsilon}{4} \leqslant (K\eta)^b|y_0 - r| + \frac{\varepsilon}{4}(K\eta)^b + \frac{\varepsilon}{4} < \frac{\eta}{2}(K\eta)^b + \frac{\varepsilon}{2} < \varepsilon$. □

Lemma 8.5.18. *$r \in \mathbb{R}_2$.*

Proof. It remains to show that $\sum_{i=0}^{\infty} |y_{i+1}-y_i| < \infty$. We have $\sum_{i=0}^{\infty} |y_{i+1}-y_i| \leqslant \sum_{i=0}^{\infty} \Big((K\eta)^{i+1} + (K\eta)^i \Big) |y_0 - r_{d(i)}| + \sum_{s=0}^{\infty} \frac{1}{m} |f_s(r'_{s+1})|$, by considering the two different cases. Here, $d(i)$ is a non decreasing sequence of integers. Since $|r_{s+1} - r'_{s+1}| < \frac{1}{2^{s+1}m'}$, by the Mean Value theorem, we have $\frac{1}{m}|f_{s+1}(r'_{s+1})| \leqslant m' \frac{1}{2^{s+1}m'} < \frac{1}{2^s}$. By Lemma 8.5.10, finally we have

$$\begin{aligned}\sum_{i=0}^{\infty} |y_{i+1} - y_i| \quad &\leqslant \quad \sum_{i=0}^{\infty} \Big((K\eta)^{i+1} + (K\eta)^i \Big) L \\ &\quad + \sum_{s=0}^{\infty} \frac{1}{m} |f_{s+1}(r'_{s+1}) - f_s(r'_{s+1})| + \sum_{s=0}^{\infty} \frac{1}{2^s} < \infty.\end{aligned}$$

□

Lemma 8.5.19. *Let f be a real function which is analytic at some $u_0 \in \mathbb{R}_2$. Let its Taylor series converge in some open interval E centered at u_0, and assume that there is a uniform recursive sequence of rationals $\langle a_{k,n} \rangle_{k<\omega} \to f^{(n)}(u_0)$, such that $\sup_{n \in \mathbb{N}} \sum_{k=0}^{\infty} |a_{k+1,n} - a_{k,n}| < \infty$.*
Then, every root of $f(x)$ in E is also in $\mathbb{R}_2$.

Proof. Combine all of the above. □

Finally we are ready to finish the proof that the d.c.e. reals form a real closed field. Any polynomial $f(x) \in \mathbb{R}_2[x]$ must satisfy the conditions •, since for all m sufficiently large, $f^{(m)}(0) = 0$. Since $\sqrt{\gamma}$ is a root of $x^2 - \gamma$ for every positive d.c.e. real γ, the result then follows. □

We remark that essentially the same method shows the following.

Corollary 8.5.20 (Raichev [241, 242])**.** *The Δ_2^0 reals form a real closed field.*

This will also follow from the fact that the computable reals form a real closed field and relativization. The method of proof of this last result is again similar and is omitted for space considerations.

8.5.4 D.c.e reals and the Ershov Hierarchy

The results above would suggest that the d.c.e. reals might be like n-c.e. reals where we recall the following definition.

Definition 8.5.21 (Ershov [101, 102])**.** The *difference hierarchy* is defined as follows:

(i) A set $A \subseteq \omega$ is called *n-computably enumerable* (n-c.e., for short), if there is a computable function f such that for al l $x \in \omega$,

(a) $f(x, 0) = 0$,
(b) $\lim_s f(x, s) \downarrow= A(x)^2$, and
(c) $|\{s + 1 \mid f(x, s) \neq f(x, s + 1)\}| \leqslant n$.

(ii) A set $A \subseteq \omega$ is ω-c.e. if and only if there are two computable functions $f(x, s), g(x)$ such that for all $x \in \omega$,

(a) $f(x, 0) = 0$,
(b) $\lim_s f(x, s) \downarrow= A(x)$, and
(c) $|\{s + 1 \mid f(x, s) \neq f(x, s + 1)\}| \leqslant g(x)$.

(iii) A set $A \subseteq \omega$ is *α-computably enumerable* (α-c.e., for short) relative to a computable system $\mathcal{S}$ of notations for α if and only if there is a *partial computable* (p.c.) function f such that for all k, $A(k) = f(k, b)$, where b is the $\mathcal{S}$-least notation x such that $f(k, x)$ converges[3]

Proposition 8.5.22 (Arslanov [18])**.** *Let A be any ω-c.e. set. Then there exists an ω-c.e. set $B \equiv_T A$ and a computable function $f(x, s)$ such that for all $x \in \omega$,*

(a) $f(x, 0) = 0$,

(b) $\lim_s f(x, s) \downarrow= B(x)$, *and*

(c) $|\{s + 1 \mid f(x, s) \neq f(x, s + 1)\}| \leqslant x$.

Here is a simple proof of this fact, which was surely known before it was stated by Arslanov. Suppose that A is ω-c.e. with at most $g(x)$ mind changes for each x. Without loss of generality we can suppose that g is an increasing computable function. Now define B by putting $g(x) \in B$ iff $x \in A$. Then $B \equiv_m A$ and B is ω-c.e. via the identity function.

It is easy to see that for any n-c.e. set A, the binary expansion of A is a d.c.e. real. The converse is not true.

Theorem 8.5.23 (Zheng [330])**.** *There is a d.c.e. real not contained in any ω-c.e. degree.*

Proof. To build such an A we diagonalize against requirements listing the e-th triple consisting of a φ_e-c.e. set V_e[4] pair of Turing procedures, Φ_e, Γ_e

---

[2] Here, as usual, we identify sets and reals with their characteristic functions.

[3] Since we will not be using this notion elsewhere in the book, for simplicity, we will assume that the reader is familiar with the definitions of a computable ordinal and Kleene's ordinal notations. We refer the reader to Rogers [253] and Odifreddi [233, 234]. Additionally, we refer the reader to Epstein, Haas, and Kramer [100] for more on α-c.e. sets and degrees.

[4] That is, φ_e is a partial computable function, and anu n is allowed to enter and leave V_e at most $\varphi_e(n)$ many times. The enumeration of V_e on argument n can be initially 0

meeting

$$R_e : \Gamma_e^{V_e} \neq A \vee \Phi_e^A \neq V_e.$$

Additionally we will build left-c.e. *reals* B, C, with $A = B - C$. To meet R_e we monitor the length of agreement between V_e and A defining the length of agreement as

$$\ell(e, s) = \max\{x : \forall y \leqslant x(\Gamma_e^{V_e}(y) = A(y) \wedge \Phi^A \upharpoonright (\varphi(y)) = V_e(\varphi(y))[s])\}.$$

The the argument is finite injury. We pick some follower $x = x(e)$. and keep $A(x) = 0$ and $A(x-1) = 0$, by keeping both x and $x - 1$ out of both B and C. We await some stage s where $\ell(e, s) > x$. After this stage, we will promise to freeze $A \upharpoonright x - 1\mathcal{A}[s] \upharpoonright x - 1$, by initializing the lower priority requirements.

Now the idea is to change A so many times that V_e being only φ_e-c.e. cannot respond enough. We want to change $A(x) = 1$. The easiest way to do this is to simply add 2^{-x} to B_s and keep C_s the same. Of course, if nothing else changes, then V_e must below $\gamma_e(x)[s]$.

Now we then want to be able to take x *out* of A, and repeat this process many times. Clearly we can't do this by taking 2^{-x} out of B_t at $t > s$ as B is a left-c.e. real.

The idea then is as follows. We will add to B_s the rational $0^{x-1}1^n0^n$ where $n = n(x, e)$ is a sufficiently large number. Then to C_s we add $0^x1^n0^n$. Additionally, R_e asserts control of this region of B and C. Notice that this action will, in effect add 2^{-x} to A_s as required.

Later when we wish to change $A_t \upharpoonright s = A_s \upharpoonright s$, all we need to do is to add $0^{n+x}10^{n-1}$ to C_t. We can then repeat this process by adding this back into $B_{t'}$ at some stage $t' > t$, etc. Choosing the number n large enough V_e can't respond sufficiently many times. The rest is a standard finite injury argument. □

We remark that the proof above can be modified to prove the following.

Corollary 8.5.24. *Suppose that $\alpha < \omega^2$ is a computable ordinal. Then there is a d.c.e. real not of any α-c.e. degree.*

This result will be obvious to anyone who is familiar with the notions once they have seen the proof of Zheng's Theorem. We also remark also that this result is best possible in terms of the Ershov hierarchy, since, as is well known, every Δ_2^0 set is ω^2-c.e. for a suitably chosen notation for ω^2. We refer the reader to Epstein, Haas and Kramer [100] for more on this.

We have seen that every n-c.e. degree contains a d.c.e. real. We can push this one level further.

---

and not allowed to change unless $\varphi(j) \downarrow$ for all $j \leqslant n$; and the enumeration of the V_e can be taken as primitive recursive by slowing things down.

Theorem 8.5.25 (Downey, Wu, Zheng [99])**.** *Let* **a** *be any ω-c.e. degree, then* **a** *contains a d.c.e. real.*

Proof. The proof of Theorem 8.5.25 is separated into two parts. First, we prove that if the bounding function does not increase too fast, then the binary expansion of the corresponding set is a d.c.e. real.

Lemma 8.5.26. *Let A be any ω-c.e. set, and f, g be two functions given in Definitio n 8.5.21. If $\sum_{x\in\mathbb{N}} g(x)\cdot 2^{-x}$ is bounded, then .A is a d.c.e. real.*

Proof. Let c be a constant such that $\sum_{x\in\mathbb{N}} g(x)\cdot 2^{-x} \leqslant c$. Without loss of generality, suppose that $|A_{s+1}\Delta A_s| \leqslant 1$ for any s. Then $\{0.A_s : s\in\mathbb{N}\}$ is a computable sequence of rational numbers converging to $0.A$ and

$$\sum_{s\in\mathbb{N}} |0.A_s - 0.A_{s+1}| = \sum_{s\in\mathbb{N}}\{2^{-x} : x\in A_{s+1}\Delta A_s\} = \sum_{x\in\mathbb{N}} 2^{-x}\cdot g(x) \leqslant c.$$

By Proposition 8.5.2, $0.A$ is d.c.e.. □

Now we combine Lemma 8.5.26 with Proposition 8.5.22 to give a proof of Theorem 8.5.25.

Proof. (of Theorem 8.5.25) Let **a** be any ω-c.e. Turing degree and $A\in\mathbf{a}$ be an ω-c.e. set. By Proposition 8.5.22, there is an ω-c.e. set B Turing equivalent to A and a computable function f satisfying (a)-(c) in Proposition 8.5.22. Since $\sum_{n\in\mathbb{N}} n\cdot 2^{-n}\leqslant 2$ is bounded, by Lemma 8.5.26, $0.B$ is a d.c.e. real. Therefore, **a** contains a d.c.e real. □

□

8.5.5 A Δ^0_2 degree containing no d.c.e. reals

Since, as we have seen, Demuth [62] and Ho [127] proved whilst we can construct degrees without that every Δ^0_2 real is the limit of a computable sequence of rationals. Combining Zheng's result, and Ho's result, it even seems reasonable to conjecture that perhaps *every* Δ^0_2 degree contains a d.c.e. real. In this section, we construct a Δ^0_2 set (indeed, an $\omega+1$-c.e. set) not Turing equivalent to any d.c.e. real.

Theorem 8.5.27 (Downey, Wu, Zheng [99])**.** *There are Δ^0_2 degrees containing no d.c.e. reals.*

Proof. We construct a Δ^0_2-set A which is not Turing equivalent to any d.c.e. real. That is, A is constructed to satisfy the following requirements:

$$\mathcal{R}_e : A \neq \Phi_e^{\alpha_e-\beta_e} \ \vee\ \alpha_e-\beta_e \neq \Psi_e^A \tag{8.1}$$

where $\{\langle\Phi_e,\Psi_e,\alpha_e,\beta_e\rangle : e\in\mathbb{N}\}$ is an effective enumeration of all 4-tuples $\langle\Phi,\Psi,\alpha,\beta\rangle$, Φ,Ψ computable functionals, and α,β left-c.e. reals. We say that requirement $\mathcal{R}_e$ has priority higher than $\mathcal{R}_{e'}$ if $e<e'$.

A is constructed as a Δ_2^0 set by stages. Let A_s be the approximation of A at the end of stage s. Then $A = \lim_{s\to\infty} A_s$. We now describe a strategy satisfying a single requirement. First we define the length function of agreement for $\mathcal{R}_e$ at stage s as follows:

$$l(e,s) = \max\{x : A_s(x) = \Phi_{e,s}^{\alpha_{e,s}-\beta_{e,s}}(x) \& (\alpha_{e,s}-\beta_{e,s}) \upharpoonright \varphi_{e,s}(x) = \Psi_{e,s}^{A_s} \upharpoonright \varphi_{e,s}(x)\},$$

where φ_e is the use function of the functional Φ_e. Our strategy will ensure that $l(e,s)$ is bounded during the construction, and hence $\mathcal{R}_e$ is satisfied.

We first choose a witness x as a big number and wait for a stage s with $l(e,s) > x$. Put x into A, and wait for another stage $s' > s$ with $l(e,s') > x$. If there is no such a stage, then $\mathcal{R}_e$ is evidently satisfied. Otherwise, we have that $\alpha - \beta$ changes below $\varphi_{e,s}(x)$ between stages s and s'. That is, $\Psi_e^A \upharpoonright \varphi_{e,s}(x)$ changes between s and s'. Note that the only small number enumerated into A between s and s' is x, so by taking x out of A, we recover the computations $\Psi_e^A \upharpoonright \varphi_{e,s}(x)$ to $\Psi_{e,s}^{A_s} \upharpoonright \varphi_{e,s}(x)$, and we have a temporary disagreement between $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$ and $\Psi_e^A \upharpoonright \varphi_{e,s}(x)$. If $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$ fails to change later, then by preserving A on $\psi_{e,s}(\varphi_{e,s}(x))$, we will have

$$(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x) \neq \Psi_e^A \upharpoonright \varphi_{e,s}(x),$$

and $\mathcal{R}_e$ is satisfied again. By iterating such a procedure, we put x into A and take x out of A alternatively, trying to get a disagreement between A and $\Phi_e^{(\alpha_e-\beta_e)}$ or between $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$ and $\Psi_e^A \upharpoonright \varphi_{e,s}(x)$. It is easy to check that if $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$ change only finitely often, then we can get the wanted disagreement eventually, and $\mathcal{R}_e$ is satisfied. However, $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$ may change infinitely often, as pointed out below, even though both $\alpha_e \upharpoonright \varphi_{e,s}(x)$ and $\beta_e \upharpoonright \varphi_{e,s}(x)$ settle down after a sufficiently large stage.

Fix i. $(\alpha_e - \beta_e)(i)$ can be changed by changes of $\alpha_e(j)$ or $\beta_e(j)$, where $j > i$. For example, let

$$\alpha_{e,1} = \alpha_{e,2} = 0.101w0, \quad \beta_{e,1} = 0.100w1 \text{ and } \beta_{e,2} = 0.100w0,$$

for some $w \in \{0,1\}^n$ and $n \in \mathbb{N}$. Then we have

$$\alpha_{e,1} - \beta_{e,1} = 0.0010\overbrace{1\cdots1}^{n}1 \quad \text{and} \quad \alpha_{e,2} - \beta_{e,2} = 0.0011\overbrace{0\cdots0}^{n}0.$$

The change of $\beta_e(n+4)$ from 1 to 0 leads to the change of $(\alpha_e - \beta_e)(4)$ from 1 to 0. We call such a change of $\alpha_e - \beta_e$ as a " nonlocal-disturbance". Note that $(\alpha_e - \beta_e)(4)$ can be changed infinitely often by these nonlocal-disturbances since we have infinitely many such ws. Fortunately, if such a "nonlocal-disturbance" happens, then the corresponding segments of $\alpha_e - \beta_e$ will be in quite simple forms. This is summarized below:

Proposition 8.5.28. *Let $\alpha^j = 0.a_1^j a_2^j \cdots a_n^j$, $\beta^j = 0.b_1^j b_2^j \cdots b_n^j$ and $\alpha^j - \beta^j = 0.c_1^j c_2^j \cdots c_n^j$ for $j = 0, 1$. If there are numbers $i < k \leqslant n$ such that*

$c_i^0 \neq c_i^1$, and $a_t^0 = a_t^1$, $b_t^0 = b_t^1$ for all $t \leqslant k$. Then, there is a $j \in \{0,1\}$ such that

$$c_i^j c_{i+1}^j \cdots c_k^j = 011\cdots 1 \qquad \& \qquad c_i^{1-j} c_{i+1}^{1-j} \cdots c_k^{1-j} = 100\cdots 0. \tag{8.2}$$

Now let's turn back to consider $(\alpha_e - \beta_e) \upharpoonright \varphi_{e,s}(x)$. Suppose that both $\alpha_e \upharpoonright \varphi_{e,s}(x)$ and $\beta_e \upharpoonright \varphi_{e,s}(x)$ do not change after a stage large enough, s_1 say, then by Proposition 8.5.28, the initial segment $(\alpha_e - \beta_e) \upharpoonright \varphi(x)$ can have only one of two different forms: $0.w011\cdots 1$ or $0.w100\cdots 0$ for some fixed binary word w. It leads us to use two-attacks to satisfy $\mathcal{R}_e$, instead of using a single attacker. That is, at stage s', instead of taking x out of A, we put $x-1$ into A and wait for a stage $s'' > s'$ with $l(e,s'') > x$. At stage s'', we take $x-1$ and x out of A, and wait for a stage $s''' > s''$ with $l(e,s''') > x$. As a consequence, $A \upharpoonright \psi_e(\varphi_{e,s}(x))$ is recovered to that of stage s. Now we have three uses of $\varphi_e(x)$, i.e., $\varphi_{e,s}(x)$, $\varphi_{e,s'}(x)$, and $\varphi_{e,s''}(x)$. At stage $s'''+1$, we will have $(\alpha_{e,s'''} - \beta_{e,s'''}) \upharpoonright \varphi_{e,s}(x) = (\alpha_{e,s} - \beta_{e,s}) \upharpoonright \varphi_{e,s}(x)$. As in stage s, we put x into A again. We call the procedure between s and $s'''+1$ a complete cycle.

Let k be the maximum among $\varphi_{e,s}(x)$, $\varphi_{e,s'}(x)$, and $\varphi_{e,s''}(x)$. Then in a complete cycle, $(\alpha_e - \beta_e) \upharpoonright k$ has three different forms. By Proposition 8.5.28, in each complete cycle, at least one of α_e and β_e must have a change below k. Since α_e, β_e are both left-c.e., we can assume that after a stage t large enough, $\alpha_e \upharpoonright k$ and $\beta_e \upharpoonright k$ don't change anymore, and therefore, after stage t, no cycle can be complete. As a consequence, one of the combinations of $A(x-1)$ and $A(x)$, 00, 01, or 11, satisfies the requirement $\mathcal{R}_e$.

We describe the whole construction of A below.

Construction of A

During the construction, say that a requirement $\mathcal{R}_e$ *requires attention* at stage $s+1$ if x_e is defined and $l(e,s) > x_e$. When we initialize a requirement $\mathcal{R}_e$, all parameters associated will be canceled.

Stage $s=0$: Do nothing.

Stage $s+1$: If no requirement requires attention at stage $s+1$, then choose a least e such that x_e is not defined and define $x_e = s+2$.

Otherwise, let $\mathcal{R}_e$ be the requirement of the highest priority requiring attention and define

$$A_{s+1}(x_e-1)A_{s+1}(x_e) = \begin{cases} 01 & \text{if } A_s(x_e-1)A_s(x_e) = 00; \\ 11 & \text{if } A_s(x_e-1)A_s(x_e) = 01; \\ 00 & \text{if } A_s(x_e-1)A_s(x_e) = 11. \end{cases} \tag{8.3}$$

Initialize all the requirements with lower priority, and declare that $\mathcal{R}_e$ *receives attention* at stage $s+1$.

This completes the construction.

We now verify that A constructed above satisfies all the requirements. We only need to prove the following lemma.

Lemma 8.5.29. *For any $e \in \mathbb{N}$, $\mathcal{R}_e$ requires and receives attention finitely often.*

Proof. We prove Lemma 8.5.29 by induction on e. Assume that, for any $i < e$, $\mathcal{R}_i$ requires and receives attention only finitely often. Let s_0 be the least stage after which no requirement $\mathcal{R}_i$, $i < e$, requires attention. By the choice of s_0, R_e is initialized at stage s_0. Let $s_1 > s_0$ be the stage at which x_e is defined. Since $\mathcal{R}_e$ cannot be initialized after stage s_0, x_e cannot be canceled afterwards. We prove below that after a stage large enough, $s_2 > s_1$ say, $\mathcal{R}_e$ does not require attention anymore, and hence $l(e,s)$ cannot exceed x_e for any $s > s_2$, $\mathcal{R}_e$ is satisfied.

For a contradiction, suppose that after stage s_1, there are infinitely many stages $t_0+1, t_1+1, t_2+1, \cdots$ at which $\mathcal{R}_e$ requires attention. Then, at stage t_0+1, we have $l(e,t_0) > x_e$, $A_{t_0}(x_e-1)A_{t_0}(x_e) = 00$, $A_{t_0+1}(x_e-1)A_{t_0+1}(x_e) = 01$. By the choice of s_0, no requirement with higher priority can put numbers less than $\psi_{e,t_0}(\varphi_{e,t_0}(x_e))$ into A. Since all requirements with lower priority are initialized at stage t_0+1, when these requirements receive attention after t_0+1, the numbers they put into A or take out of A are all larger than t_0, and hence larger than $\psi_{e,t_0}(\varphi_{e,t_0}(x_e))$. Therefore, the computations $\Psi^{A_{t_0}}_{e,t_0}(\varphi_{e,t_0}(x_e))$ can only be changed by $\mathcal{R}_e$ itself by changing $A(x_e-1)$ or $A(x_e)$. Thus, by a simple induction, we have for all $n \in \mathbb{N}$,

$$A_{t_0} \upharpoonright \psi_{e,t_0}(\varphi_{e,t_0}(x_e)) = A_{t_{3n}} \upharpoonright \psi_{e,t_0}(\varphi_{e,t_0}(x_e))$$

because $A(x_e-1)A(x_e)$ always changes in the order $00 \to 01 \to 11 \to 00$. Therefore,

$$(\alpha_{e,t_{3n}} - \beta_{e,t_{3n}}) \upharpoonright \varphi_{e,t_0}(x_e) = (\alpha_{e,t_0} - \beta_{e,t_0}) \upharpoonright \varphi_{e,t_0}(x_e).$$

This means that the computation $\Phi^{A_{t_{3n}}}_{e,t_{3n}}(x_e)$ is actually the same as that of $\Phi^{A_{t_0}}_{e,,t_0}(x_e)$. Similarly, we can prove that the computation $\Phi^{A_{t_{3n+1}}}_{e,t_{3n+1}}(x_e)$ is the same as that of $\Phi^{A_{t_1}}_{e,t_1}(x_e)$, and the computation $\Phi^{A_{t_{3n+2}}}_{e,t_{3n+2}}(x_e)$ is the same as that of $\Phi^{A_{t_2}}_{e,t_2}(x_e)$.

Let $k = \max\{\varphi_{e,t_i}(x_e) : i \leqslant 3\}$. Choose an n large enough such that $l(e,t_n) > x_e$, and

$$\alpha_{e,t_n} \upharpoonright k = \alpha_{e,t} \upharpoonright k \quad \& \quad \beta_{e,t_n} \upharpoonright k = \beta_{e,t} \upharpoonright k$$

for any $t \geqslant t_n$. Without loss of generality, suppose that $n = 3m$ for some m. Then $A_{t_n}(x_e-1)A_{t_n}(x_e) = 00$, $A_{t_{n+1}}(x_e-1)A_{t_{n+1}}(x_e) = 01$, and

$A_{t_{n+2}}(x_e-1)A_{t_{n+2}}(x_e) = 11$. By our choices of t_n, t_{n+1}, t_{n+2}, we have

$$\Phi_e^{\alpha_e-\beta_e}(x_e-1)[t_n]\Phi_e^{\alpha_e-\beta_e}(x_e)[t_n] = 00, \quad \text{and}$$

$$\Phi_e^{\alpha_e-\beta_e}(x_e-1)[t_{n+1}]\Phi_e^{\alpha_e-\beta_e}(x_e)[t_{n+1}] = 01.$$

This implies that

$$(\alpha_{e,t_n}-\beta_{e,t_n}) \upharpoonright \varphi_{e,t_n}(x_e) \neq (\alpha_{e,t_{n+1}}-\beta_{e,t_{n+1}}) \upharpoonright \varphi_{e,t_{n+1}}(x_e)$$

and hence $(\alpha_{e,t_n}-\beta_{e,t_n}) \upharpoonright k \neq (\alpha_{e,t_{n+1}}-\beta_{e,t_{n+1}}) \upharpoonright k$. By Proposition 8.5.28, there exists a binary word w such that $(\alpha_{e,t_n}-\beta_{e,t_n}) \upharpoonright k$ takes one of the forms $0.w100\cdots0$ and $0.w011\cdots1$, and $(\alpha_{e,t_{n+1}}-\beta_{e,t_{n+1}}) \upharpoonright k$ takes the other one. Assume that $(\alpha_{e,t_n}-\beta_{e,t_n}) \upharpoonright k$ takes the form $0.w100\cdots0$ and $(\alpha_{e,t_{n+1}}-\beta_{e,t_{n+1}}) \upharpoonright k$ takes the form $0.w011\cdots1$. By the same argument, since $(\alpha_{e,t_{n+1}}-\beta_{e,t_{n+1}}) \upharpoonright k$ takes the form $0.w011\cdots1$, we know that $(\alpha_{e,t_{n+2}}-\beta_{e,t_{n+2}}) \upharpoonright k$ takes the form $0.w100\cdots0$. Thus,

$$(\alpha_{e,t_{n+2}}-\beta_{e,t_{n+2}}) \upharpoonright k = (\alpha_{e,t_n}-\beta_{e,t_n}) \upharpoonright k,$$

and hence

$$00 = \Phi_e^{\alpha_e-\beta_e}(x_e-1)[t_n]\Phi_e^{\alpha_e-\beta_e}(x_e)[t_n] = \Phi_e^{\alpha_e-\beta_e}(x_e-1)[t_{n+2}]\Phi_e^{\alpha_e-\beta_e}(x_e)[t_{n+2}] = 11.$$

A contradiction. Therefore, after stage t_n, $\mathcal{R}_e$ can require (and hence receive) attentions at most two more times.

This ends the proof of Lemma 8.5.29. □

□

Part II

Randomness of Reals

9
Complexity of reals

9.1 Introduction

In a way, the definition of algorithmic randomness is more satisfying for reals than it is for strings. One of the main reasons for this, as we will see, is that it is absolute, in that no matter which universal machine one chooses, the class of random reals remains the same, rather than just asymptotically the same.

There are three cornerstone approaches to the definition of algorithmic randomness for reals.

(i) *The computational paradigm*: Random reals should be those whose initial segments are all hard to compress. This is probably the easiest approach to understand in terms of the previous sections.

(ii) *The measure-theoretical paradigm*: Random reals should be those with no *effectively rare* properties. If the class of reals satisfying a given property is an effectively null set, then a random real should not have this property. This approach is the same as the *stochasticity paradigm*: a random real should pass all effective statistical tests. (See below for more on this notion.)

(iii) *The unpredictability paradigm*: This approach stems from the most common answer one would get if one were to ask mathematicians what a random real should be. Namely, that the sequence of bits of a random real should be such that one cannot predict the next bit even if one knows all preceding bits, in the same way that a coin toss is unpredictable even given the results of previous coin tosses.

In this chapter, we will examine the first two approaches. We leave the third approach to Chapter 10, in which it will be used as a motivation for the notions of *computable randomness* and *Schnorr randomness*. We will see that, at least for the most common notion of algorithmic randomness, *Martin-Löf randomness*, the notions of randomness arising from the three approaches coincide. As to how these approaches developed historically, and the philosophical insights behind them, we refer the reader to Van Lambalgen's thesis [314].

9.2 The computational paradigm

9.2.1 C-oscillations

A first attempt to define a random real would be to say that A is random if $C(A \upharpoonright n) \geqslant n - O(1)$. Unfortunately, *no* real satisfies this condition. This is a fundamental observation of Martin-Löf which we have already seen in Theorem 6.1.5, where we also disproved the formula $C(xy) \leqslant C(x) + C(y) + O(1)$. Recall that the idea was to take as input a segment τ of a string $\sigma\tau$ such that the length of τ codes σ. This reasoning is refined in the following result of Martin-Löf, the proof here being drawn from Li-Vitanyi [185], 2.5.

Lemma 9.2.1 (Martin-Löf [199]; see Li-Vitanyi, [185] and Staiger [288])**.** *Let f be a total computable function such that $\sum_n 2^{-f(n)} = \infty$. Then for any real α we have $C(\alpha \upharpoonright n \mid n) \leqslant n - f(n)$ for infinitely many n.*

Proof. Let $\widehat{f}(n) = \lfloor \log(\sum_{i=1}^{n} 2^{-f(i)} \rfloor$. (This is to get rid of a constant term in the computations below.) Noticing that $\sum_{\widehat{f}(n)=m} 2^{-f(n)} \geqslant 2^m - 1$, if we define a computable $g = f(n) + \widehat{f}(n)$, we have

$$\sum_{n=1}^{\infty} 2^{-g(n)} = \sum_{m=1}^{\infty} \sum_{\widehat{f}(n)=m} 2^{-f(n)-m} \geqslant \sum_{m=1}^{\infty} 2^{-m}(2^m - 1) = \infty.$$

Now we use a trick: consider the closed interval $[0, 1]$ as being laid out in a circle, so that 0 and 1 are identified. Then we have intervals $I_n = [\sum_{i=1}^{n-1} 2^{-g(i)}, \sum_{i=1}^{n} 2^n)(\mod 1)$ marking off intervals around the circle. Let $G_n = \sum_{i=1}^{n} 2^{-g(n)}$. Li-Vitanyi's idea is that any point on the circle will lie in many of these intervals and this is computationally short to resurrect. To wit: for each $x \in 2^{<\omega}$, recall that $[x]$ as the set of reals with initial segment x, so that $[x] = [0.x, 0.x + 2^{-|x|})$, as used in the proof of Kraft's inequality, Theorem 6.6.1. Define

$$A_n = \{x \in 2^n : [x] \cap I_n \neq \emptyset\}.$$

Since $\widehat{f}$ diverges, for any α, there is an infinite set $N \subseteq \mathbb{N}$, consisting of infinitely many n, with $\alpha \upharpoonright n \in A_n$. We can describe $\alpha \upharpoonright n$ by its position (as a binary string) in A_n. Then for such n,

$$C(\alpha \upharpoonright n|n) \leqslant \log |A_n| + \mathcal{O}(1) \leqslant \log(\frac{G_n - G_{n-1}}{2^{-n}}) + \mathcal{O}(1)$$

$$= n - g(n) + \mathcal{O}(1) \leqslant n - f(n).$$

□

Corollary 9.2.2 (Li-Vitanyi [185], after Martin-Löf [199])**.** *Let f be a total computable function such that $\sum_n 2^{-f(n)} = \infty$ and $C(n \mid n-f(n)) = O(1)$. Then for any real α we have $C(\alpha \upharpoonright n) \leqslant n - f(n)$ for infinitely many n.*

Proof. Let $p(n) = \tilde{C}(\alpha \upharpoonright n|n)$. Let f, g be as in Lemma 9.2.1. Let $q(n) = \tilde{C}(n|n-f(n))$. Remember, $|q(n)|$ is $\mathcal{O}(1)$. Note that $|1^{q(n)}0p(n)| \leqslant n-f(n)$ since $|p(n)| \leqslant n - g(n)$ and $g(n) - f(n) \to \infty$. Let

$$z(n) = 1^{q(n)}01^{n-f(n)-|1^q 0p(n)|-1}0p(n).$$

This has length $n - f(n)$ and is a description for $\alpha \upharpoonright n$. To compute $\alpha \upharpoonright n$, first compute $q(n)$. The length of $z(n)$ is $n - f(n)$. The assumption about $q(n)$ is that $q(n)$ can compute n from this. Then, given n we can compute $\alpha \upharpoonright n$ from this using $p(n)$. □

Corollary 9.2.3. *For any real α we have $C(\alpha \upharpoonright n) \leqslant n - \log n$ for infinitely many n.*

Proof. The function $n - logn \mapsto n$ is computable, so $C(n \mid n - (\log n)) = O(1)$. Furthermore, $\sum 2^{-\log n} = \infty$. Hence we can apply Corollary 9.2.2. □

There has been much work on analyzing C-complexity oscillations related to the above results. See Li-Vitanyi [185, page 138] for more on this topic.

9.2.2 K works

We can vindicate the intuition leading to the previous incorrect attempt at a definition of randomness for reals by using prefix-free complexity in place of plain complexity.

Definition 9.2.4 (Levin [179], Chaitin [42])**.** A real α is *Levin-Chaitin random* if $K(A \upharpoonright n) \geqslant n - \mathcal{O}(1)$.

Notice that this means that, in the nomenclature of Chapter 6, a real is Levin-Chaitin random iff all its initial segments are weakly K-random as strings. Since $K(xy) \leqslant K(x) + K(y) + O(1)$ for all x, y, it follows that if α is Levin-Chaitin random then given any initial segment xy of α, at least one of x or y is weakly Chaitin random. One cannot replace weakly random by strongly random as we will see below in Theorem 9.2.1.

We still do have any example. Here is one. The most famous of all random reals is due to Chaitin:

Theorem 9.2.5 (Chaitin [43], Chaitin's Ω)**.** *The halting probability* Ω *below is Levin-Chaitin random.*

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

Proof. Here is a straightforward proof based on the recursion theorem. As above, U is the universal prefix-free minimal Turing machine. We build a machine M and it has coding constant e given by the recursion theorem. (This means that if we put σ in $\mathrm{dom}(M)$, U *later* puts something of length $|\sigma| + e$ into $\mathrm{dom}(U)$.) At stage s, if we see Let $\Omega_s = \sum_{M(\sigma)\downarrow[s] \wedge |\sigma| \leqslant s} 2^{-|\sigma|}$. For $n \leqslant s$, if we see $K_s(\Omega_s \upharpoonright n) < n - e$, since $U_s : \sigma \mapsto K_s(\Omega_s \upharpoonright n)$, declare $M_s(\sigma)\downarrow$, which causes $\Omega_s \upharpoonright n \neq \Omega \upharpoonright n$. Note we cannot put more into the domain of M than U has in its domain. □

In the paper Chaitin [44], Chaitin defined another version of Ω. The proof above also shows that

$$\widehat{\Omega} = \sum_{\sigma \in \Sigma^*} 2^{-K(\sigma)}$$

is also 1-random. The only change necessary is that we define $M(\sigma) = \tau$ for some τ not in the range of U at stage s. We remark that we have assigned both the names of Levin and Chaitin to this notion of ramdomness. Levin (see Zvonkin and Levin [332] and Schnorr [265, 266]), used (versions of) monotone complexity to characterize the class of random reals in much the same way and the essential idea of prefix-free complexity is implicit in that paper. A similar method was devised by Schnorr [266], (and earlier, less correctly, Solomonoff [282]) using what he called *process complexity.* The history of this subject is quite involved, and we refer the reader to Li-Vitanyi [185] for detailed historical remarks.

Thus we have an example of a real that is random. In fact as we will soon see, almost every real is random (as we would expect). It was an interesting question of Solovay whether $\liminf_s K(\Omega \upharpoonright n) - n \to \infty$. This question was eventually solved by Chaitin [44]. However, recently Joseph Miller and Yu Liang (in [216]) gave a very powerful characterization of randomness which has this as a corollary. They proved that a real α is Levin-Chaitin random iff $\sum_{n \in \mathbb{N}} 2^{n - K(\alpha \upharpoonright n)} < \infty$. We will prove this result in Theorem 9.4.1.

Before we leave this section, we remark that it is clear that Ω is intimately related to the set

$$D_n = \{x : |x| \leqslant n \wedge U(x)\downarrow\}.$$

Recall we met this set in Section 6.12. Indeed in Theorem 6.12.4, Solovay proved that $K(D_n) = n + \mathcal{O}(1)$. We have the following basic relationships between D_n and $\Omega \upharpoonright n$, the first n bits of Ω.

Theorem 9.2.6 (Solovay [284]). *(i)* $K(D_n|\Omega \upharpoonright n) = \mathcal{O}(1)$. *(Indeed $D_n \leqslant_{wtt} \Omega \upharpoonright n$ via a weak truth table reduction with identity use.)*

(ii) $K(\Omega \upharpoonright n|D_{n+K(n)}) = \mathcal{O}(1)$.

Proof. (i) is easy. We simply wait till we have a stage s where $\Omega_s =_{\text{def}} \sum_{U(\sigma)\downarrow[s]} 2^{-|\sigma|}$ is correct on its first n bits. Then we can compute D_n.

The proof of (ii) is more involved. We follow Solovay [284]. Let $\widehat{D} = D_{n+K(n)}$. Note that $K(n+K(n)|\widehat{D}) = \mathcal{O}(1)$. We can simply compute from $\widehat{D}$, $K(j)$ for all $j \leqslant n+K(n)$, by looking for the length of the least $x \in D_n$ with $U(x) = j$. Hence we can find the least j such that $j+K(j) = n+K(n)$. Then we claim that $j - n = \mathcal{O}(1)$. To see this note that

$$K(j) - K(n)| \leqslant K(|j-n|) + \mathcal{O}(1).$$

Hence, $|j-n| \leqslant K(|j-n|)+\mathcal{O}(1) \leqslant 2\log|j-n|+\mathcal{O}(1)$. Therefore $|j-n| = \mathcal{O}(1)$. Also this means $K(j|\widehat{D}) = \mathcal{O}(1)$, and hence $K(n|\widehat{D}) = \mathcal{O}(1)$.

We prove that there is a q such that $K(\Omega \upharpoonright n-q|\widehat{D})\mathcal{O}(1)$. We construct a machine M that does the following. $M(xy)$ is defined if

(i) $U(x) = n$.

(ii) $|y| = n$.

(iii) $\Omega \geqslant \frac{y}{2^n}$.

Here of course we are interpreting $y \in \{0, \ldots, 2^n - 1\}$. Now we can find q such that, for all n,

$$|\Pi_M| + K(n-q) + n - q \leqslant n + K(n).$$

But then,

$$\Omega \geqslant \frac{y}{2^{n-q}} \text{ iff } \Pi_M(n-q)^* y \in D_{n+K(n)}.$$

Therefore $K(\Omega \upharpoonright n-q|\widehat{D}) = \mathcal{O}(1)$, since $K(n|\widehat{D}) = \mathcal{O}(1)$. Clearly, $K(\Omega \upharpoonright n|\Omega \upharpoonright n-q) = \mathcal{O}(1)$. Thus $K(\Omega \upharpoonright n|\widehat{D}) = \mathcal{O}(1)$, as claimed. □

9.3 The measure-theoretical paradigm

9.3.1 Ville's Theorem

The main idea is that a real would be random iff it had no rare properties.

The original person to try to formalize the notion of randomness was von Mises [317], who, in a remarkable paper, argued that a random real should be one that passed all statistical tests. For example if one thought of a real α as random if $\alpha(i)$ was the result of the i-th coin toss, then we would expect surely as many heads as tails. The law of large numbers would say that $\lim_{n\to\infty} \frac{(\alpha(1)+\cdots+\alpha(n))}{n} = \frac{1}{2}$. We could represent the set of reals satisfying the law of large numbers as L and would note that $\mu(\overline{L}) = 0$. Although von Mises lacked the terminology, his intuition seems to be that a real should be random iff it avoided all computable tests, or at least ones that were effectively given.

Von Mises idea of a statistical test was the folowing. We could imagine a set of allowable "selection functions" f, which choose bits of a real. We would consider $f : 2^{<\omega} \to \{\text{care, don't care}\}$. Thus for a real $q = q_1 q_2 \ldots$, we would look at the bits gotten by including q_{n+1} iff $f(q \upharpoonright n) = \text{care}$. If the real is random then it ought to pass the test, in that the number of zeroes chosen should be the same as the ones in the limit. We let $S(q \upharpoonright n) = \sum_{i \leqslant n} q_i$, and if the selected sequence is infinite, say, $j_1, j_2, \ldots$ have $f(q \upharpoonright j_k) = \text{care}$, we will let $S_f(q, n) = \sum_{k=1}^{n} q_{j_k+1}$. Von Mises idea was dealt a brutal (and maybe fatal[1]) blow by Ville [312] because of the following important theorem which says that *no* choice of "acceptable selection function" will work to give a true interpretation of what we would expect from random reals. The key problem is caused by (iii) since it makes q plainly non-random; saying as it does that the number of 1's does not exceed the number of 0's.

Theorem 9.3.1 (Ville [312])**.** *Take any countable collection E of selection functions. Then there is a real $q = q_1 q_2 \ldots$*

(i) $\lim_{n\to\infty} \frac{S(q\upharpoonright n)}{n} = \frac{1}{2}$.

(ii) For every f in E, if the subsequence of q that f cares about is infinite, then

$$\lim_{n\to\infty} \frac{S_f(q,n)}{n} = \frac{1}{2}.$$

(iii) For all n, $\frac{S(q\upharpoonright n)}{n} \leqslant \frac{1}{2}$.

Proof. The proof of this rather difficult result will be taken from Lieb, Osherson and Weinstein [188], which is a simplified version of the proof of Uspenskii, Semenov, and Shen [309].

This proof begins with a fnite version of Ville's Theorem as follows.

Lemma 9.3.2 (Finite version of Ville's Theorem)**.** *Let E be any finite collection of selection functions. Then there is a real q satisfying the conclusion of Theorem 9.3.1.*

Proof. (of the lemma) Without loss of generality, we will assume that the selection fnction h which always cares is amongst E, and hence we only need to meet (ii) and (iii). We will build q in stages. At each stage n we

[1] But see chapter 12 where von Mises ideas make a comeback.

will construct a subset $C(n)$ of E that cares about q_{n+1}. At stage 1, we will let $C(n) = \{f \in E : f(\lambda) = \text{care}\}$.
Stage $n+1$. Suppose that $q \upharpoonright n$ and $C(n)$ have already been defined. Set $C(n+1) = \{f \in E : f(q \upharpoonright n) = \text{care}\}$. Define

$$q_{n+1} = |\{j : j < n \wedge C(j) = C(j+1)\}| \mod 2.$$

That is, we will set the bit q_{n+1} to be 0 iff the subset of E that cares about position $n+1$ appears an even number of times. Evidently, q satisfies (iii) since each 1 appearing in q is preceded by an occurrence of 0 that can be uniquely chosen to match it.

As an illustration, let us suppose that $E = \{h, f_1, f_2\}$ with f_1 caring about positions 0,2,4,..., and f_2 about 0,3,6,.... (That is, the selection is oblivious and only cares about the position n itself, and not $q \upharpoonright n$.)

Then $C(1) = \{h, f_1, f_2\}$, and $q_1 = 0$, $C(2) = \{h\}$, $q_2 = 0$. $C(3) = \{h, f_1\}$, $q_3 = 0$. $C(4) = \{h, f_2\}$, $q_4 = 0$,. $C(5) = \{h, f_1\}$, $q_5 = 1$. (This is because $C(5) = C(3)$.) $C(6) = \{h\}$, $q_6 = 1$, as $C(2) = C(6)$. $C(7) = \{f, f_1, f_2\}$ and hence $q_7 = 1$, $C(8) = \{h\}$ $q_8 = 0$ giving $q \upharpoonright 7 = 00001110$.

Now suppose that $|\{n : f(q \upharpoonright n) = \text{care}\}| = \infty$. Let $n_1, n_2, \ldots$ denote an enumeration of those n with $f(q \upharpoonright n) = \text{care}$. Then $B = C(n_1), C(n_2), \ldots$ contains exactly those members of the sequence C that include f. The point is that for all m the value of q_{n_m} depends on B alone. Subsets of E that only occur finitely often in B eventually stop occurring altogether. Therefore the number of 0's and 1's in $q \upharpoonright n_m$ is ultimately determined by the subsets of E that occur infinitely often in B. This collection is infinite as B is infinite and there are nly finitely many distinct subsets of B cntaning f; meaning that one or more of them occurs infinitely often in B. For $k = 2^{|E|}$, no more than 2^k zeroes can occur consecutively in q since a block of zeroes requeres that different subsets of E care about each coordinate in the block. Consequently,

$$\lim_{m\to\infty} \frac{|\{j : q_{n_j} = 1 \wedge j \leqslant m\}|}{m} = \frac{1}{2}.$$

This gives (ii). □

Notice that the proof above actually shows that for every selectio function which has infinitely many n with $f(q \upharpoonright n) = \text{care}$,

$$0 \leqslant \frac{n}{2} - S_f(q, n) \leqslant 2^{|E|}.$$

Now the method above clearly fails when we atry to use it in the case that E is infinite. The problem is that (for example) each subset of E might occur only once and hence q might be the sequence of all 0's. he proof of Ville's Theorem in the case that E is infinite uses the construction of a *finite* subset $C(n)$ to determine q_n, but uses a combinatorial trick to decide which of the $f \in E$ are allowed to in $C(n)$. Roughly, f_k is allowed to occur with f_{k+m+1} only if it has occurred sufficiently often by itself, or with some of $f_1, \ldots, f_{k+m}$, as we see below.

Let $\mathcal{A}$ denote the class of infinite sequences of subsets of $\mathbb{N}$ containing 1. Thus $A \in \mathcal{A}$ implies $A(i) \subseteq \mathbb{N}$ and $1 \in A$. The technique is to define a special map $^* : \mathcal{A} \to \mathcal{A}$, as follows. For each $A \in \mathcal{A}$, each coordinate of A^* will be a nonempty finite subset of the corresponding coordinate in A.

Construction, Stage n. Suppose that we have already defined $A^*(m)$ and $I(m)$ for all $m < n$, so that $A^*(m) = \{j : j \in A(m) \text{ for } 1 \leqslant j \leqslant I(m)\}$. Then we define

$$I(n) = \min i(\exists j \in A(n) : |\{m : m < n \text{ such that } j \in A^*(m) \wedge I(m) = i\} \leqslant 3^i),$$

$$A^*(n) = \{j : j \in A(n) \wedge j \leqslant I(n)\}.$$

Notice that $I(1) = 1$ and $A^*(1) = \{1\}$. The construction of $A^*(n)$ depends only on $\{A(i) : i \leqslant n\}$. Furthermore, for all $i \in \mathbb{N}$, $I(n) = i$ for at most $i3^i$ many n. Now fix $\ell \in \mathbb{N}$ which occurs in infinitely many A. Let $\{n : \ell \in A(n)\}$ be enumertaed in increasing order, $n_1, n_2, \ldots$. Then since we know $I(n) = i$ for at most finitely many n, it follows that for cofinitely many m, $\ell \in A^*(n_m)$.

If we consider the sequence of integers $D = I(n_1)I(n_2)\ldots$, then by definition of I and A^*, for all $k \geqslant \ell$ there are at least 3^k many occurrences of k in the list D prior to the first occurrence of $k+1$ in D.

Now we define for $k \geqslant \ell$,

$$\alpha(k) = A^*(n_m), A^*(n_{m+1}), A^*(n_{m+2}), \ldots, A^*(n_{m+r}),$$

where $A^*(n_m)$ is the first ofccurrence of $k \in D$, and $A^*(n_{m+r+1})$ is the first of $k+1$. It follows that there is a $k \geqslant \ell$ and tail t of $A^*(n_1), A^*(n_2), \ldots$, such that

(a) t has the form $\alpha(k), \alpha(k+1), \alpha(k+2), \ldots$

(b) ℓ is a member of every coordinate of t.

In particular, k can be chosen to the the first occurrence of a number in D such that all later numbers occurring in D exceed ℓ.

that for cofinitely many members of $\{n : \ell \in A(n)\}$, $A^*(m)$ appears in t.

By definition of $\alpha(i)$, for all $i \geqslant k$, each of the sets in $\alpha(i)$ is a subset of $\{1, \ldots, i\}$, and hence there are at most 2^i of them. Therefore, since there are at least 3^k occurrences of k in D prior to the first occurrence of $k+1$, we see that

t has the form $\alpha(k), \alpha(k+1), \alpha(k+2)\ldots$, where for all $m \geqslant 0$, $\alpha(k+m)$ has length at least 3^{k+m} and contains at most 2^{k+m} distinct sets.

Now we will need a method of mapping A^* into a real q.

Definition 9.3.3. For $n \in \mathbb{N}$, the *preceding parity* of $A^*(n)$ in A^* is defined as

$$|\{j < n : A^(j) = A^*(n)\}| \mod 2.$$

Now we define q by letting $q(n)$ be the preceding parity of $A^*(n)$ in A^*. For $n \in \mathbb{N}$, let $B_0 = \{i \leqslant n : q(i) = 0\}$, and $B_1 = \{i \leqslant n : q(i) = 1\}$. As with the finite construction, we can pair each member of B_1 with a unique smaller member of B_0/ Thus we will have $\frac{S(q \restriction n)}{n} \leqslant \frac{1}{2}$ for all n.

Fix $\ell \in \mathbb{N}$ that occurs in infinitely many coordinates of A and, as before, let $\{n : \ell \in A(n)\}$ be enumerated as $n_1, n_2, \ldots$. Let $\widehat{q}$ denote $q(n_1), q(n_2), \ldots$. We wish to show that

$$\lim n \to \infty \frac{S(\widehat{q} \restriction n)}{n} = \frac{1}{2}.$$

It is enough to exhibit a tail s of $\widehat{q}$ with $\lim_{n\to\infty} \frac{S(s \restriction n)}{n} = \frac{1}{2}$. This tail s is specified by letting t be the tail of $A^*(n_1), \ldots$ of the form $\alpha(k), \alpha(k+1), \ldots$ where each $\alpha(k+m)$ has length at least 3^{k+m} and contains at most 2^{k+m} distinct sets specified above. Then specify $s(1) = \widehat{q}(n_m)$ iff $t(1) = A^*(n_m)$. That is, s excludes an initial segment of $\widehat{q}$ equal in length to the initial segmenr of $A^*(n_1), A^*(n_2), \ldots$ excluded by t. We show that this s works.

Recall that t is of the form $\alpha(k)\alpha(k+1)\ldots$ and for all i, $\ell \in t(i)$. Let $j \geqslant o$ be given thought of as both a coordinate in t and in s. We assume that j is sufficiently large that there is an $m(j)$ such that $t(j)$ is within $\alpha(k + m(j) + 1)$. Let $N_0(j)$ denote the number of 0's in $s \restriction j$ and $N_1(j)$ the number of 1's. Now since the block $\alpha(k+m)$ has at least 3^{k+m} many coordinates,

$$N_0(j) + N_1(j) \geqslant 3^{k+m(j)}.$$

There are at most 2^{k+i} distinct sets in $\alpha(k+i)$, and this number bounds the number of unmatched 0's. Therefore,

$$N_0(j) \leqslant N_1(j) + \sum_{i=0}^{m(j)+1} 2^{k+i} \leqslant N_1(j) + 2^{k+m(j)+2}.$$

Thus, $N_1(j) \geqslant \frac{1}{2}(N_0(j) + N_1(j) - 2^{K+m(j)+2})$. Let p denote the length of the "head" missing from s. Then $N_1(j) \leqslant N_0(j) + p$. This inequality implies

$$N_1(j) \leqslant \frac{1}{2}(N_0(j) + N_1(j) + p).$$

We finish by evaluating $R(j) = \frac{N_1(j)}{(N_0(j)+N_1(j))}$. Since we have neglected only finitely many terms (i.e. $R(j)$ for t with $t(j)$ a coordinate of $\alpha(k)$), it is clear that if $\lim_{j\to\infty} R(j) = \frac{1}{2}$ then $\lim_{n\to\infty} \frac{S(s\restriction n)}{n} = \frac{1}{2}$. We see that since $N_1(j) \leqslant \frac{1}{2}(N_0(j) + N_1(j) + p)$,

$$R(j) \leqslant \frac{N_0(j) + N_1(j) + p}{2(N_0(j) + N_1(j))}.$$

This has limit $\frac{1}{2}$. For the lower bound, we use the other estimate on $R(j)$, which gives,

$$R(j) \geqslant \frac{N_0(j) + N_1(j) - 2^{k+m(j)+2}}{2(N_0(j) + N_1(j))} = \frac{1}{2} - \frac{2^{k+m(j)+2}}{N_0(j) + N_1(j)}.$$

This converges to $\frac{1}{2}$ as $N_0(j) + N_1(j) \geqslant 3^{k+m(j)}$.

Now to complete the proof of Ville's Theorem, we may assume that E is enumerated without repititions as $h = f_1,\ f_2, f_3, \ldots$. We can concieve as the members of $A(i)$-the coordinates of the infinite sequences of subsets of $\mathbb{N}$ - as indices of the selction funbctions f in E.

In the previous construction, we were give an infinite sequence $A(1), A(2), \ldots$ of subsets of $\mathbb{N}$, and we reduced this to the construction based on an infinite sequence $A^*(1), A^*(2), \ldots$ of finite subsets of $\mathbb{N}$, and then defined a real q based on these $A^*(i)$.

Notice that the value of $q(n+1)$ depends only upon $A \restriction n = \{A(1), dots, A(n)\}$. For Ville's Theorem, we start with $A(1) = \{m \in \mathbb{N} : f_m(\lambda) = \text{care}\}$, and then produce $q(1)$ on the basis of $A^*(1)$. Next we define $A(2) = \{m \in \mathbb{N} : f_m(q(1)) = \text{care}\}$ producing $q(2)$ from $A^*(1)A^*(2)$, and so forth. The calculated bounds describe the number of 0's and 1's in the sequence q about which f_ℓ cares. That ends the proof. $\square$

Lieb, Osherson, and Weinstein Made the following observations about their proof above. Choose a selection function f_ℓ which cares aboute q infinitely often. Then define th fluctuation about the mean to be

$$\delta_\ell(n) = S_{f_\ell}(q, n) - \frac{1}{2}.$$

Then by the above we have seen that δ_ℓ is bounded by an ℓ-dependent constant. This property mimic the behaviour of the h function whose fluctuation is never positive. Furthermore, using th reasoning above, ther eis a number $C_\ell \geqslant 0$ such that $\delta_\ell(n) \geqslant -C_\ell n^{\frac{\log 2}{\log 3}}$. The 3 occurs because of the use of 3^i in the proof, and r could have been used instead of 3. The conclusion would be that for each $\varepsilon > 0$, there is a constant $C_\ell(\varepsilon) \geqslant 0$ such that for every n,

$$\delta_\ell(n) \geqslant -C_\ell(\varepsilon) n^{\varepsilon}.$$

Lieb, Osherson, and Weinstein [188] remark that this bound is remarkable. For a random coin toss, the law of iterated logarithm states that the fluctuations exceed $(1-\varepsilon')\frac{\sqrt{n \log\log n}}{\sqrt{2}}$ for any $\varepsilon' > 0$ infinitely often almost surely (Feller [104]). Lieb, Osherson, and Weinstein observe that for any slow growing function g, ther eis a suitably fast growing one p, so that using $p(i)$ in place of 3^i will enforce an analogous bound with $g(n)$ in place of n^{ε}, and a suitable constant $C_\ell(g)$. This would reduce the fluctuations even further.

9.3.2 Martin-Löf : a new start

Martin-Löf realized that von Mises ideas were really only looking at certain kinds of stochasticity. He suggested that we should try to encompass all types of (effective) statistical tests, not just those defined by selection. Martin-Löf [198] formalized this idea as follows. (In later chapters we will look at other ideas, such as those of Schnorr [264]. It is this version of randomness that we will look at in the present section.)

A collection of reals D that is effectively enumerated is a Σ^0_1 class. The main idea is that we wish a random real to pass all "tests" meaning that the real should not be in any any effectively presented null Σ^0_1 class of reals. Recall from Observation 5.16.3, that a Σ^0_1 class can be represented as $D = \cup\{[\sigma] : \sigma \in W\}$ for some c.e. W. Thus we define c.e. open set to be a c.e. collection of open rational intervals. Given the discussion above, the first guess one might make for a random real is that

"a real x is random iff for all computable collections of c.e. open sets $\{U_n : n \in \omega\}$, with $\mu(U_n) \to 0$, $x \notin \cap_n U_n$."

This is a very strong definition, and is stronger than the most commonly accepted version of randomness. Let's call this *strong randomness*[2]. The key is that we wish to avoid all "effectively null" sets. Surely an effectively null set would be one where the measures went to zero in some computable way. Such considerations lead Martin-Löf to the definition of randomness below.

Definition 9.3.4 (Martin-Löf, [198])**.** We say that a real is *Martin-Löf random* or 1- *random* iff for all computable collections of c.e. open sets $\{U_n : n \in \omega\}$, with $\mu(U_n) \leqslant 2^{-n}$, $x \notin \cap_n U_n$.

We call a computable collection of c.e. open sets a *test* since it corresponds to a statistical test as above, and ones with $\mu(U_n) \leqslant 2^{-n}$ for all n, a *Martin-Löf test*. The usual terminology is to say that a real is Martin-Löf random if it passes all Martin-Löf tests. Of course a real passes the test if it is not in the intersection.

We remark that while strong randomness clearly implies Martin-Löf randomness, the converse is not true. This is an observation of Solovay. Later we show that there are c.e. reals that are Martin-Löf random. Hence the inequivalence of strong randomness and Martin-Löf randomness will follow by showing that no strong random real is c.e.. The following proof of this observation is due to Martin (unpublished).

Let $\alpha = \lim_s q_s$ as usual, and define

$$U_n = \{y : \exists s \geqslant n[y \in (q_n, q_n + 2(q_s - q_n))]\}.$$

[2] This notion has been examined. It is equivalent to A is in every Σ^0_2 class of measure 1. Kurtz and Kautz call this notion *weakly* Σ^0_2*-random* It was also used by Gaifman and Snir [119]. The reader is referred to Li-Vitanyi, [185], p164, where they call it Π^0_2-randomness. We will look at this natural notion in Chapter 11

Then $\mu(U_n) \to 0$, yet $\alpha \in \cap_n U_n$. (Actually this shows that α cannot even be Δ_2^0.)

Martin-Löf tests are particularly interesting, and provide natural ways to generate random reals.

Theorem 9.3.5 (Martin-Löf [198])**.** *There exist* universal *Martin-Löf tests: That is there is a Martin-Löf test $\{U_n : n \in \mathbb{N}\}$ such that, for any Martin-Löf test $\{V_n : n \in \mathbb{N}\}$, $x \in \cap_{n\in\mathbb{N}} V_n$ implies $x \in \cap_{n\in\mathbb{N}} U_n$.*

Proof. Naturally, it is easy to enumerate all the Martin-Löf tests. One enumerates all c.e. tests, $\{W_{e,j,s} : e, j, s \in \mathbb{N}\}$ and stops the enumeration of one if the measure $\mu(W_{e,j,s})$ threatens to exceed $2^{-(j+1)}$ at any stage s of the simultaneous enumeration. Let

$$U_n = \cup_{e\in\mathbb{N}} W_{e,n+e+1}.$$

Note that $\mu(U_n) \leqslant \sum_e \mu(W_{e,n+e+1}) \leqslant \sum_e 2^{-(n+e+1)} \leqslant 2^{-n}$. Also, U_n is clearly computably enumerable, and hence $\{U_n : n \in \mathbb{N}\}$ is a Martin-Löf test. Finally, suppose that $x \in \cap_n W_{e,n}$. The $x \in \cap_{n\geqslant e+1} W_{e,n}$ and hence $x \in \cap_n U_n$ □

We will meet another construction of a universal Martin-Löf test due to Kučera in Chapter 11, Section 11.4. When combined with a kind of fixed point theorem, and an effective 0-1 law, Kučera's method allows for subtle coding arguments, as we see in that chapter.

Corollary 9.3.6 (Martin-Löf [198])**.** *There are 1-random reals. The set of such reals has measure 1.*

Proof. Let $\{U_n : n \in \mathbb{N}\}$, be a universal Martin-Löf test. Let $B = \{x \in \cap_n U_n\}$. Then by construction, $\mu(B) = 0$. □

We remark that it is easy to construct a Martin-Löf random real below $\emptyset'$ since one can use $\emptyset'$ as an oracle to force a string σ out of U_n and hence one can build a random real by the finite extension method. We have already constructed one explicit Martin-Löf random real, Chaitin's Ω. As we will see in later sections, there exist Martin-Löf random reals that have low Turing degree, using the Low Basis Theorem.

Solovay proposed the following variant of Martin-Löf randomness:

Definition 9.3.7 (Solovay [284])**.** We say that a real x is *Solovay random* iff for all computable collections of c.e. $\{U_n : n \in \omega\}$ such that $\Sigma_n \mu(U_n) < \infty$, x is in only finitely many U_i.

The reader should note the following alternative version of Definition 9.3.7.

A real is Solovay random iff for all computably enumerable collections of rational intervals $I_n : n \in \omega$, if $\Sigma_n |I_n| < \infty$, then $x \in I_n$ for at most finitely many n.

Again, we can define a Solovay test as a collection of rational intervals $\{I_i : i \in \omega\}$, with $\sum_i I_i < \infty$. Then a real is Solovay random iff it passes every Solovay test, meaning that it is in only finitely many I_i. Clearly if x is Solovay random, then it is Martin-Löf random. The converse also holds.

Theorem 9.3.8 (Solovay [284]). *A real x is Martin-Löf random iff x is Solovay random.*

Proof. Suppose that x is Martin-Löf random. Let $\{U_n\}$ be a computable collection of c.e. open sets with $\Sigma_n \mu(U_n) < \infty$. We can suppose, by leaving some out, that $\Sigma_n \mu(U_n) < 1$. Define a c.e. open set

$$V_k = \{y \in (0,1) : y \in U_n \text{ for at least } 2^k\ U_n\}.$$

Then $\mu(V_k) \leqslant 2^{-k}$ and hence as x is Martin-Löf random, $x \notin \cap_n V_n$, giving the result. $\square$

It is also true that Levin-Chaitin random is equivalent to Martin-Löf random.

Theorem 9.3.9 (Schnorr [264]). *A real x is Levin-Chaitin random iff it is Martin-Löf random.*

Proof. $(\rightarrow)$ Suppose that x is Martin-Löf random. Let

$$U_k = \{y : \exists n K(y \upharpoonright n) \leqslant n - k\}.$$

Since the universal machine is prefix-free, we can estimate the size of U_k.

$$\mu(U_k) = \sum \{2^{-|\sigma|} : K(\sigma) \leqslant n - k\}$$

$$\leqslant \sum_{n \in \mathbb{N}} 2^{-(n+k)} = 2^{-k}.$$

Hence the sets $\{U_k : k \in \omega\}$ form a Martin-Löf test, and if x is Martin-Löf random $x \notin \cap_n U_n$. Thus there is a k such that, for all n, $K(x \upharpoonright n) > n - k$. $\square$

The other direction of the proof is more difficult, and the most elegant proof known to the authors is the one of Chaitin [44]. As we discussed in Chapter 6, Chaitin's approach is more or less the same as that of Levin [179] and in Gács [114], where Levin used the notion of a discrete semimeasure.

The Chaitin approach is rather more abstract since it is *axiomatic* and (again) stresses the *minimality* of K as a measure of complexity. Recall from Chapter 6, Chaitin defined an *information content measure* as any function $\widehat{K}$ such that

$$\alpha_{\widehat{K}} = \Sigma_{\sigma \in 2^{<\omega}} 2^{-\widehat{K}(\sigma)} < 1, \text{ and,}$$

$$\{\langle \sigma, k \rangle : \widehat{K}(\sigma) \leqslant k\} \text{ is c.e..}$$

Recall that amongst information content measures, K was minimal in the sense that

$$K(\sigma) \leqslant K_k(\sigma) + \mathcal{O}(1).$$

Proof. (cont'd) This time suppose that x is not Martin-Löf random. We prove that x is not Chaitin random. Thus we have $\{U_n : n \in \mathbb{N}\}$ with $x \in \cap U_n$ and $\mu(U_n) \leqslant 2^{-n}$. We note that $\sum_n 2^{-n^2+n}$ converges, and indeed, $\sum_{n \geqslant 3} 2^{-n^2+n} < 1$. Notice that

$$\sum_{n \geqslant 3} \sum_{\sigma \in U_{n^2}} 2^{-(|\sigma|-n)} \leqslant \sum_{n \geqslant 3} 2^n \mu(U_{n^2}) \leqslant \sum_{n \geqslant 3} 2^{-n^2+n} < 1.$$

Thus by the minimality of K, $\sigma \in U_{n^2}$ and $n \geqslant 3$ implies that $K(\sigma) \leqslant |\sigma| - n + \mathcal{O}(1)$. Therefore, as $x \in \cap U_{n^2}$ for all $n \geqslant 3$ we see that $K(x \upharpoonright k) \leqslant k - n + \mathcal{O}(1)$, and hence it drops arbitrarily away from k. Hence, x is not Chaitin random. $\square$

If the reader wished to reinstate Kraft-Chaitin here, then the argument above is roughly the following. Since $x \in U_{n^2}$ (or any reasonable function of n, $2n$ would probably be enough), since the measure is small ($< 2^{-n^2}$), we can use Kraft-Chaitin to enumerate a machine which maps strings of length $k - n$ to initial segments of length k of strings in U_{n^2}. Specifically, as we see strings σ with $I(\sigma) \in U_{n^2}$ and length at least n^2, then we could enumerate a requirement $|\sigma| - k, \sigma$. (The total measure will be bounded by 1 and hence Kraft-Chaitin applies.) Notice that there is nothing special about n^2 here any computable function would do. That means that the complexity of a nonrandom real must dip arbitrarily low, infinitely often. That is we have the following.

Corollary 9.3.10. *Suppose that α is not random, and that f is any increasing computable function. Then there exist infinitely many k with $K(\alpha \upharpoonright k) < k - f(n)$.*

We are now able to prove a Theorem mentioned earlier that there are many strings that are (weakly) K-random yet highly not C-random.

Corollary 9.3.11. *There are infinitely many n and strings x of length n such that*

(i) $K(x) \geqslant n$ *and*

(ii) $C(x) \leqslant n - \log n$.

Proof. Let α be Martin-Löf random. Then by Schnorr's theorem, Theorem 9.3.9, for all n, $K(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$. On the other hand, by Corollary 9.2.3, for infinitely many n, $C(\alpha \upharpoonright n) \leqslant n - \log n$. $\square$

There are a number of variations of the above. In the next section we will look at sharpenings due to Miller and Yu. The following result due to

Merkle is similar in spirit and deals with blocks in the real Z rather than elements.

Theorem 9.3.12. *If* $Z = z_0 z_1 z_2 \dots$ *where* $K(z_i) \leqslant |z_i| - 1$ *for each* i, *then* $Z \notin \mathsf{MLRand}$.

Proof. Fix n and consider the machine M which, on an input σ, searches for an initial segment $\rho \subseteq \sigma$ such that $U(\rho)\!\downarrow = n$, and then for $\nu_0, \dots, \nu_{n-1} \in \mathrm{dom}(U)$ such that $\rho\nu_0 \dots \nu_{n-1} = \sigma$. Should the search be successful, M prints $U(\nu_0) \dots U(\nu_{n-1})$. Given a string $z_0 \dots z_{n-1}$, let σ be a concatenation of a shortest U-description of n followed by shortest U-descriptions of $z_0, \dots, z_{n-1}$. Then $M(\sigma) = z_0 \dots z_{n-1}$. Thus $K(z_0 \dots z_{n-1}) \leqslant K(n) + \sum_{i<n} K(z_i) \leqslant K(n) + |z_0 \dots z_{n-1}| - n + \mathcal{O}(1)$. Since $K(n) \leqslant 2\log n + \mathcal{O}(1)$, we obtain $K(z_0 \dots z_{n-1}) \leqslant |z_0 \dots z_{n-1}| - (n - 2\log n) + \mathcal{O}(1)$, and hence $K(Z \upharpoonright n)$ is bounded away from n, contradicting Schnorr's Theorem, Theorem 9.3.9. $\square$

9.4 Theorems of Miller and Yu

We are now ready to generalize Corollary 9.3.10 and to prove the Miller-Yu Theorem stated in the previous section.

Theorem 9.4.1 (Ample Excess Lemma, Miller and Yu [216])**.** *A real* α *is 1-random iff*

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

Proof. One direction is easy. Suppose that α is not 1-random. Then we know that for all c, for infinitely many n, $K(\alpha \upharpoonright n) < n - c$. That means that $\sum_{n \in \mathbb{N}} 2^{n - K(\alpha \upharpoonright n)} = \infty$.

Now for the nontrivial direction. For the other direction, note that, for any $m \in \mathbb{N}$,

$$\begin{aligned} \sum_{\sigma \in 2^m} \sum_{n \leqslant m} 2^{n - K(\sigma \upharpoonright n)} &= \sum_{\sigma \in 2^m} \sum_{\tau \prec \sigma} 2^{|\tau| - K(\tau)} \\ &= \sum_{\tau \in 2^{\leqslant m}} 2^{m - |\tau|} 2^{|\tau| - K(\tau)} = 2^m \sum_{\tau \in 2^{\leqslant m}} 2^{-K(\tau)} \leqslant 2^m, \end{aligned}$$

where the inequality is Kraft's. Therefore, for any $p \in \mathbb{N}$, there are at most $2^m/p$ strings $\sigma \in 2^m$ for which $\sum_{n \leqslant m} 2^{n - K(\sigma \upharpoonright n)} \geqslant p$. This implies that $\mu(\{\alpha \in 2^\omega : \sum_{n \leqslant m} 2^{n - K(\alpha \upharpoonright n)} \geqslant p\}) \leqslant 1/p$. Define $\mathcal{I}_p = \{\alpha \in 2^\omega : \sum_{n \in \mathbb{N}} 2^{n - K(\alpha \upharpoonright n)} \geqslant p\}$. We can express $\mathcal{I}_p$ as a nested union $\bigcup_{m \in \mathbb{N}} \{\alpha \in 2^\omega \mid \sum_{n \leqslant m} 2^{n - K(\alpha \upharpoonright n)} \geqslant p\}$. Each member of the nested union has measure at most $1/p$, so $\mu(\mathcal{I}_p) \leqslant 1/p$. Also note that $\mathcal{I}_p$ is a Σ^0_1 class. Therefore, $\mathcal{I} = \bigcap_{k \in \mathbb{N}} \mathcal{I}_{2^k}$ is a Martin-Löf test. Finally, note that $\alpha \in \mathcal{I}$ iff

$\sum_{n \in \mathbb{N}} 2^{n-K(\alpha \upharpoonright n)} = \infty$. Now assume that $\alpha \in 2^\omega$ is 1-random. Then $\alpha \notin \mathcal{I}$, because it misses all Martin-Löf tests, so $\sum_{n \in \mathbb{N}} 2^{n-K(\alpha \upharpoonright n)}$ is finite. $\square$

Corollary 9.4.2 (Miller and Yu [216]). *Suppose that f is an arbitrary function with $\sum_{m \in \mathbb{N}} 2^{-f(m)} = \infty$. Suppose that α is 1-random. Then there are infinitely many m with $K(\alpha \upharpoonright m) > m + f(m) - \mathcal{O}(1)$.*

Proof. Suppose that for all $m > n_0$, we have $K(\alpha \upharpoonright m) \leqslant m + f(m) - \mathcal{O}(1)$. Fix $m > n_0$. Then $n - K(\alpha \upharpoonright m) \geqslant m - (m - f(m) - \mathcal{O}(1)) = -f(m) + \mathcal{O}(1)$. Hence $\sum_{m \in \mathbb{N}} 2^{m-K(\alpha \upharpoonright m)} \geqslant \sum_{m \in \mathbb{N}} 2^{(-f(m)+\mathcal{O}(1))} = \infty$, a contradiction. $\square$

Thus we see that $K(\alpha \upharpoonright n) \geqslant n - k$ for some k and all n is *equivalent* to $\liminf_n (K(\alpha \upharpoonright n) - n) = \infty$.

One useful form of the Ample Excess Lemma is the following.

Corollary 9.4.3 (Miller and Yu [216]). *If A is 1-random, then*

$$(\forall n) K^A(n) \leqslant K(A \upharpoonright n) - n + O(1),$$

where K^A denotes prefix-free Kolmogorov complexity relative to A.

Proof. Consider

$$\{\langle n, c \rangle : K_s(A \upharpoonright n) - n + c < n\}.$$

This set is computably enumerable relative to A for any set A. By the Ample Excess Theorem,

$$\sum_{n,c} 2^{-K(A \upharpoonright n)+n-c} = 2 (\sum_n 2^{n-K(A \upharpoonright n)}).$$

Thus, using Kraft-Chaitin in relativized form,

$$(\forall n) K^A(n) \leqslant K(A \upharpoonright n) - n + O(1).$$

$\square$

9.5 Levin's and Schnorr's characterization and monotone complexities

The original machine characterization of random reals was due to Levin but involved what are called *monotone* machines, also related to Schnorr's process complexity from Schnorr [266], a complexity using a different kind of monotone machine. We remark that something akin to monotone machines were also introduced by Solomonoff [282].

Here, are viewing Cantor Space as a continuous sample space, and thinking of a real as a limit of computable reals, rather than a limit of strings. Levin's original idea here was to try to assign a complexity to the *real itself*. That is, think of the complexity of the real as the shortest machine that outputs the real. Hence now we are thinking of machines that take a program σ and might perhaps output a real α.

This is, or course, nonsensical unless we are dealing with computable reals. However, we can think of Kolomorov complexity as generated by such machines. The following definition can be applied to Turing machines with potentially infinite output, and to discrete ones mapping strings to strings. In this definition, we regard $M(\sigma)\downarrow$ to mean that at some stage s, $M(\sigma)\downarrow[s]$.

Definition 9.5.1 (Solomonoff [282], Levin [180])**.** We say that a machine M is *monotone* if its action is continuous. That is, for all $\sigma \preccurlyeq \tau$, if $M(\sigma)\downarrow$ and $M(\tau)\downarrow$ then

$$M(\sigma) \preccurlyeq M(\tau).$$

A simple but important fact is that we can enumerate all monotone machines, and hence we can use the standard method of constructing a universal one. Also discrete monotone machines are all monotone in the more general sense. Also notice that prefix-free machines are special cases of monotone machines. Using monotone machine we can similarly define three varieties of *monotone (Kolmogorov) complexity*. The basic idea behind all three definitions is the the complexity of a *real* α is coded by complexities of strings σ_n with $\sigma_n \prec \alpha$ and $\sigma_n \to \alpha$. This is in keeping with the idea that our space is 2^ω and not $2^{<\omega}$ and we are covering strings by reals.

An alternative characterization is provided by the following. The proof is immediate.

Lemma 9.5.2. *A monotone machine is equivalent to a computably enumerable collection $M = \{\langle \sigma_i, \tau_i\rangle : i \in \ltimes\}$ of pairs of strings such that if $\sigma_i \preccurlyeq \sigma_j$ then either $\tau_i \preccurlyeq \tau_j$ or $\tau_j \preccurlyeq \tau_i$.*

Definition 9.5.3 (Monotone Kolgomorov complexity)**.** Levin's (standard) monotone complexity Km is defined as follows. Fix a universal monotone machine U.

$$Km(\sigma) = \min\{|\tau| : \sigma \preccurlyeq U(\tau)\}.$$

We can also approximate the above using discrete machines. Let M denote a universal discrete machine.

Definition 9.5.4 (Discete monotone Kolmogorov complexity)**.** We define the following two notions of discrete Kolmogorov complexity.

$$Km_D(\sigma) = \min\{|\tau| : M(\tau) = \sigma\}.$$

$$Km_D^+(\sigma) = \min\{|\tau| : K(\tau) = \sigma\}.$$

$Km_D(\sigma)$ is due to Schnorr and seems neglected. He called it *process complexity*[3] and introduced it in [265], proving a characterization of Martin-Löf random reals based upon it in [266]. Neither of these notations are really standard, but will be useful for this chapter.

Lemma 9.5.5. *(i)* $Km(\sigma) \leqslant Km_D^+(\sigma) + \mathcal{O}(1) \leqslant Km_D(\sigma) + \mathcal{O}(1)$.

(ii) α *is computable iff* $Km(\sigma) = \mathcal{O}(1)$.

Theorem 9.5.6 (Levin [180, 332])**.** *A real is Martin-Löf random iff* $Km(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$ *for all* n.

Proof. Given a montone universal machine M, make a test by putting $[\sigma]$ into U_k iff $K_M(sigma) \leqslant |\sigma| - k$. Notice that we can do this in a prefix-free way since the machine is monotone. Then the same calculation works:

$$\mu(U_k) = \sum \{2^{-|\sigma|} : K_M(\sigma) \leqslant |\sigma| - k \wedge \forall \tau \prec \sigma (K_M \tau) > |\tau| - k)\} \leqslant 2^{-k}.$$

Now if α is Martin-Löf random then for some k, $\alpha \notin U_k$ giving the result that if α is Martin-Löf random then $Km(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$ for all n.

Conversely, if $Km(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$ for all n, then in particular $K(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$ for all n, since prefix-free machines are monotone ones. Then α is Martin-Löf random by Schnorr's Theorem. □

In some ways, the theory of monotone complexity is smoother than either plain or prefix-free complexity. Several other results go through.

Theorem 9.5.7. *Chaitin's characterization of the computable sets by plain complexity works for discrete monotone complexity. That is* α *is computable iff* $Km_D(\alpha \upharpoonright n) \leqslant Km(1^n) + \mathcal{O}(1)$ *for all* n.

Proof. We sketch the proof. Take a discrete monotone machine M which using length lexicographic ordering has $M(\sigma) = 1^{1\sigma}$ where we are interpreting 1σ as a binary number. This proves that $Km_D(1^n) \leqslant \log n + \mathcal{O}(1)$, and since monotone machines are also plain machines, we see that at C-random n, $Km(n) = \log n + \mathcal{O}(1)$. Now we simply follow Chaitin's proof of Theorem 6.4.2. □

Notice that the identity machine $I(\sigma) = \sigma$ is a monotone machine. Thus we see that for all σ, $Km_D(\sigma) \leqslant |\sigma| + \mathcal{O}(1)$. Thus we get the following:

---

[3] The notion was called process complexity by Schnorr since he regarded information as being in given with a direction. In Schnorr's words: "He who wants to read a book will not read it backwards, since the comments or facts given in its first part will help him to understand subsequent chapters (this means that they help him to find regularities in the rest of the book). Hence anyone who tries to detect regularities in a process (for instance an infinite squence or an extremely long finite sequence) proceeds in the direction of the process. Regularities that have ever been found in an inital segment of the process are regularities for ever. Our main argument is the the interpretation of a process (for example to measure the complexity) is a process itself that proceeds in the same direction." (Schnorr [266], top page 378.)

Corollary 9.5.8 (Levin [180, 332], Schnorr [266]). *It follows that α is Martin-Löf random iff for all n,*

$$Km_D(\alpha \upharpoonright n) = Km(\alpha \upharpoonright n) + \mathcal{O}(1) = Km_D^+(\alpha \upharpoonright n) + \mathcal{O}(1) = n + \mathcal{O}(1).$$

Since part of our theme is to try to classify the complexity of reals via their initial segment complexity, this shows that Km, and even Km_D, is too coarse to apply directly. More on this later.

We finish this section with some miscellaneous results relating Km and Km_D to C and K. We remark that the precise nature of how these measures interact is not really known. Some material can be found in Uspelsky and Shen [**?**], but there are no sharp estimates akin to those of Solovay's relating C to K.

Since $C(\sigma) \leqslant^+ Km_D(\sigma) \leqslant^+ K(\sigma)$ some results come for free. For instance, the estimation in Lemma 7.2.1, gives the following:

Lemma 9.5.9. *For any σ, and $\varepsilon > 0$,*

$$K(\sigma) \leqslant^+ Km_D(\sigma) + \log|\sigma| + \log\log(\|\sigma|) + \cdots + (1+\varepsilon)\log^k(|\sigma|).$$

Actually, Lemma 9.5.9 is true of Km.

Theorem 9.5.10 (Uspensky and Shen [310]). *For any σ, and $\varepsilon > 0$,*

$$K(\sigma) \leqslant^+ Km_D(\sigma) + \log|\sigma| + \log\log(\|\sigma|) + \cdots + (1+\varepsilon)\log^k(|\sigma|).$$

Proof. The proof uses a familiar device from the proof of Lemma 7.2.1. Let $\overline{|\sigma|}$ denote any reasonable prefix-free self-delimiting reprentation of $|\sigma|$. As in Lemma 7.2.1 if we were only after $K(\sigma) \leqslant \mathcal{O}(\log|\sigma|) + Km(\sigma)$, we could use $\overline{|\sigma|}$ as $a_1a_1\ldots a_na_n01$ where $|\sigma| = a_0a_1\ldots a_n$. As in Lemma 7.2.1, we have such representations of $|\sigma|$ of length $\log|\sigma| + \log\log|\sigma| + \cdots + (1+\varepsilon)\log^k|\sigma|$. Then consider the prefix-free machine V which, on input z, parses z as pq where p is some $\overline{|\sigma|}$ and V outputs τ if the universal monotone machine outputs some ν with $\tau \preccurlyeq \nu$ and $|\sigma| = |\tau|$. $\square$

It is also possible to show that Km is not subadditive.

Theorem 9.5.11 (Folklore). *For all c, there exists strings σ, τ with*

$$Km(\sigma\tau) \geqslant Km(\sigma) + Km(\tau) + c.$$

Proof. We can find sufficiently long n such that $Km(0^n1^n) >> e+f$ where $e = Km(1^d)$ and $f = Km(0^d$ for all d. $\square$

??PROCESS COMPLEXITY

However, it is possible to recover a mild kind of subadditivity.

Theorem 9.5.12 (Folklore). *For any σ, τ,*

$$Km(\sigma\tau) \leqslant^+ K(\sigma) + Km(\tau).$$

Proof. let V denote the universal monotone machine. Consider the machine M which, on input z, tries to parse this as $z = pq$ with $p \in \mathrm{dom}U$. When it does so, M then computes $U(p)V(q)$. Clearly this machine is monotone, and running that machine on input $\sigma_U^* \tau_V^*$ shows that $Km(\sigma\tau) \leqslant^+ K(\sigma) + Km(\tau)$. □

9.6 Kolmogorov random vs Levin-Chaitin random

9.6.1 Kolmogorov complexity and finite strings

Before we turn to examining the relationship between C- and K- complexity for infinite strings, we will first look at finite strings, and C-complexity. We have defined a strings σ to be random relative to C iff $C(\sigma) = n$. We might wonder if there is a test set characterization of this for statistical tests upon *finite* strings, and hence relative to the uniform distribution.

Definition 9.6.1 (Martin-Löf test for finite strings)**.** A uniformly computable collection of c.e. sets $\{V_k : k \in \mathbb{N}\}$ with $V_{k+1} \subseteq V_k$, is a called a Martin-Löf test for C-randomness for finite strings iff for all k, and for all n,

$$\sum\{2^{-n} : |\sigma| = n \wedge \sigma \in V_k\} \leqslant 2^{-k}.$$

The idea here is that the tests represent statistical tests with sensitivity 2^{-k} relative to the uniform distribution. The notion of C-incompressibility and that of randomness relative to all C-Martin-Löf tests does not coincide, but it very nearly does.

Theorem 9.6.2 (Martin-Löf [198])**.** *Define $\{V_k : k \in \mathbb{N}\}$ as follows:*

$$\sigma \in V_k \text{ iff } |\sigma| - C(\sigma||\sigma|) - 1 \geqslant k.$$

Then $\{V_k : k \in \mathbb{N}\}$ is a universal C-Martin-Löf test in the sense that for all C-Martin-Löf tests, $\{\widehat{V}_k : k \in \mathbb{N}\}$ there is a c such that for all ν $\nu \in \widehat{V}_k$ iff $\nu \in V_{k-c}$.

Proof. The number of σ with $C(\sigma||\sigma|) \leqslant |\sigma| - k - 1$ is bounded by the number of programs of length $\leqslant |\sigma| - k - 1$, and hence bounded by $2^{-(|\sigma|-k-1)}$. Thus the quantity $\sum\{2^{-n} : |\sigma| = n \wedge \sigma \in V_k\} \leqslant 2^{-k}$. Thus the V_k have the right size.

Now let $\{\widehat{V}_k : k \in \mathbb{N}\}$ be a given C-Martin-Löf test. We know that there are at most 2^{-k} many elements of length n in $\widehat{V}_k$, and hence we can, given n, specify such a string σ of length n in $\widehat{V}_k$ with a string of length k, saying which one it is in the standard enumeration of the set of such strings. Hence $C(\sigma||\sigma| = n) \leqslant n - k + c$, as required. □

Clearly we could also have done the above for prefix-free complexity.

9.6.2 Kolmogorov Randomness

The reader might wonder if there is a condition on the initial segment complexity of α in terms of the naively more natural C-complexity which guarantees Martin-Löf randomness. The answer is yes.

Theorem 9.6.3 (Martin-Löf [198, 199]).

(i) $\mu(\{\alpha : \exists c \exists^\infty n [C(\alpha \upharpoonright n) \geqslant n - c]) = 1\}$.

(ii) $\exists^\infty n [C(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)]$ *implies that* α *is Martin-Löf random. We call such* α *Kolmogorov random.*

(iii) The collection of reals which are Levin-Chaitin random but not Kolmogorov random has measure zero.

Martin-Löf observed that there is an equivalent formulation of the concept of Kolmogorov randomness in terms of relative initial segment complexity.

Theorem 9.6.4. $\exists^\infty n C(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1)$ *iff* $\exists^\infty n C(\alpha \upharpoonright n|n) \geqslant n - \mathcal{O}(1)$.

Proof. We prove $\rightarrow$ since the other direction is clear. Note that

$$C(\alpha \upharpoonright n) \leqslant C(\alpha \upharpoonright n|n) + 2|n - C(\alpha \upharpoonright n|n)| + \mathcal{O}(1).$$

This follows since the term $2|n - C(\alpha \upharpoonright n|n)|$ allows for a description of the n if we are given $C(\alpha \upharpoonright n|n)$ and hence we can recover $\alpha \upharpoonright n$ from these two parameters. We are assuming that $C(\alpha \upharpoonright n) \geqslant n - c$ for infinitely many n. Hence,

$$n - C(\alpha \upharpoonright n|n) \leqslant 2|n - C(\alpha \upharpoonright n|n)| + \mathcal{O}(1),$$

for infinitely many n. But then for these n,

$$n - C(\alpha \upharpoonright n|n) \leqslant \mathcal{O}(1).$$

□

Proof. (ii) To prove (ii), we will use Theorem 9.6.4 as a bridge to Theorem 9.6.2, and C-Martin-Löf tests. Let $\{V_k : k \in \mathbb{N}\}$ be the universal C-test of Theorem 9.6.2. Let $\{\widehat{V}_k : k \in \mathbb{N}\}$ be a universal Martin-Löf test for infinite strings. Since a Martin-Löf test for infinite strings can be considered a C-Martin-Löf test, we know that there is a constant c such that for all σ (by abuse of notation), $\sigma \in \widehat{V}_k$ iff $\sigma \in V_{k+c}$. Thus is $\alpha \notin \cap_k \widehat{V}_k$, then $\liminf_n \alpha \upharpoonright n(n - C(\alpha \upharpoonright n|n))$ is finite. Hence by Theorem 9.6.4, α is K-random. □

In Section 9.7.2, we will give a proof of a stronger result due to Miller, Nies Stephan, and Terwijn that if α is Kolmogorov random then α is 2-random.

9.7 Arithmetical randomness and strong randomness

Our proof of Theorem 9.6.3 (iii) works via another much stronger concept of randomness. Evidently, the notion of Martin-Löf randomness can be both relativized and generalized. For our purposes, we have the following noticed by many authors including Solovay, Kurtz, and Martin-Löf:

Definition 9.7.1. (i) A Σ_n^0 test is a computable collection $\{V_n : n \in \mathbb{N}\}$ of Σ_n^0 classes such that $\mu(V_k) \leqslant 2^{-k}$.

(ii) A real α is Σ_n^0-random or n-random iff it passes all Σ_n^0 tests.

(iii) One can similarly define Π_n^0, Δ_n^0 etc tests and randomness.

(iv) A real α is called *arithmetically random* iff for any n, α is n-random.

We remark that one can also relativize the notion of n-randomness to n-D-randomness by using Σ_n^D classes in place of Σ_n^0 classes.

9.7.1 Approximations and open sets

The reader might note the subtle difference between the notion of Σ_1^0 test and Martin-Löf test, especially in the analogs used in Definition 9.7.1. The former is defined in terms of *classes* of *reals* and the latter in terms of c.e. *open sets* of *strings*. By Observation 5.16.3, this is immaterial for $n = 1$ since every Σ_1^0 class D is equivalent to $\cup\{[\sigma] : \sigma \in W\}$ for some c.e. set W. However, consider the case $n = 2$. Consider the Σ_2^0 class consisting of reals that are always zero from some point onwards. This Σ_2^0 class is *not* equivalent to one of the form $\cup\{[\sigma] : \sigma \in W\}$ for some Σ_2^0 set W.

The use of open sets has been crucial in our arguments. Fortunately, this use of open sets can be resurrected in the $n > 1$ cases also, as we now see.

Lemma 9.7.2 (Kurtz [165])**.** *Let $q \in \mathbb{Q}$. The predicate*

$$\text{“}\mu(S) > q\text{''}$$

is uniformly Σ_n^0 where S is a Σ_n^0 class. The predicate

$$\text{“}\mu(S) < q\text{''}$$

is uniformly Σ_n^0 where S is a Π_n^0 class.

Proof. The proof works by way of induction upon n. For $n = 1$ we use Observation 5.16.3 that a Σ_1^0 class D is one that can be expressed of the form $D = \cup\{[\sigma] : \sigma \in W\}$ for some c.e. W. Then $\mu(D) > q$ iff $\exists s(\sum_{\sigma \in W_s} 2^{-|\sigma|} > q)$.

For the case $n + 1$, if S is Σ_{n+1}^0, then there is a uniform sequence of Σ_n^0 subsets $\{T_j : j \in \mathbb{N}\}$ such that $\overline{S} = \cap_{j \in \mathbb{N}} T_j$. Let $\widehat{T}_k = \cap_{j<k} T_j$. Then

$\mu(S) = 1 - \lim_{k\to\infty} \widehat{T}_k$. Thus $\mu(S) > q$ iff $\mu(\widehat{T}_k) < 1 - q$, this last predicate being Π^0_n uniformly in k and hence the result follows by induction. $\square$

The following technical result is central for almost all work on n-randomness for $n > 1$. In the case of n-Martin-Löf randomnes a summary version is provided by Corollary 9.7.4.

Theorem 9.7.3 (Kurtz [165], Kautz [140])**.** *Let $q \in \mathbb{Q}$.*

(i) For S a Σ^0_n class, we can uniformly computably compute the index of a $\Sigma^{\emptyset^{(n-1)}}_1$ class which is also an open Σ^0_n class $U \supseteq S$ and $\mu(U) - \mu(S) < q$.

(ii) For T a Π^0_n class T, we can uniformly computably compute the index of a $\Pi^{\emptyset^{(n-1)}}_1$ class which is also a closed Π^0_n class $V \subseteq T$ and $\mu(T) - \mu(V) < q$.

(iii) For each Σ^0_n class S we can uniformly in $\emptyset^{(n)}$ compute a closed Π^0_{n-1} class $V \subseteq S$ such that $\mu(S) - \mu(V) < q$. Moreover, if $\mu(S)$ is a real computable from $\emptyset^{(n-1)}$ then the index for V can be found computably from $\emptyset^{(n-1)}$.

(iv) For a Π^0_n class T we can computably from $\emptyset^{(n)}$ obtain an open Σ^0_{n-1} class $U \supseteq T$ such that $\mu(U) - \mu(T) < q$. Moreover, if $\mu(S)$ is a real computable from $\emptyset^{(n-1)}$ then the index for U can be found computably from $\emptyset^{(n-1)}$.

Proof. Again this works by induction. Let $n = 1$. Then (i) follows by taking $U = S$ and applying Observation 5.16.3. For (iii), again let $S = \cup\{[\sigma] : \sigma \in W\}$. Computably from $\emptyset'$, we can determine (if there is) a rational $\widehat{q}$ such that $\mu(S) > \widehat{q} > \mu(S) - q$. (This uses uses the uniformity of Lemma 9.7.2.) Again using $\emptyset'$ we can compute a stage t such that $\mu(S_t) =_{\text{def}} \sum_{\sigma\in W_t} 2^{-|\sigma|} > \widehat{q}$. Note that the set of strings in S_t is computable from $\emptyset'$, and is hence a $\Pi^{\emptyset'}_0$ class. For (ii) and (iv) we apply the same arguments to the complement of the given Π^0_1 class.

Now for the case $n + 1$.

(i) Let S be a Σ^0_{n+1} class. Then we can express S as a union of a computable sequence of Π^0_n classes $\{T_i : i \in \mathbb{N}\}$. By induction, for each i we can uniformly compute from $\emptyset^{(n)}$ a $\Sigma^{\emptyset^{(n-1)}}_1$ class $U_i \supseteq T_i$ such that

$$\mu(U_i) - \mu(T_i) \leqslant \frac{q}{2^{i+1}}.$$

Then let $U = \cup_{i\in\mathbb{N}} U_i$. Note that $S \subseteq U$, and hence $U - S = \cup_{i\in\mathbb{N}}(U_i - T_i)$. Thus,

$$\mu(U) = \mu(S) = \mu(U - S) = \mu(\cup_{i\in\mathbb{N}}(U_i - T_i)$$

$$\leqslant \sum_{i\in\mathbb{N}} \mu(U_i - T_i) \leqslant \sum_{i\in\mathbb{N}} \frac{q}{2^{i+1}} \leqslant q.$$

Since U is a union of $\Sigma_1^{\emptyset^{(n-1)}}$ classes whose indices are uniformly computable from $\emptyset^{(n)}$, we can regard U as a $\Sigma_1^{\emptyset^{(n)}}$ class, whose index is uniformly obtainable from the given information.

(iii) Let S be a Σ_{n+1}^0 class, so that $S = \cup\{T_i : i \in \mathbb{N}\}$, with $\{T_i : i \in \mathbb{N}\}$ a uniform sequence of Π_n^0 classes uniformly computed from $\emptyset^{(n)}$. Again, using $\emptyset^{(n+1)}$ this time, find a rational $\widehat{q}$ so that

$$\mu(S) > \widehat{q} > \mu(S) - \frac{q}{2}.$$

Then computably from $\emptyset^{(n)}$, we can find $j \in \mathbb{N}$ such that

$$\mu(\cup_{i \leqslant j} T_i) > \widehat{q}.$$

Now since $\cup_{i \leqslant j} T_i$ is also a Π_n^0 class, by the induction hypothesis, we can uniformly obtain the index of a $\Pi_1^{\emptyset^{(n-1)}}$ class $V \supseteq \cup_{i \leqslant j} T_i$ whose measure is within $\frac{q}{2}$, so that $\mu(S) - \mu(V) < q$. The procedure of obtaining V from S is uniform. The only place that the $\emptyset^{(n+1)}$ oracle is used is to find the number $\widehat{q}$ approximating the measure of S. Thus if $\mu(S)$ is computable from $\emptyset^{(n)}$, the index for V can be computed computably from $\emptyset^{(n)}$.

For (ii) and (iv), given a Π_n^0 class T we apply the arguments of (i) and (iii) to $\overline{T}$. □

We may define a Σ_n^0 *open* test as a uniformly computable collection of Σ_n^0 sets S_i (identified with)

$$V_i = \cup\{[\sigma] : \sigma \in S_i\},$$

so that $\mu(V_i) \leqslant 2^{-i}$. Now we can use *open* sets for our definition of Martin-Löf n-randomness. To wit, we have the following.

Corollary 9.7.4 (Kurtz [165])**.** *A real α is n-random iff for all Σ_n^0 open tests, $\alpha \notin \cap_{k \in \mathbb{N}} V_k$.*

Proof. Use Lemma 9.7.3 to approximate. □

Thus, $n+1$ randomness is the same as 1-randomness relative to $\emptyset^{(n)}$. This allows for relativization in many proofs.

9.7.2 Infinitely often maximally complex reals

The highest prefix-free complexity a string of length n string can have is $n + K(n) - \mathcal{O}(1)$. It is impossible for a real to have $K(\alpha \upharpoonright n) \geqslant n + K(n) - \mathcal{O}(1)$ for all n, since that would entail $C(\alpha \upharpoonright n) > n - \mathcal{O}(1)$ for all n, and we have seen that this is impossible. Hence, since every essentially strongly Chaitin random string is essentially Kolmogorov random, no real can always have maximal prefix-free complexity. However, reals can have maximal complexity *infinitely often*.

Definition 9.7.5. We say that a real α is *strongly Chaitin random* iff

$$\exists^{\infty} n(K(\alpha \restriction n) > n + K(n) - \mathcal{O}(1).$$

Theorem 9.7.6 (Solovay [284])**.** *Almost every real α has the property that is strongly Chaitin random.*

Indeed Yu, Ding and Downey showed that Theorem 9.7.6 is true of 3-randoms.

Lemma 9.7.7 (Yu, Ding, Downey [328], after Solovay [284])**.** *(i) Suppose that α is 3-random. Then*

$$\exists^{\infty} n(K(\alpha \restriction n) = n + K(n) + \mathcal{O}(1).$$

(ii) Suppose that α is 3-random. Then

$$\exists^{\infty} n(C(\alpha \restriction n) = n + \mathcal{O}(1).$$

Proof. Consider the test $V_c = \{\alpha : \exists m \forall n(n > m \to K(\alpha \restriction n) \leqslant n + K(n) - c\}$. Now $K \leqslant_T \emptyset'$, and hence V_c is $\Sigma_1^{\emptyset''}$, and hence Σ_3^0. Now we estimate the size of V_c. We show in fact $\mu(V_c) \leqslant \mathcal{O}(2^{-c})$. Let $U_{c,n} = \{x | \ (\forall m \geqslant n) K(y \restriction m) \leqslant m + K(n) - c\}$. It suffices to get an estimate $\mu(U_{c,n}) = \mathcal{O}(2^{-c})$ uniform in n since $V_c \subseteq \cup_{n \in \omega} U_{c,n}$. But $\mu(U_{c,n}) \leqslant 2^{-m} |\{\sigma : |\sigma| = m \ \ \& K(\sigma) \leqslant m + K(n) - c\}|$ for any $m > n$ and by Chaitin's Theorem 6.4.2, this last expression is $O(2^{-c})$.

We see that (ii) follows from (i) by Theorem 7.3.3, we proved that every (strongly) Chaitin-Schnorr random string is Kolmogorov random. $\square$

Proof. (of Theorem 9.7.6, concluded.) It is clear that the measure of the set of strongly Chaitin random reals has measure 1, since the collection of arithmetically random reals does. (There is a test computable from $\mathbf{0}^{(\omega)}$ which is universal for Σ_n^0 tests in the sense that any real passing such a test would be arithmetically random.) Since the collection of Martin-Löf random reals has measure 1, the result follows, by the Lemma above. $\square$

The reader should note that the above leaves open the possibility that the collections of Kolmogorov random reals strongly Chaitin random reals and Martin-Löf random reals all coincide. This is *not* true. No Δ_2^0 real can be Kolmogorov random. Notice that Ω is a Δ_2^0 real in the sense that its dyadic expansion can be computed by $\emptyset'$. The fact that Δ_2^0 reals have different initial segment complexity than other random reals was noted by Solovay [284], van Lambalgen [314], and others. The following proof was suggested by an observation of Fortnow.

Theorem 9.7.8 (Yu, Ding, Downey [328])**.** *No Δ_2^0 real is Kolmogorov random.*

Proof. Suppose that $\alpha \leqslant_T \emptyset'$. By the limit lemma, there is a computable function $f(n,s)$ such that $\alpha \restriction n = \lim_s f(n,s)$. Let g be any computable function, such as $g(n) = 2^{2^n}$, say. Let s_n be sufficiently large that $f(g(n), s)$

has reached its limit. Then for $k \geqslant n+s_n$, the following is a short C-program for $\alpha \upharpoonright k$:

The input is n, γ where γ is the part of α of lengths between $g(n)$ and k, which we write as $\gamma = \alpha[g(n), k]$. This input has length n plus $k - g(n)$. Then on this input, we first scan the length (say t) then calculate $f(g(n), t)$ and output $f(g(n), t)\gamma$, which will equal $\alpha \upharpoonright k$. The length of this program is bounded away from $k - c$ for any c. $\square$

Nies Stephan and Terwijn [232] observed that the reasoning above can be extended to prove the following.

Theorem 9.7.9 (Nies, Stephan and Terwijn [232]). *Suppose that α is Kolmogorov random. Then α is 2-random.*

Proof. Suppose that U is the usual universal prefix-free machine which is necessarily prefix-free relative to all oracles. Suppose that α is *not* 2-random. Let K' denote $K^{\emptyset'}$. Then for all c, $\exists^\infty n(K'(\alpha \upharpoonright n) < n - c$. Let σ denote the string witnessing this, so that $U^{\emptyset'}(\sigma) = \alpha \upharpoonright n$. Let s be sufficiently large that $U^{\emptyset'}(\sigma) \downarrow [s]$, with $\emptyset'$ correct use. Then consider the plain machine M which runs as follows. M looks at the input ν and attempts to parse it as $\sigma'\tau$, where it runs U with oracle $\emptyset'[t]$ for t steps, where $t = |\tau|$ steps, and for all such simulations, and σ if it gets a result it outputs $U^{\emptyset'}(\sigma')[t]\tau$. Then for inputs with $t > s$, we have $M(\sigma\tau) = \alpha \upharpoonright n+t$, and hence α is not Kolmogorov random. $\square$

This leaves open the question of whether 3-random is equivalent to being Kolmogorov random. Nies, Stephan, and Terwijn observed that if you put a time bound on the running time for measuring the plain Kolmogorov complexity, then Kolmogorov random with a computable time bound then it is equivalent to being 2-random.

Lemma 9.7.10 (Nies, Stephan and Terwijn [232]). *A real α is 2-random iff for all computable g, with $g(n) > n^2$, say, $\exists^\infty n(C^g(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1))$.*

Proof. Use the proof above, noting that the plain machine runs in ploynomial. For the other direction, we lose a quantifier because of the time bound g. $\square$

We remark that Lemma 9.7.10 is quite useful in analysing various aspect of 2-randomness. For instance, Nies, Stephan, and Terwijn used it to show that all 2-randoms are of hyperimmune degree. We give this proof in Chapter 11 after we prove the stronger result due to Martin that all weakly 2-randoms are of hyperimmune degree.

Thus there are distictly different notion of randomness. However, we will see in the next chapters that, for effective reals such as c.e. reals, in some sense the level of randomness of all c.e. random reals is the *same*.

In 2003, after hearing of Nies-Stephan-Terwijn result, Joe Miller was able to prove, remarkably, that 2-randomness and Kolmogorov randomness are the same. This was somewhat later also established independently by Nies, Stephan and Terwijn. We will follow the latter proof as it is a little simpler.

Theorem 9.7.11 (Miller [209], Nies, Stephan and Terwijn [232]). *A real α is 2-random iff α is Kolmogorov random.*

Proof. (Nies, Stephan, and Twerijn [232]) Our idea is to replace the computable g in the above by a total extension which has the appropriate complexity, so that C^g-random can be expressed in 2 quantifiers.

We say that $F : \Sigma^* \mapsto \Sigma^*$ is a compression function if for all x $|F(x)| \leqslant C(x)$ and F is 1-1.

Lemma 9.7.12 (Nies, Stephan, and Twerijn [232]). *There is a compression function F with $F' \leqslant_T \emptyset'$.*

Proof. Consider the Π^0_1 class of 1-1 functions $\widehat{F}$ with the property that $|\widehat{F}(\sigma)| \leqslant C(\sigma)$. Let F be a low path through this Π^0_1 class chosen via the Low Basis Theorem. Then F is a compression function. $\square$

The main idea is that most of the basic facts of plain complexity can be re-worked with any compression function. To complete the proof we use the following lemma. For a compression function F we can define F-Kolmogorov complexity C_F: α is F-Kolmogorov random iff $\exists^\infty n(C_F(\alpha \upharpoonright n) > n - \mathcal{O}(1))$.

Lemma 9.7.13 (Nies, Stephan, and Twerijn [232]). *If Z is 2-random relative a compression function F, then Z is Kolmogorov F-random.*

Proof. If α is not Kolmogorov random relative to F, we will make a F' Martin-Löf test $\{U_n : n \in \mathbb{N}\}$ covering α. Let

$$P_{b,t} = \{X : (\forall n \geqslant t)[C_F(X \upharpoonright n) < n - b]\}.$$

Then $\alpha \in \cap_b V_b$ where $V_b = \cup_t P_{b,t}$. Now $P_{b,t}$ is a Π^0_1 class relative to F, $\mu(P_{b,t}) \leqslant 2^{-b}$ as F is injective, and for each n there are fewer than 2^{n-b} strings σ of length n with $C_F(\sigma) < n - b$. This implies $\mu(V_{b,t}) \leqslant 2^{-b}$. If we set

$$R_{b,t,k} = \{X : \forall n(t \leqslant n \leqslant k \to C_F(X \upharpoonright n) < n - b)\},$$

and let $U_b = \cup_t R_{b,t,k}$, then the U_b are Σ^F_2, uniformly in b. Moreover, $V_b \subseteq U_b$, $\mu(U_b - V_b) \leqslant 2^{-b}$, and so $\mu(U_b) \leqslant 2.2^{-b}$. Thus the U_b constitute a F'-Martin-Löf test covering α. $\square$

Finally we get Miller's Theorem by applying Lemma 9.7.13 to the low compression function F obtained in Lemma 9.7.12. For such a low F, α is 2-random iff α is 2-F-random iff α is Kolmogorov-F-random only if α is Kolmogorov random. $\square$

There still remains a gap:*Is every Kolmogorov random real also strongly Chaitin random?* If not, does strong Chaitin randomness coincide with, say, 3-randomness? Quite recently, this problem was solved by Joe Miller:

Theorem 9.7.14 (Miller [**?**])**.** *A is 2-random real iff strongly Chaitin random.*

We will need to delay the proof of the proof of Theorem 9.7.14 until Chapter 15 as the proof uses concepts on computational lowness and triviality which have not yet been developed. We remark in passing that we don't know of natural characterizations of other higher levels of randomness in terms of initial segment complexity. For example, is it possible to give a reasonable characterization of, say, 3-randomness in terms of the behaviour of $K(\alpha \upharpoonright n)$?

9.7.3 Notes on 2-Randomness

9.8 Plain complexity and randomness

Though the characterizations of 1-randoms in terms of initial segment complexities such as monotone and refix-free are satisfying, there remains a longstanding question whether there was a *plain complexity* characterization of 1-randomness. Whilst we have seen that having infinitely often maximal C-complexity *sufficed* we have also seen that it was not *necessary.* In this section we will look at the recent work of Miller and Yu [216] who indeed gave a characterization of 1-randomness in terms of plain complexity, solving an question which had been open for nearly 40 years.

Definition 9.8.1 (Miller and Yu [216])**.** [4] Define a computable function $G\colon \omega \to \omega$ by

$$G(n) = \begin{cases} K_{s+1}(t), & \text{if } n = 2^{\langle s,t\rangle} \text{ and } K_{s+1}(t) \neq K_s(t) \\ n, & \text{otherwise.} \end{cases}$$

Note that $\sum_{n\in\omega} 2^{-G(n)} \leqslant \sum_{n\in\omega} 2^{-n} + \sum_{t\in\omega}\sum_{m\geqslant K(t)} 2^{-m} = 2 + 2\sum_{t\in\omega} 2^{-K(t)} < \infty$.

Theorem 9.8.2 (Miller and Yu [216], Gács [115] for (ii))**.** *For $x \in 2^\omega$, the following are equivalent:*

(i) x *is 1-random.*

(ii) $(\forall n)\ C(x \upharpoonright n) \geqslant n - K(n) - \mathcal{O}(1)$.

(iii) $(\forall n)\ C(x \upharpoonright n) \geqslant n - g(n) - \mathcal{O}(1)$, *for every computable $g\colon \omega \to \omega$ such that $\sum_{n\in\omega} 2^{-g(n)}$ is finite.*

[4] Also compare with Solovay's function of Theorem 6.8.1.

(iv) $(\forall n)\ C(x \upharpoonright n) \geqslant n - G(n) - \mathcal{O}(1)$.

We prove the most difficult implication first.

Lemma 9.8.3. *If* $(\forall n)\ C(x \upharpoonright n) \geqslant n - G(n) - \mathcal{O}(1)$*, then* $x \in 2^\omega$ *is 1-random.*

Proof. By Chaitin's Counting Theorem 6.7.4 (iii), there a c such that

$$|\{\tau \in 2^t \mid K(\tau) \leqslant t - k\}| \leqslant 2^{t-K(t)-k+c},$$

for all $t, k \in \omega$. We construct a partial computable (non-prefix-free) function $M\colon 2^{<\omega} \to 2^{<\omega}$. For $s, t \in \omega$, let $n = 2^{\langle s,t\rangle}$. To $\langle s,t\rangle$ we devote the M programs with lengths from $\frac{n}{2} + c + 1$ to $n + c$. Note that distinct pairs do not compete for elements in the domain of M. For $k \in \omega$, let $m = n - K_{s+1}(t) - k + c$. Clearly, $m \leqslant n + c$. If $m \geqslant \frac{n}{2} + c + 1$, then for every $\sigma \in 2^n$ such that $K(\sigma \upharpoonright t) \leqslant t - k$, try to give σ an M program of length m. Different k do not compete for programs, but it is still possible that there are not enough strings of length m for all such σ. However, this cannot happen if $K_s(t) = K(t)$. This is because the number of $\sigma \in 2^n$ for which $K(\sigma \upharpoonright t) \leqslant t - k$ is bounded above by $2^{t-K(t)-k+c}2^{n-t} = 2^{n-K(t)-k+c} = 2^m$, so there is enough room in the domain of M to handle every such σ. This completes the construction of M.

Assume that $x \in 2^\omega$ is not 1-random. For each $k \in \omega$, there is an $t \in \omega$ such that $K(x \upharpoonright t) \leqslant t - k$ and t is large enough that $K(t) \leqslant 2^{t-1} - k - 1$. Take the least $s \in \omega$ such that $K_{s+1}(t) = K(t)$ and let $n = 2^{\langle s,t\rangle}$. Then

$$m = n - K(t) - k + c \geqslant n - 2^{t-1} + k + 1 - k - c \geqslant \frac{n}{2} + c + 1,$$

because $n = 2^{\langle s,t\rangle} \geqslant 2^t$. This implies that there is an M program for $x \upharpoonright n$ of length $m = n - K(t) - k + c$. Also note that $G(n) = K_{s+1}(t) = K(t)$. So,

$$C(x \upharpoonright n) \leqslant C_M(x \upharpoonright n) + \mathcal{O}(1) \leqslant n - K(t) - k + c + \mathcal{O}(1) \leqslant n - G(n) - k + \mathcal{O}(1),$$

where the constant is independent of x, n and k. Because k is arbitrary,

$$\liminf_{n\to\infty} C(x \upharpoonright n) - n + G(n) = -\infty.$$

Therefore, if $(\forall n)\ C(x \upharpoonright n) \geqslant n - G(n) - \mathcal{O}(1)$, then x is 1-random. This completes the proof. $\square$

Proof of Theorem 9.8.2. (i) $\Longrightarrow$ (ii): Define

$$\mathcal{I}_k = \{x \in 2^\omega \mid (\exists n)\ C(x \upharpoonright n) < n - K(n) - k\}.$$

As usual, let K_s and C_s denote the approximations to K and C at stage s. Then $(\exists n)(\exists s)\ C_s(x \upharpoonright n) + K_s(n) < n - k$ iff $x \in \mathcal{I}_k$. Therefore, $\mathcal{I}_k$ is a Σ^0_1 class. Fewer than $2^{n-K(n)-k}$ $\mathcal{V}$-programs have length less than $n - K(n) - k$, so $|\{\sigma \in 2^n \mid C(\sigma) < n - K(n) - k\}| \leqslant 2^{n-K(n)-k}$. Therefore,

$$\mu(\mathcal{I}_k) \leqslant \sum_{n\in\mathrm{N}} \mu(\{x \mid C(x \upharpoonright n) < n - K(n) - k\})$$
$$\leqslant \sum_{n\in\mathrm{N}} 2^{-n}2^{n-K(n)-k} = 2^{-k} \sum_{n\in\mathrm{N}} 2^{-K(n)} \leqslant 2^{-k}.$$

So, $\{\mathcal{I}_k\}_{k\in\mathrm{N}}$ is a Martin-Löf test. If x is 1-random, then $x \notin \mathcal{I}_k$ for large enough k. In other words, $(\forall n)\ C(x \upharpoonright n) \geqslant n - K(n) - k$.

(ii) $\implies$ (iii): Let $g\colon \omega \to \omega$ be a computable function such that $\sum_{n\in\omega} 2^{-g(n)} < \infty$. By the Kraft–Chaitin Theorem, $(\forall n)\ K(n) \leqslant g(n) - \mathcal{O}(1)$. Therefore, if $(\forall n)\ C(x \upharpoonright n) \geqslant n - K(n) - \mathcal{O}(1)$, then $(\forall n)\ C(x \upharpoonright n) \geqslant n - g(n) - \mathcal{O}(1)$.

(iii) $\implies$ (iv) is immediate because G is computable and $\sum_{n\in\omega} 2^{-G(n)}$ is finite.

Finally, (iv) $\implies$ (i) was proved in Lemma 9.8.3. $\square$

Actually, Gá cs [115] proved something stronger than $(\forall n)\ C(x \upharpoonright n) \geqslant n - K(n) - \mathcal{O}(1)$ for 1-random x.

Theorem 9.8.4 (Gács [115])**.** *x is 1-random iff $(\forall n)\ C(x \upharpoonright n|n) \geqslant n - K(n) - \mathcal{O}(1)$.*

Proof. One direction is immediate from the Miller-Yu proof. The other needs essectially the same proof, save to use

$$\mathcal{I}'_k = \{x \in 2^\omega \mid (\exists n)\ C(x \upharpoonright n|n) < n - K(n) - k\}.$$

in place of $+I_k$ of the proof above (i) implies (ii). $\square$

9.9 Levin's Theorem, measures and degrees

9.9.1 Non-Lebesgue measures

The reader should note that all of the results of the book so far have involved (uniform) Lebesgue measure. They might well ask whether randomness is invariant under change of measure. This question will be treated in this section, culminating in Levin's basic theorem that up to degree you get the same randoms for any reasonable measure.

We remark that since this material is not central to the topics of the present book, we will not treat it further. However, we remark that such considerations are fully discussed in van Lambalgen thesis [314], and in Li-Vitanyi [185].

Finally we will use the material from this section to prove Demeth's remarkable theorem that 1-randomness is closed downwards in the nonzero Turing degrees of sets tt-below a 1-random.

Our approach will be to follow the account of Kautz [140] from his thesis. We will retain the notation of μ as being reserved for Lebesgue measure and use λ for other measures.

Definition 9.9.1 (Folklore-Levin and others)**.** A measure $\lambda : \mathcal{P}(2^\omega) \mapsto [0,1]$ is called *computable* iff the measures of basic clopen sets can be uniformly computably approximated: there is a computable function $\widehat{\lambda}$ such that for all σ and s,

$$|\widehat{\lambda}(\sigma, s) - \lambda([\sigma])| < 2^{-s}.$$

Two classical varieties of measure are the following. A measure λ is *atomic* if there is some $A \in 2^\omega$ such that $\lambda(\{A\}) \neq 0$. Non-atomic measures are sometimes called *continuous* measures, for obvious reasons. Of course, reasonable measures are nonatomic. The extreme atomic measure is the *trivial* one where $\sum_{A \in 2^\omega} \lambda(A) = 1$.

9.9.2 Representing reals

The key intuition is that the identification of 2^ω with $[0,1]$ actually depends on the measure; viz which subsets of 2^ω to identify with what intervals will depend on the choice of measure. In the usual representation, if a $\alpha \in 2^\omega$ begins with 10, the 1 corresponds to the fact that α is in $[\frac{1}{2}, 1]$ and hence corresponds to $\mu([1]) = \frac{1}{2}$, etc. That is, looking at the first n bits of a real α corresponds to knowing α to within 2^{-n}, by computing an interval of length 2^{-n} within which α must lie. Now suppose that λ is the measure corresponding to the distribution where 1's are twice as likely as 0's, so that $\lambda([1]) = \frac{2}{3}$ and $\lambda([0]) = \frac{1}{3}$. Then the first digit of α being 1 indicates that α represents a real in $[\frac{1}{3}, 1]$, etc.

Thus a string σ of length n interpreted with respect to λ determine a subinterval of $[0,1]$ *which we call* $(\sigma)_\lambda$.

This notation is taken from Kautz [140], but the idea is well-known from classical measure theory.

Definition 9.9.2. The interval determined by σ with respect to λ, $(\sigma)_\lambda = [l(\sigma), r(\sigma)]$, is defined as follows:

$$l(\sigma) = \sum_{|\sigma|=|\tau|, \tau <_L \sigma} \lambda([\tau]),$$

$$r(\sigma) = l(\sigma) + \lambda([\sigma]).$$

Notice that $\mu((\sigma)_\lambda) = \lambda([\sigma])$. Following Kautz [140], we will use hatted versions to denote the approximations.

Definition 9.9.3 (Kautz [140]). For a computable measure λ, and σ a string of length k, $(\sigma)_{\lambda,s}$ denotes the interval $[l(\sigma,s), r(\sigma,s)]$ where

$$l(\sigma,s) = \sum_{|\sigma|=|\tau|, \tau <_L \sigma} \widehat{\lambda}(\tau, s+k+1) - 2^{-(s+1)},$$

$$r(\sigma,s) = l(\sigma,s) + \lambda(\sigma, s+k+1) + 2^{-(s+1)}.$$

The following is a straightforward observation.

Lemma 9.9.4. *$(\sigma)_{\lambda,s} \supseteq (\sigma)_\lambda$, and $|(\sigma)_{\lambda,s} - (\sigma)_\lambda| \leqslant 2^{-s}$.*

Kautz used the following notation : if $\lambda(\{\alpha\} = 0$, let $\mathrm{real}_\lambda(\alpha) = \cap_s(\alpha \upharpoonright s)_\lambda$. In the case where λ is computable, note that

$$\mathrm{real}_\lambda(\alpha) = \cap_s(\alpha \upharpoonright s)_{\lambda,s}.$$

Finally, for a real α suppose that for each n, there is a unique string σ of length n with $\alpha \in (\sigma)_\lambda$. Then we will define the sequence A representing α in 2^ω as the unique A such that

$$\sigma \preccurlyeq A \text{ iff } \alpha \in (\sigma)_\lambda.$$

This is denoted by $\mathrm{seq}_\lambda(\alpha)$. For the standard Lebesgue measure, if α is not a dyadic rational, then $\mathrm{seq}_\mu(\alpha)$ is defined. In particular, if α is noncomputable, $\mathrm{seq}(\alpha)$ exists.

The following Lemma is explicit in Kautz [140], but is also implicit in Levin's work, such as in Zvonkin and Levin [332].

Lemma 9.9.5 (Kautz [140], Levin [332]). *Suppose that λ is computable and $\mathrm{seq}_\lambda(\alpha)$ is defined. Then*

(i) $\mathrm{seq}_\mu(\alpha) \leqslant_T \mathrm{seq}_\lambda(\alpha)$.

(ii) If $\lim_s \lambda([\alpha \upharpoonright n]) = 0$, then $\mathrm{seq}_\lambda(\alpha) \leqslant_T \alpha$.

In fact as we see in the proof below, the above are *wtt*-procedures, not just T-procedures.

Proof. (i) For each n, we need to compute a string σ_n of length n such that $\alpha \in (\sigma_n)_\lambda$. Given $\sigma_n \prec \mathrm{seq}_\lambda(\alpha)$, for $i \in \{0,1\}$, compute approximations $(\sigma_n i)_{\lambda,s}$ until we see $\alpha \notin (\sigma_n\langle 1-i)_{\lambda,s}$, when we know that $\alpha \in (\sigma_n i)_\lambda$. Note that this is actually a wtt-procedure from $\mathrm{seq}_\mu(\alpha)$.

(ii) Use essentially the same argument as in (i), using the fact that hypothesis implies that $\cap_s(\alpha \upharpoonright s)_\lambda$ contains a unique real. ☐

The following lemma says that even in the case of atomic measures, noncomputable reals are still well-behaved.

Lemma 9.9.6 (Kautz [140]). *If α is noncomputable, and λ is a computable measure, then $\lim_{s\to\infty} \lambda([\alpha \upharpoonright s]) = 0$. Hence all atoms are computable.*

Proof. If $\lim_{s\to\infty} \lambda([\alpha \upharpoonright s]) \neq 0$, then the interval $I = \cap_s(\alpha \upharpoonright s)_\lambda$ has positive measure. Thus $\mu\{\beta \in [0,1] : \text{seq}_\lambda(\beta) = \alpha\} = \mu(I) > 0$. By the previous lemma, from any noncomputable member of I we can compute α, it follows that $\mu\{(\beta : \alpha \leqslant_T \beta\}) > 0$. But then by Sacks theorem, $\alpha \equiv_T \emptyset$. $\square$

Note that is α is computable, then so are $\text{seq}_\lambda(\alpha)$ and $\text{seq}_\mu(\alpha)$. However, if λ is atomic and α is noncomputable it is still possible for $A = \text{seq}_\lambda(\alpha)$ to be an atom of λ and hence computable. But if λ is continuous, then $\lim_{s\to\infty} \lambda([A \upharpoonright s]) = 0$. This means by the above that $\text{seq}_\mu(\alpha) \leqslant_{wtt} \text{seq}_\lambda(\alpha)$. Consequently we have the following.

Corollary 9.9.7 (Kautz [140], Levin [332])**.** *Let λ be a computable measure and $\alpha \in 2^\omega$. Suppose that seq(α) is defined.*

(i) $\text{seq}_\lambda(\alpha) \leqslant_{wtt} \text{seq}_\mu(\alpha)$.

(ii) $\text{seq}_\mu(\alpha) \leqslant_{wtt} \text{seq}_\lambda(\alpha)$ if either α is computable or $\text{seq}_\lambda(\alpha)$ is noncomputable.

- *If λ is continuous, then $\alpha \equiv_{wtt} \text{seq}_\lambda(\alpha) \equiv_{wtt} \text{seq}_\mu(\alpha)$.*

Here is the main result. It says that using Lebesgue measure defines the same class of randoms *up to Turing degree.* We state this using the methods of Kautz [140].

Theorem 9.9.8 (Levin [332], also Kautz [140])**.** *Let λ be a computable measure. Then*

(i) if α is 1-random, $\text{seq}_\lambda(\alpha)$ is 1-random with respect to λ,

(ii) if $\text{seq}_\lambda(\alpha)$ is 1-random with respect to λ and noncomputable, then α is 1-random,

(iii) if λ is a continuous measure and **a** *is a nonzero degree, then* **a** *contains a 1-random real iff* **a** *contains a λ-1-random real.*

Proof. (Kautz [140]) This is not difficult, but involves chasing through the definitions. Suppose that $U_i = \cup\{[\sigma] : \sigma \in V_i\}$ is a Martin-Löf test with respect to λ covering $\text{seq}_\lambda(\alpha)$. We will define a ($\mu$-) Martin-Löf test $\widehat{U}_i$. For σ, the k-th string in V_i, compute $[p,q] = (\sigma)_{\lambda,k+i+1}$. Compute strings $\{\tau_0, \ldots, \tau_n\}$ such that $[p,q] = \cup_{\{0\leqslant j\leqslant n\}}[\tau_j]$. Put the $[\tau_j]$ into $\widehat{U}_i$. Then note that if $\sigma \preccurlyeq \text{seq}_\lambda(\alpha)$, then $\alpha \in (\sigma)_\lambda \subseteq [p,q]$, and hence, $\alpha \in \widehat{U}_i$, which has measure at most twice that of U_i. Thus if $\text{seq}_\lambda(\alpha)$ is not 1-λ-random it it not 1-random.

(ii) Now suppose that $U_i = \cup\{[\sigma] : \sigma \in V_i\}$ is a Martin-Löf test covering α. One can assume that U_i is a disjoint set of strings. This time let σ denote the k-th string eunumerated into U_i, and let $[p,q] = [\sigma]$. Put $I_k = [p - 2^{-(i+k+2)}, q + 2^{-(i+k+2)}]$. We may then define

$$\widehat{U}_{i,k} = \{\tau : \exists n((\tau)_{\lambda,n} \subseteq I_k\},\ \text{and,}$$

let $\widehat{U}_i = \cup_k \widehat{U}_{i,k}$. Now we observe that since $\lambda([\tau]) = \mu((\tau)_\lambda)$, $\lambda([\tau]) \leqslant \mu(I_k)$. Hence we see

$$\lambda(\widehat{U}_i) \leqslant \sum_k \mu(I_k) \leqslant 2.2^{-i}.$$

Thus to complete the proof, we need only show that $\mathrm{seq}_\lambda(\alpha) \in \widehat{U}_i$. As pointed out by Kautz [140], p91, the only real difficulty is now in the case that λ is atomic. Then it might not be possible to cover $[p, q]$ with intervals of the form $(\tau)_\lambda \in I_k$. (Because if λ assigns positive measure to some singleton $\{\beta\}$, it might be that for all $\tau \prec \beta$, $(\tau)_\lambda$ includes points from both the inside and outside of I_k.) However, under the hypothesis that $\mathrm{seq}_\lambda(\alpha)$ is noncomputable, there is some interval $(\tau)_\lambda \subseteq I_k$, containing α, so that τ is enumerated into $\widehat{U}_i$, and hence $\mathrm{seq}_\lambda(\alpha) \in \widehat{U}_i$. To see that this is true, let $X = \mathrm{seq}_\lambda(\alpha)$. Note that $\alpha \neq p, q$. Let $d = \min\{|\alpha - p|, |\alpha - q|\}$. For each n, $X \upharpoonright n$ is the unique strings of length n with $\alpha \in (X \upharpoonright n)_\lambda$. Suppose that for all n, $(X \upharpoonright n)_\lambda \nsubseteq [p, q]$. Then, for all n, p or q is in $(X \upharpoonright n)_\lambda$ along with α, meaning that for all n, $(X \upharpoonright n)_\lambda$ has length greater than d. But then X is computable, and hence $\mathrm{seq}_\lambda(\alpha)$ is computable contrary to hypothesis.

(iii) is now immediate. □

9.9.3 Atomic measures

The results of the previous section show that for continuous computable measures, it does not matter what measure we use to define 1-randomness (or indeed n-randomness) at least in terms of degrees. The question of what degrees can contain randoms for nontrivial computable atomic measures seems open at present. All problems disappear when we get to 2-randoms. The proof below relies on a later result on effective 0-1 laws, but we put the material here as it seems appropriate.

Theorem 9.9.9 (Kautz [140]). *Suppose that λ, ν are nontrivial computable measures. Then if* **a** *is a degree containing a λ-2-random real then it contains a ν2-random real*

Proof. Assume that λ is a nontrivial computable measure. Hence $\{x : \mathrm{seq}_\lambda(x)$ exists and is computable $\}$ has measure less than 1. Thus
$$\{x : \mathrm{seq}_\lambda(x) \text{ exists and } \exists \varepsilon > 0 \forall n (\lambda(\mathrm{seq}_\lambda(x) \upharpoonright n) \geqslant \varepsilon)\}$$
has measure less than 1. Therefore the class

$$C = \{x :\ \mathrm{seq}_\lambda(x) \text{ exists and } \forall \varepsilon > 0 \exists n (\lambda(\mathrm{seq}_\lambda(x) \upharpoonright n) < \varepsilon)\}$$

has positive measure. Let $P_i = \overline{U_i}$, where $\{U_i : i \in \mathbb{N}\}$ is the nu-universal Martin-Löf test.. Choose k large enough that $\nu(P_k) \geqslant 1 - \frac{1}{2}\lambda(C)$. Now we know that P_k is a Π^0_1 class containing all ν-1-random reals, and is $\mathrm{seq}_\lambda(x)$ is 1-random then x is noncomputable and hence $\mathrm{seq}_\nu(x)$ exists. Consequently we can express $P_k \cap C$ as

$$P_k \cap \{ \operatorname{seq}_\lambda(x) : \forall \varepsilon > 0 \exists n (\nu(\operatorname{seq}(x) \upharpoonright n) < \varepsilon)\}.$$

Thuis is clearly a Π^0_2 class, and hence by Lemma 11.9.1, since it has positive measure, it will contain all representatives of all $2-\nu$-random reals. $\square$

The question for 1-randoms is open and might be difficult.

The following theorem of Kautz shows that strange things can happen.

Theorem 9.9.10 (Kautz [140]). *There is a nontrivial computable measure λ such that no Δ^0_2 set is 1-random with respect to λ.*

As we know, by the low basis theorem, there are 1-random reals of low degree with respect to the uniform measure and hence any continuous computable measure.

9.9.4 Making reals random

A natural question is to ask is whether, given a real x is a there a (possibly noncomputable) measure relative to which it is random. In the case were mass can be concentrated, the following result gives a complete characterization of noncomputability in terms of randomness.

Theorem 9.9.11 (Reimann and Slaman [246]). *Suppose that x is a noncomputable real. Then there is a measure λ such that x is random relative to λ.*

Proof. The proof uses ideas from Kučera's proof that every real is computable from a 1-random real. Thus we will delay the proof until Chapter 11, after the proof of Theorem 11.4.1. $\square$

9.9.5 Making reals random : continuous measures

A rather more subtle question asks what happens when the measure is asked to be continuous.

Theorem 9.9.12 (Reimann and Slaman [246]). *There is a noncomputable real which is not random relative to any continuous measure.*

More generally, Kjos-Hannsen and Montalbán [**?**] prove the following.

Theorem 9.9.13 (Kjos-Hannsen and Montalbán [**?**], see Reimann and Slaman [246]). *If $A \in 2^\omega$ is a countable Π^0_1 class, then no member of A is continuously random.*

Proof. Let $A \in P$, a countable Π^0_1 class. Let $P = \cap_s P_s$ be a computable approximation to P. Let λ be any continuous measure. Then as P is countable, $\lambda(P) = 0$. Computably from λ, we can compute a stage $t(s)$ where

$$\lambda(P_{t(s)}) \leqslant 2^{-s}.$$

Then the set $[P_{t(s)}]$ is a (finite) λ-Martin-Löf test[5] with $A \in \cap_s P_{t(s)}]$. $\square$

The collection NCR, of reals, not random relative to any continuous measure, is a countable set. The following is proven by writing out the definition of not being Martin-Löf random (in relativized form), and noting that continuity and covering is arithmetic.

Theorem 9.9.14 (Reimann and Slaman [246]). *NCR is* Π^1_1

9.10 Demuth's Theorem

The only c.e. degree that contains a 1-random set is $\mathbf{0}'$, so the class of noncomputable 1-random degrees is definitely not closed downwards. In Chapter 11 we will see a result of Kučera that shows that the class of 1-random degrees is also not closed upwards. The following result show that there is a sense in which this class *is* closed downwards, but with respect to truth table reducibility. The proof we give is due to Kautz [140].

Theorem 9.10.1 (Demuth [65]). *If A is 1-random and $B \leqslant_{\mathrm{tt}} A$ is not computable, then there is a 1-random set $\widehat{B} \equiv_{\mathrm{wtt}} B$.*

Proof. If $B \leqslant_{\mathrm{tt}} A$ then there is an e such that $\Phi_e^A = B$ and Φ_e^X is total for all oracles X. Define a measure λ by

$$\lambda([\sigma]) = \mu(\bigcup\{[\tau] : \forall n < |\sigma|\, \Phi_e^\tau(n)\downarrow = \sigma(n)\}.$$

Then B is covered by a a 1-λ-Martin-Löf test implies that A is covered by a Martin-Löf test by looking at the pre-images of initial segments of strings. Thus, B is 1-λ-random. Hence, $\widehat{B} = \mathrm{real}_\lambda(B)$ is defined, has the same *wtt*-degree as B as is 1-random. $\square$

Corollary 9.10.2 (Kautz [140]). *There is a 1-random degree* $\mathbf{a}$ *such that for all* $\mathbf{b} \leqslant \mathbf{a}$*, if* $\mathbf{b}$ *is noncomputable,* $\mathbf{b}$ *is 1-random.*

Proof. Consider the Π^0_1 classes $P_c = \{X : \forall n(K(X \upharpoonright n) > n - c\}$. For some c this is nonempty, and contains only 1-randoms. By the hyperimmune-free basis theorem, thee is a hyperimmune-free 1-random real A. Being hyperimmune-free means that for all $B \leqslant_T A$, $B \leqslant_{tt} A$. Now apply Demuth's Theorem. $\square$

Another nice corollary is the an easy proof of an extension the result of Calude and Nies [37] that Ω is not *tt*-complete.

Corollary 9.10.3. *Suppose that A is any wtt incomplete c.e. real. Then $A \not\leqslant_{tt} \Omega$.*

[5] Actually, this shows that A is not even Kurtz random relative to λ, where Kurtz randomness is a weak form of randomness (or genericity) discussed later in Chapter 10.

Proof. No wtt-incomplete degree of a c.e. real can contain a 1-random as we have seen by Theorem 11.2.2. □

10

The unpredictability paradigm and Schnorr's critique

10.1 The unpredictability paradigm

10.1.1 Martingales and Supermartingales

It would be fair to say that most people would identify the intuitive notion of random with the idea that coin tosses, decay of uranium etc are random events in the sense that they are *unpredictable*. Knowledge of the first n coin tosses is of no help for the $n+1$-st. In this section we formalize this idea. We perform the following mind game. Suppose that you are given a real α. Imagine that you had some *computable* betting strategy which worked on the bits of α. At each stage you get to try to predict the next bit of α, knowing the previous n bits. That is, at each stage given a working capital, you could choose not to gamble on the the $n+1$-st bit, or could gamble some of your capital on the next bit. Then if the real is random, we would argue that no computable betting strategy should be able to succeed. It is this intuition that is behind the next definition.

Definition 10.1.1 (Levy [181]). A *martingale* is a function $f : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$ such that for all σ,

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

We say that the martingale *succeeds* on a real α, if $\limsup_n f(\alpha \upharpoonright n) \to \infty$.

Variations of the idea above are the following, which will turn out useful later in this section.

Definition 10.1.2. (i) A *supermartingale* is a function $f : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$ such that for all σ,

$$f(\sigma) \geqslant \frac{f(\sigma 0) + f(\sigma 1)}{2}.$$

We say that the supermartingale *succeeds* on a real α, if $\limsup_n F(\alpha \upharpoonright n) \to \infty$.

(ii) Similarly we can define *submartinagle* and its success if we ask that

$$f(\sigma) \leqslant \frac{f(\sigma 0) + f(\sigma 1)}{2}.$$

The idea is that martingales capture betting strategies. They were introduced by Levy [181], and Ville [312] proved that null sets correspond to success sets for martingales. They were used extensively by Doob in the study of stochastic processes. We will work with a simple version of the idea suitable for our purposes. For example suppose that α had the property that every 10th bit was 1. Then, starting with $f(\lambda) = 1$ and keeping $f(\nu) = 1$, until bit 10 where we would have $f(\nu 0) = 0$ and $f(\nu 1) = 2$, etc, we could build a computable martingale which would succeed on α. Schnorr [264] effectivized this.

Definition 10.1.3. We will define a (super-, sub-)martingale f as being *effective* or *computably enumerable* if $f(\sigma)$ is a c.e. real, and at every stage we have effective approximations to f in the sense that $f(\sigma) = \lim_s f_s(\sigma)$, with $f_s(\sigma)$ a computable increasing sequence of rationals.

We remark that the reader might have expected that an effective martingale would be one with f a computable function rather than one with computable *approximations*. We will return to this very important point in soon.

Before we prove Schnorr's result that Martin-Löf randomness and effective martingale randomness are the same, we need a very important classical lemma:

Theorem 10.1.4 (Kolmogorov's inequality, see Ville [312])**.** *(i) Let f be a (super-) martingale. For any string σ and prefix-free set $X \subseteq \{x : \nu \preccurlyeq x\}$,*

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

(ii) Let $S^k(f) = \{\sigma : f(\sigma) \geqslant k\}$, then

$$\mu(S^k(f)) \leqslant f(\lambda) \frac{1}{k}.$$

Proof. By compactness, we need only prove (i) for finite X. Suppose the lemma for $|X| = n$, and let Y have $n + 1$ elements. Let σ have greatest length with $Y \subseteq \{x : w \preccurlyeq x\}$. Then $Y_i = \{x \in Y : wi \preccurlyeq x\}$ $(i \in \{0,1\})$, have size $\leqslant n$. Thus by hypothesis,

$$\sum_{x \in Y} 2^{|w|-|x|} f(x) = \frac{1}{2}(\sum_{i=0}^{1} \sum_{x \in Y_0} 2^{|wi|-|x|} f(x)).$$

$$= \frac{1}{2}(f(w0) + f(w1)) = f(w).$$

Because any ν with $Y \subseteq \{x : \nu \preccurlyeq x\}$, satisfies $\nu \preccurlyeq x$, and f is a martingale,

$$f(w) = 2^{|w|-|\nu|} f(\nu),$$

we get (i) by multiplying the equations by $2^{-|w|}$.

To get (ii), if $X \subseteq S^k(f)$, is prefix free and $\mu(X) = \mu(S^k(f))$, by (i),

$$k\mu(X) = k(\sum_{x \in X} 2^{-|x|} \leqslant \sum_{x \in X} 2^{-|x|} f(x) \leqslant f(\lambda).$$

□

Theorem 10.1.5 (Schnorr [264]). *A real α is Martin-Löf random iff α does not succeed on any effective (super-)martingale.*

Proof. We show that test sets and martingales are essentially the same. This effectivizes Ville's work. Firstly suppose that f is an effective (super-)martingale. Define open sets

$$V_n = \cup\{[\beta] : f(\beta) \geqslant 2^n\}.$$

Then V_n is clearly a c.e. open set. Furthermore, $\mu(V_n) \leqslant 2^{-n}$ by Kolmogorov's inequality. Thus $\{V_n : n \in \mathbb{N}\}$ is a Martin-Löf test. Moreover, $\alpha \in \cap_n V_n$ iff $\limsup_n f(\alpha \upharpoonright n) = \infty$, by construction. Hence α succeeds on a martingale f iff it fails the derived Martin-Löf test.

For the other direction, we show how to build a martingale from a Martin-Löf test. Let $\{U_n : n \in \mathbb{N}\}$ be a Martin-Löf test. We represent U_n by extensions of a prefix-free set of strings σ, and whenever such a σ is enumerated into $\cup_{n,s} U_n^s$, increase $F(\sigma)[s]$ by one. To maintain the martingale nature of F, we also increase F by 1 on all extensions of σ, and by 2^{-t} on the substring of σ of length $(|\sigma| - t)$.) □

Corollary 10.1.6 (Levin [178, 332], Schnorr [264]). *There is a universal effective martingale. That is there is an effective martingale f, such that for all martingales g, and reals α, g succeeds on α implies f succeeds on α.*

Proof. Apply the proof above to the universal Martin-Löf test. □

As we have seen in an earlier section, Section 6.5, not only is there a universal prefix-free machine, but in fact there is one where the K-complexity is *minimal* amongst prefix-free Kolmogorov complexity. Clearly, given a martingale f, if g is a constant multiple of f then f succeeds on α iff g

does. Thus martingales are, in some sense, really only specified up to constant multiple. The dual of a Martin-Löf test is a supermartingale. We can strengthen Corollary 10.1.6 for supermartingales.

Theorem 10.1.7 (Schnorr [264])**.** *There is a* multiplicatively optimal *supermartingale. That is there is an effective supermartingale f such that for all effective supermartingales g, there is a constant c such that, for all σ,*

$$cf(\sigma) \geqslant g(\sigma).$$

Proof. It is easy to construct a computable enumeration of all effective supermartingales, g_i for $i \in \mathbb{N}$. (Stop the enumeration when it threatens to fail the supermartingale condition.) Then we can define

$$f(\sigma) = \sum_{i \in \mathbb{N}} 2^{-i} g_i(\sigma).$$

□

10.1.2 Supermartingales and continuous semimeasures

Earlier, Levin [180] had constructed a universal continuous semi-measure which could be interpreted as a supermartingale result[1] The reader might recall from Chapter 6 a *discrete* semimeasure λ was one with nonnegative values and $\sum_{\sigma \in 2^{<\omega}} \lambda(\sigma) \leqslant 1$. A discrete semimeasure is not compatible with the usual continuous Lebesgue measure on Cantor space. The continuous analog is the following.

Definition 10.1.8. A *continuous semimeasure* is a function $\delta : [2^{<\omega}] \mapsto \mathbb{R}^+ \cup \{0\}$ satisfying

(i) $\delta([\lambda]) \leqslant 1$, and

(ii) $\delta([\sigma]) \geqslant \delta([\sigma 0]) + \delta([\sigma 1])$.

In this Chapter, we will henceforth regard "semimeasure" to mean continuous semimeasure, unless otherwise specified. Of course, there is a big difference between the discrete semimeasures of Chapter 9 and the continuous ones of the present chapter. For example, defining $m([0^k]) = 1$ for all k and $m([\sigma] = 0$ for all other $\sigma \in 2^{<\omega}$ will be a continuous semimeasure, but notice that $\sum_{\sigma \in 2^{<\omega}} m([\sigma]) = \infty$ whereas for discrete semimeasures n, we need that $\sum_{\sigma \in 2^{<\omega}} n([\sigma]) < \infty$.

In the words of Li and Vitanyi, a semimeasure is a "defective" measure, as it is only *subadditive*. Of course we can talk about effective semimeasures, as c.e. semimeasures (and hence "effective defective measures"!). Levin [180] directly constructed a multiplicatively optimal maximal c.e.

[1]History is always difficult in this area, when many people had similar but not congruent ideas.

semimeasure. Again, we could enumerate all c.e. continuous semimeasures, $\{[\delta_e] : e \in \omega\}$ and define

$$\delta([\sigma]) = \sum_e 2^{-(e+1)}\delta_e([\sigma]).$$

We remark that $\delta([\sigma])$ is sometimes called the *a priori* probability of $[\sigma]$ and the reader should see Li-Vitanyi [185] for a discussion of how this interprets Bayes rule. [**?**]

There is a natural correspondence between semimeasures and supermartingales. To wit, let F be a supermartingale. Define

$$\delta([\sigma]) = 2^{-|\sigma|}F(\sigma).$$

then δ is a semimeasure (and the process can be reversed). Then Schnorr's optimal supermartingale is equivalent to Levin's optimal semimeasure.

We can also associate a version of Kolmogorov complexity to a semimeasure.

$$KM(\sigma) = -\log\delta([\sigma]),$$

where δ is the optimal semimeasure. This notion is also called the *a priori entropy* of $[\sigma]$. Another way of getting this entropy is the following:

Lemma 10.1.9 (Uspensky [308])**.** *A priori entropy is minimal function m c.e. from above such that*

$$\sum_{[\sigma]\in M} 2^{-f([\sigma])} \leqslant 1,$$

for any prefix-free set M*.*

Proof. Notice that, if δ is a semimeasure, then defining $f([\sigma])$ as the least k such that

$$2^{-k} < \delta([\sigma])$$

satisfies this condition. Conversely, if f satisfies the condition of the lemma, then

$$\max \sum_{[\sigma]\in D} 2^{-f([\sigma])},$$

where the maximum is taken over all D finite prefix free with σ a prefix of each word in D, is a semimeasure, establishing a correspondence (to a factor of 2) between semimeasures and functions satisfying the hypotheses of the lemma. $\square$

Notice that

$$KM(\sigma) = -\log F(\sigma) + |\sigma| + \mathcal{O}(1),$$

where F is an optimal c.e. supermartingale. The notation KM is similar to the notation Km used for (continuous) monotone Kolmogorov complexity met in Chapter 9. There is no coincidence here. If α is random then the universal supermartingale is eventually constant on it and hence

$$KM(\alpha \upharpoonright n) = n + \mathcal{O}(1) = Km(\alpha \upharpoonright n).$$

(Indeed for random *strings* σ, $KM(\sigma) = Km(\sigma) + \mathcal{O}(1)$.) Furthermore, if α is computable, then the universal supermartingale is, up to a multiplicative constant, 2^n on initial segments of length n upon it. Hence KM like Km is constant on computable reals. We have the following observation.

Lemma 10.1.10 (Levin, see Uspensky, Semenov and Shen [309])**.** *For all σ, $KM(\sigma) \leqslant^+ Km(\sigma)$.*

Proof. Assume that we have an optimal monotone machine U_m. We define a semimeasure δ as follows. Suppose that $U_m(x) = y[s]$ for some x, y. In the obvious way, define $\delta([z]) = 2^{-|x|}[s]$ for all z with $z \preccurlyeq y$. Then as above, this defines a c.e. semimeasure and hence $Km(\sigma) \geqslant^+ Km(\sigma)$ for all σ. □

It was a longstanding open question whether KM and Km differed on strings. That is, whether a natural analog of the Coding Theorem, Theorem 6.9.2, held for continuous measures; namely the a priori probability of a string was the same as its descriptional complexity. Or, alternatively, the canonical way to construct c.e. semimeasures is exactly the method of the proof of Lemma 10.1.10. Levin [180] conjectured that indeed it did, that is, $Km(\sigma) = KM(\sigma) + \mathcal{O}(1)$ for all σ. This question was finally solved in 1983 by P. Gács.

Theorem 10.1.11 (Gács [116])**.** *(i) There exists a function f with $\lim_s f(s) = \infty$, such that for infinitely many σ,*

$$Km(\sigma) - KM(\sigma) \geqslant f(|\sigma|).$$

(ii) Indeed, we may choose f to be the inverse of Ackkermann's function.

The proof of Gács Theorem is long and hard. We refer the reader to [116].

An apparently remaining question is if the corresponding reducibilities are different. That is, if we define $\alpha \leqslant_{KM} \beta$ to mean that for all n $KM(\alpha \upharpoonright n) \leqslant KM(\beta \upharpoonright n) + \mathcal{O}(1)$, is $\leqslant_{KM}$ different from $\leqslant_{Km}$? One could argue that since KM and Km are really designed for *reals*, if the answer is affirmative *then* the Coding Theorem really does hold for continuous semimeasures on *reals*.

Using Gács Theorem above, this question was solved by Joe Miller with the following relatively easy observation.

Corollary 10.1.12 (Miller)**.** *$\leqslant_{KM}$ is different from $\leqslant_{Km}$*

Proof. We use Gács Theorem, Theorem 10.1.11 above, and a finite extension argument. We define a real $\alpha = \lim_s \alpha_s$ in stages, with α_s a finite string. To define α_0 choose some string with

$$Km(\alpha_0) - KM(\alpha_0) > 1.$$

Now the argument is easy. Choose some α_{s+1} extending α_s with

$$Km(\alpha_{s+1}) - KM(\alpha_s) > s + 1.$$

Now the only thing we need to convince ourselves is that we can actually do this. The key observation is the following. Consider, for instance, the situation after we have defined α_0. We need to seek some extension σ of α_0 with $Km(\sigma) - KM(\sigma) > 2$. Take any string $\tau \in 2^{<\omega}$ of length at least $|\alpha_0| + 1$. Consider the string $\widehat{\tau}$ in $[\alpha_0]$ obtained by replacing the initial segment of τ of length α_0 by α_0. Then $Km(\widehat{\tau}) = Km(\tau) + \mathcal{O}(1)$, and $KM(\widehat{\tau}) = KM(\widehat{\tau}) + \mathcal{O}(1)$. Thus since f is unbounded we must have an extension of α_0 with the desired properties. The result follows. □

For more on this we refer the reader to Gács [116].

10.1.3 Martingales vs supermartingales and optimality

We remark that Theorem 10.1.7 *fails* for martingales. We begin by showing that there is no computable enumeration of all effective martingales. (We have seen statements to the contrary in the literature. Also, we can find no proof of the following theorem but suspect that it might have been known.)

Theorem 10.1.13 (Downey, Griffiths and LaForte [74][2]). *There is no effective enumeration of all c.e. martingales.*

Proof. This is a straightforward diagonalisation argument. Suppose $\widehat{M_i}$, $i \in \omega$, is an effective enumeration of all c.e. martingales, with or without repetition. (We don't require all rational non-decreasing approximations to each M_i to appear in the list, just that the limits - the martingales themselves - all appear in the enumeration.) We can effectively eliminate all martingales that are the constant-zero function to produce an enumeration M_i, $i \in \omega$, of the not-everywhere-zero c.e. martingales. Simply list $\widehat{M_0}(\lambda)[0]$, then $\widehat{M_0}(\lambda)[s], ..., \widehat{M_s}(\lambda)[s]$ for increasing values of s and select the least i such that $\widehat{M_i}(\lambda)[s] > 0$, and i has not yet been chosen, to appear next in our new enumeration.

We now derive a contradiction by defining a nowhere-zero c.e. martingale N such that $(\forall i \in \omega)(\exists \sigma \in 2^{<\omega})|\sigma| = i$ and $N(\sigma) \neq M_i(\sigma)$. In fact N will be computable map from $2^{<\omega}$ to $\mathbb{Q}$. In the following σ^c represents the string formed from σ by changing only the last bit from 0 to 1 or vice versa. For

[2] Although we have seen statements to the contrary in the literature, and this was surely known before the Downey, Griffiths, LaForte paper, and seems implicit in Levin's work such as [178].

any string τ we set $\tau^- = \tau \upharpoonright (|\tau|-1)$, that is, the string formed by removing the last bit of τ.
Step 0: Find s such that $M_0(\lambda)[s] = q_0 > 0$ and set $N(\lambda) = \frac{1}{2}q_0$.
Step $n+1$: Find s such that for some string σ of length $n+1$ we see $M_{n+1}(\sigma) = q_{n+1} > 0$. Let $N(\sigma) = \min(N(\sigma^-), \frac{1}{2}q_{n+1})$. Set $N(\sigma^c) = 2N(\sigma^-) - N(\sigma)$, this value is strictly positive. Set $N(\tau) = N(\tau^-) > 0$ for all other strings τ of length $n+1$.

It is easy to see that N is a strictly positive c.e. martingale, but it is not equal to M_i for any i, giving the contradiction. □

Next we will show that there is no effective enumeration of all effective martingales, as opposed to super-martingales.

Theorem 10.1.14 (Levin [178], Schnorr [264], also Downey, Griffiths, LaForte [74])**.** *There is no multiplicatively optimal effective martingale.*

Proof. Suppose $F : 2^{<\omega} \to \mathbb{R}$ is an effective martingale. We build a single martingale G such that for all $i \in \omega$, $(\exists\sigma)\ F(\sigma) \not\geqslant \frac{1}{i}G(\sigma)$. So F, an arbitrary martingale, cannot be optimal.

At stage 0, we set $G(\lambda) = 1$ and $G(1^n) = 1$ for all $n \in \omega$, and also $G(1^n0) = 1$ for all n. The idea is that on some extension τ of 1^n0 we will ensure $F(\tau) \leqslant \frac{G(\tau)}{n+1}$.

At stage $s > 0$ we work on strategies for $n < s$. Fix $n < s$. The strategy to defeat F with $\frac{1}{n+1}$ depends on which of finitely many *states* the strategy lies in at stage s. Let $\sigma_0 = 1^n0$. Initially, when the strategy is in state 0, if $F(\sigma_0)[s] < \frac{1}{n+1}$, we define $G(\sigma_0 0^k) = 2^k$ for $k = 1{+}s{-}n$, and $G(\tau) = 0$ for all other extensions of σ_0 of length $s{+}1$. At the first stage s where $F(\sigma_0)[s] \geqslant 1/(n+1)$, we fix $k_0 = s{-}n$ and wait until $\sum \tau \in T_0 F(\tau)[s] \geqslant \frac{2^{k_0+1}}{n+1}$, where T_0 is the set of strings of length $n{+}k_0{+}2$ extending σ_0. This wait is finite since F is a martingale and $F(\sigma_0) \geqslant \frac{1}{n+1}$. At this stage $G(\sigma_0 0^{k_0}) = 2^{k_0}$. Choose σ_1 to be whichever of $\sigma_0 0^{k_0+1}$ and $\sigma_0 0^{k_0}1$ gives the smaller value on F at stage s. In other words, $F(\sigma_1)[s] \leqslant F(\sigma_1^c)[s]$, where τ^c is the string that results from switching the last bit of τ to the opposite value. Then set $G(\sigma_1) = 2^{k_0+1}$, and and let $G(\tau)[s] = 0$ for all other extensions τ of $\sigma_0 0^{k_0}$) of length $n{+}k_0{+}1$. Note that $\sum \tau \in T_0 F(\tau)[s] - F(\sigma_1)[s] \geqslant \frac{2^{k_0}}{n+1}$. Inasmuch as any $F(\tau)[t]$ can only grow as t increases, if $F(\sigma_1) > \frac{1}{n+1}G(\sigma_1) = \frac{2^{k_0+1}}{n+1}$, then $\sum \tau \in T_0 F(\tau) \geqslant 3 \cdot \frac{2^{k_0}}{n+1}\ F(\sigma_0 0^{k_0}) > \frac{1}{n+1}2^{k_0+1}$. This implies $F(\sigma_0) > \frac{3}{2}\frac{1}{n+1}$.

The strategy now enters state 1, and repeats the process, with extensions of σ_1 rather than extensions of σ_0. In general, at stage s in state m, if $F(\sigma_0) \leqslant (\frac{2m+1}{2})\frac{1}{n+1}$ we define $G(\sigma_m 0^k) = 2^{k_{m-1}+l+1}$ for $l \leqslant s{-}k_{m-1}$, and $G(\tau) = 0$ for all other previously undefined values on extensions of σ_0 of length $\leqslant s{+}1$. At the first stage s where $F(\sigma_0) > (\frac{2m+1}{2})\frac{1}{n+1}$, we let $k_m = s{-}n$ and wait until $\sum \tau \in T_m F(\tau)[t] \geqslant (\frac{2m+1}{2})\frac{2^{k_m+1}}{n+1}$, where T_m is the set of strings of length $n{+}k_m{+}2$ extending σ_0. As before, this wait is finite since F is a martingale and $F(\sigma_0) \geqslant (\frac{2m+1}{2})\frac{1}{n+1}$. At this stage $G(\sigma_m 0^{k_m}) = 2^{k_m}$. Choose σ_{m+1} to be whichever of $\sigma_m 0^{k_m+1}$ and $\sigma_m 0^{k_m} 1$ gives the smaller value on F at stage s. Then set $G(\sigma_{m+1}) = 2^{k_m+1}$, and and let $G(\tau)[s] = 0$ for all other extensions τ of $\sigma_m 0^{k_m}$) of length $n+k_m+1$. Note that $\sum \tau \in T_m F(\tau)[s] - F(\sigma_{m+1})[s] \geqslant \frac{1}{2}(\frac{2m+1}{2})\frac{2^{k_m+1}}{n+1}$. Hence, if $F(\sigma_{m+1}) > \frac{G(\sigma_{m+1})}{n+1} = \frac{2^{k_m+1}}{n+1}$, then

$$\sum \tau \in T_m F(\tau) \geqslant \frac{2^{k_m+1}}{n+1} + \frac{1}{2}(\frac{2m+1}{2})\frac{2^{k_m+1}}{n+1} = \frac{2(m+1)+1}{2}\frac{2^{k_m}}{n+1}.$$

Since $|\sigma_{m+1}| - |\sigma_0| = k_m$, this would imply $F(\sigma_0) > \frac{2(m+1)+1}{2}\frac{1}{n+1}$.

Since $F(\sigma_0)$ is finite, there must be some least m so that $F(\sigma) \leqslant \frac{2(m+1)+1}{2}\frac{1}{n+1}$. We have, therefore, by the above argument, $F(\sigma_{m+1}) \leqslant \frac{G(\sigma_{m+1})}{n+1}$, as required. □

10.2 Schnorr's critique

In [264], Schnorr analyzed the basic theorem characterizing Martin-Löf randomness in terms of martingales, Theorem 10.1.5. He argued that Theorem 10.1.5 demonstrated a clear *failure* of the intuition behind the notion of Martin-Löf randomness. He argued that randomness should be concerned with defeating *computable* strategies rather than computably enumerable strategies, since the latter are fundamentally asymmetric, in the same way that a computably enumerable set is semi-decidable rather than decidable. One can make a similar argument about Martin-Löf tests being effectively null (in the sense that we know how fast they converge to zero), but not effectively given, in the sense that the test sets V_n themselves are not computable, rather that they are c.e.. There is indeed some weight to these arguments[3].

Armed with this fundamental insight, following Schnorr [264], we will look at two natural notions of randomness, which refine the notion of Martin-Löf randomness. Both are natural, one being inspired by the measure-theoretical approach and one via martingales. The first of these is most naturally based upon test sets.

Definition 10.2.1 (Schnorr randomness, Schnorr [264]). (i) We say that a Martin-Löf test $\{V_n : n \in \mathbb{N}\}$ is a *Schnorr test* iff for all n,

$$\mu(V_n) = 2^{-n}.$$

[3]However, there remains an interesting open question, which entails a possible refutation of Schnorr's critique is we consider *nonmonotonic* martingales. This question is analyzed in the last Chapter, Chapter 12

(ii) We say that a real α is *Schnorr random* iff for all Schnorr tests, $\alpha \notin \cap_n V_n$.

We remark that the choice of 2^{-n} in the definition of Schnorr randomness is a convenience. We could have chosen and suitable computable real.

Lemma 10.2.2 (Schnorr [264]). *If $f, g : \omega \to \mathbb{R}$ are non-increasing computable functions*[4] *with limit 0 and V_n is a (Martin-Löf) test such that $(\forall n)\mu(V_n) = f(n)$, then from an index for V_n we can effectively find a test U_n such that:*

$$\cap_{n\in\omega} V_n = \cap_{n\in\omega} U_n \textit{ and } (\forall n)\mu(U_n) = g(n)$$

Proof. (sketch) The proof of this involves manipulation of the c. e. sets to show that the following three steps can be accomplished: (i) choose increasing sequences n_i, and m_i such that such that $f(n_i) > g(m_i) > f(n_{i+1})$, (ii) build U_{m_i} as a superset of $V_{n_{i+1}}$, by adding elements of V_{n_i} but without adding any element to the null set, and (iii) define sets of the appropriate size (measure) between U_{m_i} and $U_{m_{i+1}}$. The details are quite straightforward. □

We will use this lemma when it is convenient.

The second definition of randomness is based upon martingales.

Definition 10.2.3 (Schnorr [264]). (i) A martingale f is called computable iff $f : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$ is a computable function with $f(\sigma)$ (the index of functions representing the effective convergence of) a computable real. (That is, we will be given indices for a computable sequence of rationals $\{q_i : i \in \mathbb{N}\}$ so that $f(\sigma) = \lim_s q_s$ and $|f(\sigma) - q_s| < 2^{-s}$.)

(ii) A real α is called *computably random* iff for it succeeds on no computable martingale.

In the remainder of this Chapter, we will explore these two concepts.

10.3 Schnorr randomness

Observe that for a Schnorr test, $\{V_n : n \in \mathbb{N}\}$, one can effectively compute membership. To do this, given a string σ, to see if $[\sigma] \subseteq V_n$, we merely need to wait for V_n to enumerate $2^{-n-|\sigma|}$ of its measure.

While Schnorr's argument has weight, at the time, the Schnorr notion of randomness attracted less attention than Martin-Löf randomness. Part of this was because the primary workers in this area found that the Martin-Löf

[4]Here, we say that $h : \omega \to \mathbb{R}$ is a computable function iff there is a computable function $\widehat{h}(\cdot,\cdot) : \omega \times \omega \to \mathbb{Q}$ such that for all n, m, $|\widehat{h}(n,m) - h(n)| < 2^{-m}$.

notion was enough for many results. Another important reason, however, was that the notion of Schnorr randomness proved far harder to deal with than Martin-Löf randomness. For instance, consider the following.

Lemma 10.3.1 (Schnorr [264]). *There is no universal Schnorr test.*

Proof. This follows from the contradiction that a universal Schnorr test $\{S_n\}$, $n \in \omega$, would contain all computable reals in its null set, but on the other hand there is a computable real in the complement of every Schnorr test's null set. $\square$

Another reason the theory of Schnorr randomness remained relatively undeveloped, was that many of the basic questions were open. For instance, the cornerstone of Martin-Löf's version of randomness is that the three characterizations (i) Chaitin incompressibility (ii) tests and (iii) martingales, all coincide, modulo some criticisms. This means that we have a mathematically robust notion. As we see in the next section, it is possible to give a machine characterization of Schnorr randomness. As we see in Section 10.4, when we also consider computable randomness, Schnorr provided a martingale characterization of Schnorr randomness in terms of "strong" success of computable martingales.

10.3.1 A machine characterization

Up till very recently, there was no known machine characterization of Schnorr's notion of randomness, and this was seen as a significant obstacle to the notion's development. Downey and Griffiths [72] finally gave a machine based definition of Schnorr randomness, the existence of which has been a longstanding open question (See e.g. Ambos-Spies and Kučera [8], Ambos-Spies and Mayordomo [9], etc). These are in terms of a new class of machines miniaturizations of which are relevant to resource bounded complexity as we see in Ambos-Spies and Mayordomo [9] or Lutz [192, 193].

Definition 10.3.2 (Downey and Griffiths [72]). A prefix free machine M is called *computable* iff

$$\mu(M) =_{\text{def}} \mu(\text{dom}M) = \sum_{\sigma \in \text{dom}(M)} 2^{-|\sigma|}$$

is a computable real.

Theorem 10.3.3 (Downey and Griffiths [72]). *A real α is Schnorr random iff for all computable machines M, there is a constant c such that, for all n, $K_M(\alpha \upharpoonright n) \geqslant n - c$.*

Before turning to the proof of Theorem 10.3.3, the machine characterization of Schnorr randomness we show there is another characterization analogous to one provided by Solovay for Martin-Löf randomness. Recall that a real x is *Solovay random* iff for all computable collections of c.e. sets U_n, $n \in \omega$, such that $\sum_n \mu(U_n) < \infty$, x is in only finitely many U_i. In Chapter 9 we showed that a real is Solovay random iff it is Martin-Löf random. This notion, as with many connected with Martin-Löf randomness, can be directly related to Schnorr randomness if the right way to "increase the effectivity" can be found. (The following definition is equivalent to a definition in terms of Martingales mentioned in Wang [318].)

Definition 10.3.4 (Downey and Griffiths [72]). A *total Solovay test* is a computable collection of c.e. open sets V_i: $i \in \omega$, such that the sum $\sum_{i=0}^{\infty} \mu(V_i)$ is finite and a computable real. A real α passes a total Solovay test if $\alpha \in V_i$ for at most finitely many V_i.

Theorem 10.3.5 (Downey and Griffiths [72]). *y is Schnorr random iff y passes all total Solovay tests.*

Proof. ($\leftarrow$) Suppose y is not Schnorr random, so it fails some Schnorr test $\{U_n\}_{n\in\omega}$. The infinite sum of the measures of these sets is computable, and y is in infinitely many of them so fails the total Solovay test represented by U_n.

($\rightarrow$) Suppose y is Schnorr random. Let $\{U_n\}_{n\in\omega}$ be an arbitrary total Solovay test. We note that $f(n) = \mu(U_n)$ is a computable function, since each $\mu(U_n)$ is left computable and their sum is bounded and computable. Define a c.e. open set $V_k = \{y \in (0,1) : \ y \in U_n \text{ for at least } 2^k \text{ of the } U_n\}$.

Now $\mu(V_n) < 2^{-n}$ and furthermore $g(n) = \mu(V_n)$ is a computable function of n, as to determine $\mu(V_n)$ to within ε we enumerate U_0 till its measure is within $\varepsilon 2^{-2}$ of its final value, U_1 to within $\varepsilon 2^{-\frac{5}{2}}$, and U_n to within $\varepsilon 2^{-\frac{n-4}{2}}$, up to the point where $2^{-n'} < \varepsilon 2^{-\frac{n'-4}{2}}$ (we can ignore U_m for $m > n'$). Most (in the sense of at least 'final measure'$-\varepsilon$) of the elements of V_k are already in V_k defined as in terms of being in at least 2^k of these approximations to each U_n, even if $n' < 2^k$.

As y is Schnorr random, $y \notin \cap_n V_n$ so y is in only finitely many U_i, y passes the total Solovay test. □

Proof. (Of Theorem 10.3.3) (*Only If* direction) Suppose z is Schnorr random. For any f given by a computable machine, suppose for the sake of contradiction that $(n - K_f(z \restriction n))$ is unbounded as a function of n. Let $M = \sum_{x\in dom(f)} 2^{-|x|}$.

Define $U_k = \{x : \exists n K_f(x \restriction n) \leqslant n - k\}$. If $\mu(U_k) > \delta$ then there exists a prefix-free subset of $2^{<\omega}$, $\{x_1, ..., x_n\}$, such that $\sum_{j=1}^{n} 2^{-|x_j|} > \delta$ and $K_f(x_j) \leqslant |x_j| - k$ for all $j = 1, 2, ..., n$, in each case via a p_j with $f(p_j) = x_j$ and $|p_j| \leqslant |x_j| - k$. We notice that:

$$\sum_{j=1}^{n} 2^{-|p_j|} \geqslant 2^k \sum_{j=1}^{n} 2^{-|x_j|} > \delta 2^k$$

Now as $\delta 2^k < M$ we have $\delta < M2^{-k}$, and considering this for all $\delta > 0$ we have $\mu(U_k) \leqslant M2^{-k}$. Furthermore $\mu(U_k)$ is a computable function of k as to approximate $\mu(U_k)$ to within ε we need only enumerate the strings of the domain of f in order of increasing length $y_1, y_2, ..., y_t$ until $M - \sum_{j=1}^{t} 2^{-|y_j|} < \varepsilon 2^k$. We can then determine all possible p_j relevant to the definition of U_k, except some that may provide extra x_i with the sum of $2^{-|x_i|}$ less than ε. From some point on the U_k form a Schnorr test giving the contradiction, since $z \in \cap_{k\in\omega} U_k$. Hence $(n - K_f(z \upharpoonright n))$ must be bounded for any such f: $(\exists d) n - K_f(z \upharpoonright n) \leqslant d$, so $K_f(x \upharpoonright n) \geqslant n - d$.
(*If* direction) Suppose z is not Schnorr random. Let U_k be a Schnorr test such that $z \in \cap_k U_k$, $U_{k+1} \subset U_k$, and $\mu(U_k) = 2^{-k}$. Represent each U_k as a union of extensions $[\sigma_{k,i}]$ of a prefix-free set $\{\sigma_{k,i} \ : \ i \in \omega\}$, such that $g(\langle k, i\rangle) = \sigma_{k,i}$ is a computable function from ω to $2^{<\omega}$. Note that $\sum_{a\in\omega} 2^{-|g(a)|} = 2$.

Since $\mu(U_{2n+2}) = 2^{-2n-2}$ we have, for all $i \in \omega$, $|\sigma_{2n+2,i}| \geqslant (2n+2)$. Consider the collection of 'lengths' $(|\sigma_{2n+2,i}| - n)$ for $n, i \in \omega$.
$\sum_{n,i\in\omega} 2^{-(|\sigma_{2n+2,i}|-n)} = \sum_{n\in\omega}(2^n \sum_{i\in\omega} 2^{-|\sigma_{2n+2,i}|}) = \sum_n 2^n 2^{-2n-2} = \frac{1}{2}$.
We wish to map a string of length $|\sigma_{2n+2,i}| - n$ to the string $\sigma_{2n+2,i}$ for each $i \in \omega$, $n \in \omega$; and by Kraft-Chaitin there is a prefix-free machine M that does precisely this. The partial computable function f defined by M satisfies $\sum_{x\in dom(f)} 2^{-|x|} = \frac{1}{2}$. Since $\forall n \exists i \ z \in [\sigma_{2n+2,i}]$, we have $(n - K_f(z \upharpoonright n))$ is unbounded. □

10.3.2 Miscellaneous results on computable machines

Not a lot is known about these machines. Certain of the combinatorial facts concerning general prefix-free machines still hold. For instance, we have an upper bound on the complexity required of such machines:

Proposition 10.3.6 (Downey, Griffiths, LaForte [74]). *There is a computable machine, M, and a constant $c \in \omega$, such that for all finite strings σ, $K_M(x) \leqslant |\sigma| + 2\log(|\sigma|) + c$*

Proof. We describe the machine M and then check that its domain is a computable real. We should like M to be as close to the identity function as is possible for a prefix-free machine. M maps the string $l(\sigma)^\frown\langle\sigma\rangle$ to σ, where $l(\sigma)$ is a special prefix-free coding of $|\sigma|$, consisting of the binary representation of $|x|$ but with every bit repeated, and then the 'end indicator' bits 01. For example, if $\sigma = 1001$ then $|\sigma| = 4$ and $l(\sigma) = 11000001$. So $l(\sigma)\sigma = 110000011001$ and $M(l(\sigma)\sigma) = \sigma$. The length of $l(\sigma)$ is of the order $2\log(|\sigma|)$, so M maps a string of length order $|\sigma| + 2\log(|\sigma|)$ to σ, and its range is $2^{<\omega}$. The domain of M is prefix-free because the set $L = \{\, l(\sigma) \,:\, \sigma \in 2^{<\omega} \,\}$ is prefix-free.

This machine M gives the result, provided μ(M) is a computable real. Consider all strings σ with $|\sigma| = n$. The the domain of M contains all possible extensions of $l(\sigma)$ of length $|l(\sigma)| + n$, so their combined measure is $2^n 2^{-|l(\sigma)|+n} = 2^{-|l(\sigma)|}$. Hence $\mu(\mathrm{M}) = \sum_{\tau \in L} 2^{-|\tau|}$. There are two strings of length 4 in L (0001 and 1101). There are two strings of length 6 (110001 and 111101) and four strings of length 8. Generally there are 2^i strings of length $2(i+2)$ for each $i \geqslant 2$. Thus $\mu(\mathrm{M}) = 2^{-3} + 2.2^{-6} + 2^2.2^{-8} + 2^3.2^{-10} + ... = 2^{-3} + 2^{-5} + 2^{-6} + 2^{-7} + ... = 2^{-3} + 2^{-4} = \frac{3}{16}$. □

Notice that no universal prefix-free machine can be computable, since such a machine must have measure equal to a Martin-Löf random real, which must have degree $\mathbf{0}'$. Computable machines can be total, (or *onto*) however, in the sense that they can give every string as an output.

Observation 10.3.7. *Suppose that M is a computable machine. Then there is a computable machine $\widehat{M}$ such that,*

(i) $ra(\widehat{M}) = 2^{<\omega}$, *and*

(ii) for all σ, $K_M(\sigma) = \min\{|\sigma| + 2\log(|\sigma|), K_{\widehat{M}}(\sigma)\} + \mathcal{O}(1)$.

Proof. (sketch) This is a padding argument. Let N be the machine giving the $|\sigma| + 2\log(|\sigma|) + c$ bound from the previous argument. Our machine $\widehat{M}$ on input 0σ and on input 1σ, $\widehat{M}(1\sigma) = M(\sigma)$. Then since $\mu(\mathrm{dom}(\widehat{M}) = \frac{1}{2}(\mu(\mathrm{dom}(N)) + \mu(\mathrm{dom}(M)))$, $\widehat{M}$ is a computable machine, with the desired properties by construction. □

If we are only interested in least Kolmogorov complexity, then we can do even better than Observation 10.3.7 by making the machine total in both *range* and *domain*.

Proposition 10.3.8. *Suppose that M is a computable machine. Then there is a computable machine $\tilde{M}$ such that,*

(i) $ra(\tilde{M}) = 2^{<\omega}$, *and*

(ii) $\mu(dom(\tilde{M}) = 1$, *and*

(ii) for all σ, $K_M(\sigma) \leqslant \min\{|\sigma| + 2\log(|\sigma|), K_{\tilde{M}}(\sigma)\} + \mathcal{O}(1)$.

Proof. (sketch) This is a straightforward application of Kraft-Chaitin. Take the machine $\widehat{M}$ produced by Observation 10.3.7. Now we will turn it into a machine whose domain has measure 1. Since $\mathrm{dom}(\widehat{M})$ is a computable real, so is $1 - \mu(\mathrm{dom}(\widehat{M}))$. For each k, once we know $1 - \mu(\mathrm{dom}(\widehat{M})) > 2^{-k}$, enumerate a Kraft-Chaitin axiom saying $\langle 2^{-k}, \lambda \rangle$. There is a machine P realizing this, and it can be combined with $\widehat{M}$ to make $\tilde{M}$. □

We remark that another fact which goes through is the subadditivity of K.

Observation 10.3.9. *For all computable machines M_1 and M_2 there is a computable machine M_3 such that for all σ, τ,*

$$K_{M_3}(\sigma\tau) \leqslant K_{M_1}(\sigma) + K_{M_2}(\tau).$$

Proof. The usual proof for K works. $\square$

10.4 Computable randomness

In this section, we return to the Martingale version of randomness, and look at the situation where we demand that the martingale be a computable function $F : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$. Actually, we can replace the computable reals by $\mathbb{Q}$ as we now see.

Lemma 10.4.1 (Schnorr [264]). *For all computable martingales F there is a computable martingale $f : 2^{<\omega} \mapsto \mathbb{Q}^+ \cup \{0\}$ such that for all α, F succeeds on α iff f succeeds on α.*

Proof. Suppose that F is a computable martingale. Consider G defined via $G(\sigma) = F(\sigma) + 1$ for all $\sigma \in 2^{<\omega}$. We will show how to construct f with $F(\sigma) < f(\sigma) \leqslant G(\sigma)$. To define $f(\lambda)$, take $F(\sigma)$ and compute some rational $q = q(\lambda)$ with $F(\lambda) < q \leqslant G(\lambda)$. Assume that we have computed $f(\sigma)$ and $F(\sigma) < f(\sigma) \leqslant G(\sigma)$. It suffices to show that there exist rationals r_0 and r_1 such that

$$f(\sigma 0) < r_0 \leqslant G(\sigma 0),\ f(\sigma 1) < r_1 \leqslant \text{ and, } f(\sigma) = \frac{f(r_0) + f(r_1)}{2}.$$

This is an elementary fact in algebra. We can then compute such rationals and set $f(\sigma i) = r_i$ for $i \in \{0, 1\}$. The clearly f is a computable martingale, the range of f is $\mathbb{Q}^+$, and for all β, $F(\beta \restriction n) + 1 \geqslant f(\beta \restriction n) \geqslant F(\beta \restriction n)$. Hence f succeeds on α iff F does. $\square$

The proof of Theorem 10.1.13 shows that there is no enumeration of computable martingales.

Corollary 10.4.2 (Corollary to the proof of Theorem 10.1.13).

- *There is no effective enumeration of all computable martingales $2^{<\omega} \to \mathbb{R}$.*
- *There is no effective enumeration of computable martingales $2^{<\omega} \to \mathbb{Q}$ (here we're thinking of computable functions that map immediately to their value in $\mathbb{Q}$, not computable martingales into $\mathbb{R}$ that happen to have range a subset of $\mathbb{Q}$).*

Thus, as in the case of Schnorr randomness, we will need to deal with Π^0_2 approximations.

10.4.1 Schnorr randomness via computable randomness

We begin by examining how Schnorr, computable and Martin-Löf randomness relate. The following shows that it is possible to characterize Schnorr randomness in terms of computable randomness.

Definition 10.4.3. We say that a computable martingale *strongly* succeeds on a real x iff there is a computable unbounded nondecreasing function $h : \mathbb{N} \mapsto \mathbb{N}$ such that $F(x \upharpoonright n) \geqslant h(n)$ infinitely often. We say that F *h-succeeds* for the particular computable order h.

Theorem 10.4.4 (Schnorr [264])**.** *A real x is Schnorr random iff no computable martingale strongly succeeds on x.*

Proof. We show that if there is a Schnorr test x does not withstand then we can construct a martingale that strongly succeeds on x, and also that given such a martingale we can construct a Schnorr test x does not withstand.

Lemma 10.4.5. *Given a computable martingale F there is a computable martingale $f : 2^{<\omega} \mapsto \mathbb{Q}^+$ such that for all reals α, F strongly succeeds on α iff f does.*

Proof. (of Lemma) This follows directly from the proof of Lemma 10.4.1. $\square$

We return to the proof of Theorem 10.4.4.

Constructing a test from a martingale. Given a computable martingale F (into the rationals) that strongly succeeds on x, via unbounded computable h, we define a Schnorr test U_n by $U_n = \cup\{[\sigma] : F(\sigma) \geqslant 2^n$ and $F(\sigma) \geqslant h(|\sigma|)\}$. Without loss of generality, suppose $0 < F(\lambda) \leqslant 1$.

Suppose x satisfies $F(x \upharpoonright j) \geqslant h(j)$ infinitely often. Then, for all n, $\exists j\ F(x \upharpoonright j) \geqslant h(j) \geqslant 2^n$. So, for all n, $[\sigma] \subseteq U_n$ where $\sigma = (x \upharpoonright j)$. Then $F(x \upharpoonright j) \geqslant h(j)$ infinitely often implies $x \in \cap_n U_n$.

Furthermore, there is an algorithm to compute $\mu(U_n)$ to within 2^{-s} for any given s, and any n. Find l_s such that $h(l_s) \geqslant 2^s$. Compute $U_{n,s} = \{[\sigma];\ F(\sigma) \geqslant 2^n$ and $F(\sigma) \geqslant h(|\sigma|)$ and $|\sigma| \leqslant l_s\}$. Claim that $\mu(U_n) - \mu(U_{n,s}) \leqslant 2^{-s}$.

This claim is correct because the part of U_n missing from $U_{n,s}$ is represented by a prefix-free collection of strings σ_i which satisfy $F(\sigma_i) \geqslant h(|\sigma_i|) \geqslant h(l_s)$. By Kolmogorov's inequality for martingales, the measure of all such $[\sigma_i]$ is at most $F(\lambda)/h(l_s) \leqslant 1/(2^s)$.

Constructing martingale from Schnorr test This is the more difficult direction. Suppose $x \in \cap_n U_n$, i.e. x does not withstand the Schnorr test, and $\mu(U_n) = g(n)$. We wish to build a martingale F and unbounded h. Our basic idea for F is similar to the Martin-Löf case. Recall that there we represent U_n by extensions of a prefix-free set of strings σ, and whenever such a σ is enumerated into U_n (any n), increase $F(\sigma)$ by one. To maintain the martingale nature of F, we also increase F by 1 on all extensions of σ, and by 2^{-t} on the substring of σ of length $(|\sigma| - t)$.)

We must, in the case of Schnorr tests, modify the martingale so that we have a "constructively infinite" lim sup for x that do not withstand the test. We could try to compute how far along each path in the tree of strings it is necessary to go to exceed some threshold $n \in \omega$, but in fact our approach is to add more than one to $F(\sigma)$ and F on longer strings when σ enters a test set (and 2^{-t} multiples of this amount to substrings). If the amount we add is $f(|\sigma|)$ and f is monotone and unbounded, then it will provide $F(x \upharpoonright j) \geqslant f(j)$ infinitely often as required for success of the martingale in the Schnorr sense (constructive lim sup). The identity function is a first guess for f, but increases too rapidly, as some tests will lead to "c.e. martingales" whose approximations do not converge on some strings.

We consider the sets U_n as represented, for each n, by a prefix-free sequence of strings. We let B be an effective union of these sequences, it is no longer prefix-free and may in fact contain repetitions of some strings.

Our approach has three main steps:

- Find a computable, monotone, unbounded function f such that
$$\sum_{x \in B,\ |x| \geqslant n} 2^{-|x|} \leqslant 2^{-2f(n)}$$
- Find a computable h such that $\sum_{x \in B,\ |x| \geqslant h(n)} 2^{-|x|} 2^{f(|x|)} \leqslant 2^{-n}$
- Define a martingale F from a new Schnorr test $V^k = B \cap \{\sigma : |\sigma| \geqslant h(k)\}$ by adding $2^{f(|x|)}$ to $F(x)$ when x enters any V^k.

$$F(\sigma) = \sum_{k \in \omega} \Big(\sum_{\sigma y \in V^k} 2^{-|y|} 2^{f(|\sigma y|)} + \sum_{n < |\sigma|,\ (\sigma \upharpoonright n) \in V^k} 2^{f(n)} \Big)$$

□

Corollary 10.4.6 (Downey and Griffiths [72]). *Computably random c.e. reals have high degree.*

Proof. This follow from Theorem 13.13.5. □

We have the following:

Theorem 10.4.7. *(i)* (Schnorr [264]) *Martin-Löf randomness implies computable randomness implies Schnorr randomness*

(ii) (Schnorr [264]), Yongge Wang [318, 319] *None of these implications can be reversed.*

(iii) (Downey and Griffiths [72]) *They cannot be reversed even for c.e. reals.*

We do not prove Theorem 10.4.7 here since we will improve on it in the next section where we prove a stronger result due to Nies, Stephan and Terwijn [232], Theorem 10.4.8.

10.4.2 High degrees and separating notions of randomness

In the previous section, we mentioned that the concepts of Martin-Löf, computable, and Schnorr randomness all gave rise to distinct notions. For Schnorr randomness this was also known even for c.e. reals. (Downey, Griffiths and LaForte [74], effectivizing the arguments of Yongge Wang [318, 319].) We have also seen that all computably random c.e. reals have high degree.

The following definitive theorem was obtained by Nies, Stephan and Terwijn.

Theorem 10.4.8 (Nies, Stephan, and Terwijn [232])**.** *For every set A, the following are equivalent.*

(I) *A is high.*

(II) *$\exists B \equiv_T A$, B is computably random but not Martin-Löf random.*

(III) *$\exists C \equiv_T A$, C is Schnorr random but not computably random.*

Furthermore, the same equivalence holds if one considers left-c.e. reals.

Before we prove this result, we mention that the situation outside of the high degrees is very interesting. To wit, we have the following.

Theorem 10.4.9 (Nies, Stephan and Terwijn [232])**.** *Suppose that a set A is Schnorr random and does not have high degree. (That is, $A' \not\geq_T \emptyset''$. Then A is Martin-Löf random.*

Proof. Suppose that A is not of high degree and covered by the Martin-Löf test $A \subset \cap_i U_i$. Let f be the function that computes on argument n the stage by which U_n has enumerated a $[\sigma] \in U_{n,s}$ with $A \in [\sigma]$. Note that f is A-computable, and hence computable relative to an oracle which is not high. It follows that there is a computable function g such that $g(n) > f(n)$ for infinitely many n. Then consider the test $\{V_i : i \in \mathbb{N}\}$, found by setting $V_i = U_{i,g(i)}$. The $\cup_i V_i$ is a Schnorr-Solovay test, and hence A is not Schnorr random by Theorem 10.3.5. $\square$

Thus whenever it possible for the notions to be distinct, they are. We turn to the proof of Theorem 10.4.8. The following proof is taken from the paper of Nies, Stephan and Terwijn and in is included with their permission.

Proof. (III) $\Rightarrow$ (I) and (II) $\Rightarrow$ (I): follows from Theorem 10.4.9

(I) $\Rightarrow$ (II): Given A, the set B is constructed in two steps as follows. First a set F is constructed which contains information about A and partial information about the behaviour of recursive martingales – this information will then be exploited to define a partial recursive martingale that witnesses that the finally constructed computably random set B is not Martin-Löf random. (Note that a partial recursive martingale as is defined below can be transformed into a c. e. martingale by letting it equal 0 until, if ever, it

becomes defined.) The sets A and F will be Turing equivalent and the sets B and F will be wtt-equivalent.
Let $\langle \cdot\,,\cdot\rangle$ be Cantor's pairing function $\langle x,y\rangle = \frac{1}{2}\cdot(x+y)\cdot(x+y+1)+y$. Furthermore, the natural numbers can be split into disjoint and successive intervals of the form $\{z_0\}, I_0, \{z_1\}, I_1, \dots$ such that the following holds.

- The intervals $\{z_k\}$ contain the single element z_k.
- The intervals I_k are so long that for every $\sigma \in \{0,1\}^{z_k+1}$ and every partial martingale M defined on all extensions $\tau \in \sigma\hat{}\{0,1\}^*$ with $|\tau| \leqslant |\sigma| + |I_k|$ there are two extensions $\tau_{\sigma,0,M}, \tau_{\sigma,1,M}$ of length $|\sigma| + |I_k|$ such that M does not grow beyond $M(\sigma)\cdot(1+2^{-k})$ within I_k. These extensions can be computed from M. Without loss of generality it holds that $\tau_{\sigma,0,M} <_L \tau_{\sigma,1,M}$.
- The partition of the natural numbers in the intervals $\{z_0\}, I_0, \{z_1\}, I_1, \dots$ is computable. This can be done since one can compute from k a length for which an I_k of this length with the properties in the previous item exist [204, Remark 9] (Theorem 11.3.1), see also [265, 318].

Let $M_0, M_1, \dots$ be a recursive list of all partial recursive martingales. That is, the enumeration satisfies the following conditions:

- The uniform domain $\{(i,\sigma) : M_i(\sigma) \text{ is defined}\}$ is a recursively enumerable set.
- If $M_i(\sigma\tau)$ is defined for some non-empty string τ, then $M_i(\sigma)$, $M_i(\sigma 0)$, $M_i(\sigma 1)$ are also defined and their values are positive rational numbers.
- If $M_i(\sigma 0), M_i(\sigma 1)$ are defined, then $M_i(\sigma 0) + M_i(\sigma 1) = 2M_i(\sigma)$.
- $M_i(\lambda) = 1$ if it is defined.

Now a partial recursive martingale M is defined inductively as follows. The goal is to let M multiplicatively dominate all recursive martingales on its domain of definition (i.e. for every recursive martingale N there is a constant c such that for all σ, if $M(\sigma)\downarrow$ then $N(\sigma) \leqslant c \cdot M(\sigma)$), while M does not succeed on a set B constructed below. This then ensures that B is computably random. Furthermore, a set $F \equiv_T A$ is constructed such that

- $F(k)$ is coded into $B \upharpoonright z_{k+1}$ where "most" of the coding into B is done on the interval I_k.
- $F(\langle i,j\rangle)$ for $i \neq 0$ tells whether M_i is defined on all strings up to the length of $z_{\langle i,j+1\rangle+1}$. This is necessary to know since on each length M will be the weighted sum of some M_i's and only M_i's which are defined on all relevant inputs should be considered.

- M can decode $F(k')$ for all $k' < k$ from $B \upharpoonright z_k$. This information permits to compute M on all τ with $B \upharpoonright z_k \preccurlyeq \tau$ and $|\tau| \leqslant z_{k+1}$. If a set $\tilde{B} \neq B$ is considered, it might be impossible to retrieve F and therefore, $M(\tilde{B} \upharpoonright q)$ might be undefined for some q. Thus, M is a partial recursive and not a total recursive martingale.

Now the details of the constructions just outlined are given. First we give the definition of M. Although M will only be partial, $M(B \upharpoonright x)$ will be defined for all x. For each k and $\eta \in \{0,1\}^{z_k}$ where $M(\eta)$ is already defined, we will try to define $M(\tau)$ for all $\tau \in \eta \cdot \{0,1\}^*$ with $|\tau| \leqslant z_{k+1}$.

(1) $M(\lambda) = r_\lambda$ and $r_\lambda = 1$.

(2) Assume that $|\eta| = z_k$, $M(\eta)$, r_η are already defined and for all $l < k$ there are values a_l and strings $\sigma_l = \eta \upharpoonright z_l + 1$ such that $\tau_{\sigma_l, a_l, M}$ are defined and prefixes of η. Then

 (2.1) Compute $E = \big\{ i : \langle i, 0 \rangle < k \wedge (\forall j)[\, \langle i, j \rangle < k \rightarrow a_{\langle i,j \rangle} = 1 \,] \big\}$.

 (2.2) Let $D = \{\tau \in \eta\char`^\{0,1\}^* : |\eta| < |\tau| \leqslant z_{k+1}\}$.

 (2.3) Compute for all $e \in E$ and $\tau \in D$ the value $M_e(\tau)$.

(3) If the algorithm has gone through step (2.3) and all the computations there have terminated then

$$(\forall \tau \in D)\Big[\, M(\tau) = r_\tau + \sum_{e \in E} 2^{-2z_{\langle e,0 \rangle + 1} - 1} M_e(\tau) \,\Big]$$

where the sum is 0 for the case that $E = \emptyset$ and r_τ is defined inductively such that the conditions

$$M(\tau'0) + M(\tau'1) = 2M(\tau') \text{ and } r_{\tau'0} = r_{\tau'1}$$

are kept for all $\tau' \prec \tau$.

If the algorithm did not go through step (2.3), then $M(\tau)$, r_τ are undefined for all proper extensions τ of η.

The r_τ are necessary since at every level z_k, some M_i might be dropped from the sum and at most one new M_i is added. This new martingale is added if $k = 1 + \langle i, 0 \rangle$ and $a_{\langle i,0 \rangle} = 1$. Furthermore, it is added with the factor $2^{-2z_k - 1}$ which guarantees that $2^{-2z_k - 1} M_i(\eta)$ is at most $2^{-z_k - 1}$ for all $\eta \in \{0,1\}^{z_k}$. (Note that we have $M_i(\eta) \leqslant 2^{z_k}$ since $M_i(\lambda) = 1$.) But this increases the sum by at most $2^{-z_k - 1}$ and therefore can be compensated by r_τ: At every level, $r_\tau \geqslant 2^{-|\tau|}$ and at most $2^{-|\tau|-1}$ of this capital is lost in order to maintain the martingale property of M.
By highness of A, let f^A be an A-recursive function which dominates all recursive functions. Now define the set F as follows.

- $F(\langle 0, 0 \rangle) = 0$.
- $F(\langle 0, j+1 \rangle) = A(j)$ for all j.

- For $i > 0$, $F(\langle i,j\rangle) = 1$ if $F(\langle i,j'\rangle) = 1$ for all $j' < j$ and $M_i(\tau)$ is computed within $f^A(i+j)$ many steps for all $\tau \in \{0,1\}^*$ with $|\tau| \leqslant z_{\langle i+j+1,i+j+1\rangle+1}$. Otherwise $F(\langle i,j\rangle) = 0$.

Clearly $F \equiv_T A$. The set B is defined inductively.

(k.0) Assume that exactly $B \restriction z_k$ is defined.
Let $B(z_k) = 0$ if $M(B \restriction z_k\frown 0) \leqslant M(B \restriction z_k\frown 1)$
and $B(z_k) = 1$ otherwise.

(k.1) Assume that exactly $B \restriction z_k + 1$ is defined.
Let $\eta = B \restriction z_k + 1$ and $B \restriction z_{k+1} = \tau_{\eta,F(k),M}$.

We now need to show that the inductive definition of B goes through for all k. Note that the $a_{\langle i,j\rangle}$ in the construction of M always exist for $\eta \prec B$ and that they are just the bits $F(\langle i,j\rangle)$. So the decoding at the beginning of step (2) is possible. Furthermore, for all $i \in E$ with $i > 0$, $\langle i,j\rangle \in F$ for $j = 0,1,\ldots,j'$ where j' is the maximal j'' with $\langle i,j''\rangle < k$. Note that $j' \geqslant 0$ and thus M_i is defined on all strings of length up to z_{k+1}. Thus the computations in step (2.3) all terminate. So M is defined on all extensions of $B \restriction z_k$ of length up to z_{k+1}. It follows that B is defined up to z_{k+1} and $F(k)$ is coded into B.
Note that coding gives $F \leqslant_{wtt} B$. Furthermore, one can compute for each k the string $B \restriction z_k$ using information obtained from $F \restriction z_k$. So $B \leqslant_{wtt} F$. Since A and F are Turing equivalent, one has $B \equiv_T A$.
To see that B is not Martin-Löf random, it suffices to observe that $B(z_k)$ is computed from $B \restriction z_k$. Thus one can build a partial recursive martingale (and hence an c. e. martingale) N which ignores the behaviour of B on all intervals I_k but always bets all its capital on $B(z_k)$ which is computed from the previous values. This martingale N clearly succeeds on B.
To see that B is computably random, note first that M does not go to infinity on B: On z_k, M does not gain any new capital by the choice of $B(z_k)$. By choice of I_k, M can increase its capital on I_k at most by a factor $1 + 2^{-k}$. Since the sum over all 2^{-k} converges, the infinite product $\prod_k (1 + 2^{-k})$ also converges to some real number r and M never exceeds r. Now given any recursive martingale M' there are infinitely many programs i for M' which all compute M' with the same amount of time. Since f^A dominates every recursive function, there is a program i for M' such that for all j, $f^A(i+j)$ is greater than the number of steps to compute $M_i(\tau)$ for any string $\tau \in \{0,1\}^*$ with $|\tau| \leqslant z_{\langle i+j+1,i+j+1\rangle+1}$. It follows that $M_i(\eta) \leqslant 2^{2z_{\langle i,0\rangle+1}+1} \cdot M(\eta) \leqslant 2^{2z_{\langle i,0\rangle+1}+1} \cdot r$ for all $\eta \preccurlyeq B$. Thus B is computably random.

(I) $\Rightarrow$ (II), c. e. case[5]: If A is c. e. as a set then one can choose f^A such that f^A is approximable from below. Therefore also F is c. e. and the set B can be approximated lexicographically from the left: In step (k.0) the value $B(z_k)$ is computed from the prefix before it and in step (k.1) one first assumes that $B \upharpoonright z_{k+1}$ is given by $\tau_{B\upharpoonright z_k+1,0,M}$ and later changes to $\tau_{B\upharpoonright z_k+1,1,M}$ in the case that k is enumerated into F.

(I) $\Rightarrow$ (III): The construction of C is similar to the one of B above, with one exception: there will be a thin set of k's such that $B(z_k)$ is not chosen according to the condition (k.0) given above but $B(z_k) = 0$. These guaranteed 0's will be distributed in such a way that on the one hand they appear so rarely that the Schnorr bound cannot be kept while on the other hand they still permit a recursive winning strategy for the martingale. Now let

$$\psi(e,x) = z_{\langle\langle e,\Phi_e(x)\rangle,x\rangle+1}$$

for the case that $\varphi_e(x)$ is defined and uses $\Phi_e(x)$ many computation steps to converge, otherwise $\psi(e,x)$ is undefined. Note that ψ is one-one, has a recursive range and satisfies $\psi(e,x) \geqslant z_{x+1} > x$ for all (e,x) in its domain. Furthermore, let

$$p(x) = \begin{cases} p(y)+1 & \text{if } (\exists e \leqslant \log p(y))\,[\psi(e,y) = x\,] \text{ for some } y < x, \\ x+4 & \text{otherwise.} \end{cases}$$

The function p is computable, unbounded and attains every value only finitely often. Assume without loss of generality that φ_0 is total and that $f^A(x) \geqslant \psi(0,x)$ for all x, and let

$$g^A(x) = \max\{\psi(e,x) : \psi(e,x)\downarrow \leqslant f^A(x) \wedge e < \log(p(x)) - 1\}.$$

The set C is defined by the same procedure as B with one exception: namely $C(z_k) = 0$ if $z_k = g^A(x)$ for some $x < z_k$. So having F as above, the overall definition of C is the following:

(k.0) Assume that exactly $C \upharpoonright z_k$ is defined.
Let $C(z_k) = 0$ if $M(C \upharpoonright z_k{}^\frown 0) \leqslant M(C \upharpoonright z_k{}^\frown 1) \vee z_k \in \text{range}(g^A)$
and $C(z_k) = 1$ otherwise.

(k.1) Assume that exactly $C \upharpoonright z_k + 1$ is defined.
Let $\eta = C \upharpoonright z_k + 1$ and $C \upharpoonright z_{k+1} = \tau_{\eta,F(k),M}$.

The proof that $C \equiv_T A$ is the same as the proof that $B \equiv_T A$ except that one has to use the additional fact that g^A is recursive relative to A.
To see that C is not computably random, consider the following betting strategy for a recursive martingale N. For every x, let $G_x = \{\psi(e,x) :$

[5]The following is the crucial property of M : if $M(sigma)$ is defined for σ with domain $0,1,\ldots,z_k$ then M is defined either on no or on all extensions τ of σ with domain $0,1,2,\ldots,z_{k+1}-1$. Therefore one can compute M on all of these extensions and then search for the left-most and right-most one among these.

$\psi(e,x)\downarrow \wedge\, e < \log(p(x)) - 1\}$. Since ψ is one-one, these sets are all disjoint and every G_x contains a number z_k such that $C(z_k) = 0$. (Namely $z_k = \psi(0,x)$ for some x, since by assumption φ_0 is total.) Starting with $x = z_0$, the martingale N adopts for every G_x a St. Petersburg - like strategy to gain the amount $1/p(x)$ on it, using the knowledge that G_x contains some z_k. For this purpose, N sets aside one dollar of its capital. More precisely: If the next point y to bet on is not in the current G_x, N does not bet. If $y \in G_x$ and N has lost m times while betting on points in G_x, then N bets $2^m/p(x)$ of its capital on $C(y) = 0$. In case of failure, N stays with x and waits for the next element of G_x without betting intermediately. In case of success, N has gained on the points of G_x in total the amount $1/p(x)$ and updates x to the current value of y and m to 0. Because $|G_x| \leqslant \log(p(x)) - 1$ this strategy never goes broke. Note that $p(y) = p(x) + 1$ (because N switches from G_x to G_y on some $z_k = \psi(e,x)$). Thus one can verify inductively that – in the limit – N gains the amount $1/(z_0+4)+1/(z_0+5)+1/(z_0+6)+\ldots$, that is, goes to infinity. Thus N succeeds on C and C is not computably random.
To see that C is Schnorr random, assume by way of contradiction that for M_i and a recursive bound h we would have that $M_i(C \upharpoonright h(m)) > m$ for infinitely many m. But for almost all m, $g^A(\log\log(m)) > h(m)$. An upper bound for M on C is then given by $M(C \upharpoonright h(m)) \leqslant \log(m)\cdot r$ since M can increase its capital on any interval I_k only by $1+2^{-k}$ and furthermore only on those z_k which are in the range of g^A. But of the latter there are only $\log\log(m)$ many below $h(m)$. Since $\log(m)\cdot r\cdot 2^{2z_{\langle i,0\rangle+1}+1} < m$ for almost all m, one has that $M_i(C \upharpoonright h(m)) < m$ for almost all m. Thus C is Schnorr random.

(I) $\Rightarrow$ (III), c. e. case: If A is an c. e. set and f^A approximable from below, then g^A is also approximable from below; let g_s be this approximation. Now C is computably enumerable as witnessed by the following approximation C_s obtained from the definition of C, where the approximation C_s is defined from below by going through the stages $(k.0), (k.1)$ iteratively until the procedure is explicitly terminated.

(k.0) Assume that exactly $C_s \upharpoonright z_k$ is defined.
If $M_s(\sigma)$ is undefined for some $\sigma \in \{C_s \upharpoonright z_k{}^\frown 0, C_s \upharpoonright z_k{}^\frown 1\}$ then terminate the procedure to define C_s by going to (ter).
If $M_s(C_s \upharpoonright z_k{}^\frown 0) \leqslant M_s(C_s \upharpoonright z_k{}^\frown 1)$ or there is an $x < z_k$ such that $z_k = \psi(e,x)$ for some $x < z_k$ and $e < \log(p(x)) - 1$ and $g_s(x) = z_k$ then $C_s(z_k) = 0$ else $C_s(z_k) = 1$.

(k.1) Assume that exactly $C_s \upharpoonright z_k + 1$ is defined.
Let $\eta = C_s \upharpoonright z_k + 1$. If $M_s(\sigma)$ is undefined for some $\sigma \in \{C_s \upharpoonright z_k + 1{}^\frown\tau : |\tau| \leqslant |I_k|\}$ then terminate the procedure to define C_s by going to (ter).
Let $\eta = C_s \upharpoonright z_k + 1$ and $C_s \upharpoonright z_{k+1} = \tau_{\eta,F_s(k),M}$.

(ter) If the inductive definition above is terminated with $C_s = \eta$ for some string η, then one defines that C_s is the set with the characteristic function $\eta 0^\infty$.

Now consider different sets C_s and C_{s+1}. There is a first stage (k.a) in which the construction behaves differently for C_s and C_{s+1}. There are three cases:
Case 1. The difference occurs because one but not both procedures terminate in stage (k.a). Since this termination is due to $M_s(\sigma)$ or $M_{s+1}(\sigma)$ being undefined for the same string σ in both cases, it follows that the procedure for C_s terminates but that for C_{s+1} not. Since C_s is extended by zeroes only, it holds that so $C_s \leqslant_L C_{s+1}$.
Case 2. The procedure does not terminate for C_s, C_{s+1} at this stage and the stage is of the form (k.0). Then the only difference between the construction this stage for C_s, C_{s+1} can come from the case that $g_s(x) = z_k$ and $g_{s+1}(x) > z_k$. In this case $C_s(z_k) = 0$ and $C_{s+1}(z_k) = 1$, so $C_s <_L C_{s+1}$.
Case 3. The procedure does not terminate for C_s, C_{s+1} at this stage and the stage is of the form (k.1). Then the only possible reason is that $k \in F_{s+1} - F_s$. Recall that $\eta = C_s \upharpoonright z_k + 1 = C_{s+1} \upharpoonright z_k + 1$. It follows that $C_s <_L C_{s+1}$ by $\tau_{\eta,0,M} <_L \tau_{\eta,1,M}$.
This case distinction gives $C_0 \leqslant_L C_1 \leqslant_L \ldots$ and so the approximation witnesses that C is a left-c. e. set. □

10.4.3 Computable randomness and tests

Definition 10.4.10 (Downey, Griffiths, and LaForte [74]). A Martin-Löf test $\{V_n\}$ is *computably graded* if there exists a computable map $f : 2^{<\omega} \times \omega \to \mathbb{R}$ such that, for any $n \in \omega$, $\sigma \in 2^{<\omega}$, and any finite prefix-free set of strings $\{\sigma_i\}_{i \leqslant I}$ with $\cup_{i=0}^{I}[\sigma_i] \subseteq [\tau]$, the following conditions are satisfied:

1. $\mu(V_n \cap [\sigma]) \leqslant f(\sigma, n)$
2. $\Sigma_{i=0}^{I} f(\sigma_i, n) \leqslant 2^{-n}$
3. $\Sigma_{i=0}^{I} f(\sigma_i, n) \leqslant f(\tau, n)$

By combining conditions (1) and (2) it is immediately apparent that $\mu(V_n) \leqslant 2^{-n}$ for all n. Further, if condition (2) holds for any *finite* prefix free set $\{\sigma_i\}$ then it also holds for any infinite prefix free set of strings: the infinite sum is just the supremum of the associated finite sums, so is also no greater than 2^{-n}. Similarly, since (3) holds for finite prefix free sets it also holds for infinite prefix free sets. If $\cup_i \sigma_i = \tau$ then we can summarize conditions (1)-(3) by the following:

$$\mu(V_n \cap [\tau]) \leqslant \Sigma_{i=0}^{I} f(\sigma_i, n) \leqslant f(\tau, n) \leqslant 2^{-n}$$

.

A similar characterization was also found by Merkle, Mihailovic, and Slaman [205].

Definition 10.4.11 (Merkle, Mihailovic, and Slaman [205]). (i) A computable rational probability distribution (or measure) is a computable mapping $\nu : 2^{<}\omega \to \mathbb{Q}$, with $\nu(\lambda) = 1$ and $\nu(\sigma) = \nu(\sigma 0) + \nu(\sigma 1)$.

(ii) A *bounded Martin-Löf test* is a martin-Löf test for which there exists a computable rational probability distribution ν with

$$\mu(V_n \cap [\sigma]) \leqslant \frac{\nu([\sigma)]}{2^n},$$

for all $n \in \mathbb{N}$ and $\sigma \in 2^{<\omega}$.

It is not hard to see that a test is computably graded iff it is a bounded Martin-Löf test.

A real x withstands a computably graded test iff $x \notin \cap_n V_n$.

Theorem 10.4.12 (Downey, Griffiths and LaForte [74], Merkle, Mihailovic, and Slaman [205]). *A real x is computably random iff it withstands all computably graded tests (iff it withstands all bounded Martin-Löf tests).*

The equivalence follows immediately from the following:

Theorem 10.4.13 (Downey, Griffiths and LaForte [74][6]). *(i) From a computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ we can effectively define a computably graded test (V_n, f) such that*

$$\limsup_j G(x \restriction j) = \infty \to x \in \cap_n V_n$$

(ii) From a computably graded test (V_n, f) we can effectively define a computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ such that

$$x \in \cap_n V_n \to \limsup_j G(x \restriction j) = \infty$$

Proof. (i) Given martingale G we may assume without loss of generality that $G(\lambda) = 1$. Define test sets V_n via $V_n = \cup\{[\sigma] : G(\sigma) \geqslant 2^n\}$.

We note that not only does V_n satisfy the property $\mu(V_n) \leqslant 2^{-n}$ but also

$$\mu(V_n \cap [\sigma])/\mu(\sigma) \leqslant G(\sigma)/2^n$$

That is, the proportion of $[\sigma]$ that intersects with V_n, which is the proportion for which the martingale G exceeds 2^n, is no greater than $G(\sigma)/2^n$ (a consequence of Kolmogorov's inequality for martingales). If we define a computable function $f : 2^{<\omega} \times \omega \to \mathbb{Q}$ by $f(\sigma, n) = G(\sigma)\mu(\sigma)2^{-n}$ then the inequality can be rewritten as

$$\mu(V_n \cap [\sigma]) \leqslant f(\sigma, n)$$

We also note that for any prefix-free set of strings $\{\sigma_i\}$, $\Sigma_i f(\sigma_i, n) \leqslant 2^{-n}$, since $\Sigma_i f(\sigma_i, n) = 2^{-n}\Sigma_i G(\sigma_i)\mu(\sigma_i) \leqslant 2^{-n}$. This inequality on G follows from the fact that the average of G, weighted by $\mu(\sigma_i)$, is $G(\lambda) = 1$, if the strings σ_i partition the entire unit interval. (Kolmogorov's Inequality)

This is clear if the strings σ_i represent all 2^l strings of a fixed length l, the general case follows from this restricted case.

The function f satisfies condition (3) above as a consequence of the fact that $G(\cup_i \sigma_i) = \Sigma_i G(\sigma_i)\mu(\sigma_i)$.

Thus V_n and f satisfy (1), (2) and (3), and furthermore if $\limsup_j G(x \restriction j) = \infty$ then $x \in \cap_n V_n$ since, for all n,

$$[(\exists k_n)\ G(x \restriction k_n) \geqslant 2^n] \to [x \restriction k_n] \subseteq V_n$$

Establishing (ii) is more involved. We need a preliminary definition:

Definition 10.4.14. A function $f : X \to \mathbb{R}$ is *co-c.e.* if there exists a computable approximation $f : X \times \omega \to \mathbb{Q}$ such that for all $x \in X$, $\lim_{s\to\infty} f(x)[s] = f(x)$ and, for all $s \in \omega$, $f(x)[s] \geqslant f(x)[s{+}1]$.

Without loss of generality we may assume, for all n, that $V_{n+1} \subseteq V_n$. Given V_n and f we define the computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ via two intermediate functions: $h : 2^{<\omega} \times \omega \to \mathbb{R}$, a co-ce map, and $J : 2^{<\omega} \to \mathbb{R}$, a co-ce martingale.

Let P_σ be the collection of all finite partitions of σ into a finite prefix free set (for example $\{\sigma 0, \sigma 1\} \in P_\sigma$). We use an infimum over all such partitions to define h:

$$h(\sigma, n) = 2^{|\sigma|} \inf_{\{\sigma_i\} \in P_\sigma} \Sigma_i f(\sigma_i, n)$$

It is clear that $2^{|\sigma|}\mu(V_n \cap [\sigma]) \leqslant h(\sigma, n) \leqslant 2^{|\sigma|}2^{-n}$ as f satisfies conditions (1) and (2). Each partition of σ into a finite prefix free set, other than the singleton $\{\sigma\}$, is a union of a partition of $\sigma 0$ and a partition of $\sigma 1$. Thus the infimum in the definition of h gives us the property, for each n, $h(\sigma, n) = \min(\frac{1}{2}[h(\sigma 0, n) + h(\sigma 1, n)], 2^{|\sigma|} f(\sigma, n))$. We claim that $2^{|\sigma|} f(\sigma, n) \geqslant \frac{1}{2}[h(\sigma 0, n) + h(\sigma 1, n)]$. Otherwise, dividing both sides by $2^{|\sigma|}$, we would have for all partitions $\{\sigma_i\}_{i \leqslant I}$ of σ (other than $\{\sigma\}$) the inequality $f(\sigma, n) < \Sigma_{i=0}^{I} f(\sigma_i, n)$. But this is not possible by condition (3) on f. Thus, for each n the function h is a martingale:

$$h(\sigma, n) = \frac{1}{2}[h(\sigma 0, n) + h(\sigma 1, n)]$$

By saying h is co-ce we mean that $h(\sigma, n) = \lim_s h_s(\sigma, n)$, where h_s is a computable map into $\mathbb{Q}$, and $h_{s+1}(\sigma, n) \leqslant h_s(\sigma, n)$, for all s, σ and n. (For fixed s and n it is not necessarily the case that h_s is a martingale.) One method for effectively obtaining such a function h_s is to define $h_s(\sigma, n)$ as for $h(\sigma, n)$ *except* that the infimum is to be taken over the finite number of partitions of σ that contain strings no longer than $|\sigma| + s$.

Summarizing the bounds on h:

$$0 \leqslant 2^{|\sigma|}\mu(V_n \cap [\sigma]) \leqslant h(\sigma, n) \leqslant 2^{|\sigma|} f(\sigma, n) \leqslant 2^{|\sigma|}2^{-n}$$

Let $J(\sigma) = \Sigma_{n=0}^{\infty} h(\sigma, n)$. Clearly $J(\sigma) \leqslant 2^{|\sigma|}\Sigma_0^{\infty} 2^{-n} = 2.2^{|\sigma|}$

The proof is concluded by establishing the following three points:

- If $[x \upharpoonright k_n] \subseteq V_n$ then $J(x \upharpoonright k_n) \geqslant n$.
- J is a co-ce martingale.
- From a co-ce martingale we can define a computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ such that $\limsup_n J(x \upharpoonright n) = \infty \to \limsup_n G(x \upharpoonright n) = \infty$.

For the first of these, let $\sigma = x \upharpoonright k_n$. Then $J(\sigma) \geqslant \Sigma_{m=0}^{n} h(\sigma, n) \geqslant 2^{|\sigma|}\Sigma_{m=0}^{n}\mu(V_m \cap [\sigma]) = 2^{|\sigma|}(n+1)2^{-|\sigma|} = n+1$. Then since $x \in \cap_n V_n$ iff such a k_n exists for all n, we have $x \in \cap_n V_n \to \limsup_j J(x \upharpoonright j) = \infty$.

The martingale property is inherited by J directly from h. To see that J is co-ce we must have an effective way of approximating it from above. Let $J_s(\sigma) = [\Sigma_{p=0}^{s} h_s(\sigma, p)] + [\Sigma_{p=s+1}^{\infty} 2^{|\sigma|}2^{-p}]$. This is computable as the first sum is a finite sum of computable rational numbers, and the second sum is simply $2^{|\sigma|}2^{-s}$. To see that $J_{s+1}(\sigma) \leqslant J_s(\sigma)$ consider each function value as an infinite sum of terms for $p = 0, 1, 2, \dots$. In passing from J_s to J_{s+1} we replace the h_s values for $p \leqslant s$ with h_{s+1} values (which are no larger), we replace the $p = s+1$ term with $h_{s+1}(\sigma, s+1) \leqslant 2^{|\sigma|}2^{-(s+1)}$, and we leave the terms unchanged for $p > s+1$. Thus J is a co-ce martingale as required.

Finally, we can define the required computable martingale G from J using the following result of Schnorr. $\square$

Lemma 10.4.15 (Schnorr [264])**.** *From a co-c.e. martingale $J : 2^{<\omega} \to \mathbb{R}$ we can effectively find a computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ such that for all strings σ, $G(\sigma) \geqslant J(\sigma)$.*

Proof. Let $J_s, s \in \omega$, be an effective non-increasing approximation to J: for each s, $J_s : 2^{<\omega} \to \mathbb{Q}$, $J_{s+1}(\sigma) \leqslant J_s(\sigma)$, and $J_s \to J$ pointwise as s increases.

Without loss of generality we may assume that $J(\sigma) < J_s(\sigma)$ (otherwise replace $J_s(\sigma)$ with $J_s(\sigma) + 2^{-s}$). We inductively define a computable martingale G such that $G(\sigma) > J(\sigma) + 2^{-|\sigma|}$ for all strings σ.

Base step: let $G(\lambda) = J_0(\lambda) + 1$, note that $G(\lambda) > J(\lambda) + 2^{-|\lambda|}$.

Inductive step: assume that $G(\sigma)$ has been defined, we show how to define $G(\sigma 0)$ and $G(\sigma 1)$. Define

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

where $n = \min\{i : \ J_i(\sigma 0) + J_i(\sigma 1) < 2(G(\sigma) - 2^{-|\sigma|})\}$, and let

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

Since the functions J_s are uniformly computable we can effectively find n, which must exist as $2(G(\sigma) - 2^{-|\sigma|}) > 2J(\sigma)$.

By definition $G(\sigma 1) \geqslant J(\sigma 1) + 2^{-|\sigma 1|}$, and $G(\sigma 0)$ is defined explicitly to satisfy the martingale property.

It remains to show that $G(\sigma 0) > J(\sigma 0) + 2^{-|\sigma 0|}$. By the choice of n we have

$$2G(\sigma) - 4.2^{-|\sigma|-1} > J_n(\sigma 0) + J_n(\sigma 1)$$

.

Rearranging we obtain:

$$2G(\sigma) - J_n(\sigma 1) - 2^{-|\sigma|-1} > J_n(\sigma 0) + 3.2^{-|\sigma|-1}$$

The left hand side is $G(\sigma 0)$, so we have

$$G(\sigma 0) > J_n(\sigma 0) + 3.2^{-|\sigma|-1} > J(\sigma 0) + 2^{-|\sigma 0|}$$

□

10.4.4 A machine characterization

It is also possible to obtain a machine characterization for computable randomness. This was recently obtained by Mihailović.

If M is a prefix-free machine, define

$$S_M^n = \{\sigma : K_M(\sigma) \leqslant |\sigma| - n\}.$$

Definition 10.4.16 (Mihailović [206]). A *bounded machine* is a prefix-free machine such that there is a computable rational probability distribution ν with $\mu([S_M^n] \cap [\sigma] \leqslant \frac{\nu([\sigma]}{2^n}$ for all σ, n.

Theorem 10.4.17 (Mihailović [206]). *A real x is computably random iff for all bounded machines M, ther eis a constant c such that for all n*

$$K_M(x \upharpoonright n) \geqslant n - c.$$

Proof. Suppose that $x \in \cap_{n\in\mathbb{N}} A_n$ for some (prefix-free) bounded Martin-Löf test with computable distribution ν. As per Schnorr's Theorem, if τ occurs in A_{2n+2} we can use KC to have a machine M with axiom $\langle |\tau| - n, \tau \rangle$. Prefix-freeness means that this is a KC set. And Schnorr's argument shows that $K_M(x \upharpoonright k)$ is bounded away from $k - c$ for all c.

To complete this direction of the proof we need to show that the machine is bounded. Suppose that $A_n = \{[\tau_{n,i}] : i \in \mathbb{N}\}$. Note that $S_M^n = \cup_{k\geqslant 1}\{\tau_{2(n+k),i:i\in\mathbb{N}}\}$, and hence for each σ and n, we have

$$[S_M^n] \cap [\sigma] = \cup_{k\geqslant 1} \cup_{i\in\mathbb{N}} ([\tau_{2(n+k)}] \cap [\sigma]).$$

Consequently,

$$\mu([S_M^n] \cap [\sigma]) \leqslant \sum_{k\geqslant 1} \mu(A_{2(n+k)} \cap [\sigma]) \leqslant \sum_{k\geqslant 1} \frac{\nu([\sigma])}{2^{2(n+k)}} \leqslant \frac{\nu([\sigma])}{2^n}.$$

Conversely, suppose that M is a bounded machine with $\forall c \exists n (K_M(x \upharpoonright n) < n - c)$. Then simply define $\mathcal{A} = \{A_n : n \in \mathbb{N}\}$ by letting $A_n = S_M^n$

for each n. By definition, $\mu(A_n \cap [\sigma]) \leqslant \frac{\nu([\sigma])}{2^n}$, and hence $\mathcal{A}$ is a bounded Martin-Löf test meeting x. $\square$

10.5 Kurtz Randomness

10.5.1 Basics

In [165], Stuart Kurtz introduced a new notion of randomness which looks at the idea from another perspective. Namely, instead of thinking of a real as random if it *avoided* all effectively given null tests, Kurtz suggested that a real should be considered as random if it *obeyed* every effectively given test of measure 1.

Definition 10.5.1 (Kurtz [165])**.** (i) A *Kurtz test* is a c.e. open set U such that $\mu(U) = 1$.

(ii) A real is called *Kurtz random* (or *weakly* 1-*random*[7]) iff for all Kurtz tests U, $\alpha \in U$.

Most of the definitions so far of tests have been negative. There is an equivalent formulation of Kurtz randomness in terms of a null tests.

Definition 10.5.2 (Wang [318])**.** A *Kurtz null test* is a collection $\{V_n : n \in \mathbb{N}\}$ of c.e. open sets, such that

(i) $\mu(V_n) \leqslant 2^{-n}$, and

(ii) There is a computable function $f : \mathbb{N} \mapsto (\Sigma^*)^{<\omega}$ such that $f(n)$ is a canonical index for a finite set of σ's, say, $\sigma_1, \dots, \sigma_n$ and $V_n = \cup\{[\sigma_1], \dots, [\sigma_n]\}$.

Notice that since we are only interested in in passing such tests, in the sense that $\alpha \notin \cap_{n\in\mathbb{N}} V_n$, we may assume that if $[\tau] \in V_m$ and $m > n$ then $\tau \supseteq \sigma$ for some $[\sigma] \in V_n$.

Theorem 10.5.3 (Wang [318], after Kurtz [165][8])**.** *A real α is Kurtz random iff it passes all Kurtz null tests.*

Proof. We show how measure 1 open sets correspond to Kurtz null tests. Let U be a c.e. open set with $\mu(U) = 1$. We define V_n in stages. To define V_1, enumerate U until a stage s is found with $\mu(U_s) > 2^{-1}$. The let $V_1 = \overline{U_s}$. Note that V_1 is of the correct form, to be able to define f. Of course for V_n we enumerate enough of U to have $\mu(U_{s_n}) > 2^{-n}$. For the converse reverse the reasoning. $\square$

[7]Kurtz referred to this notion weak randomness.

[8]This characterization is not explicit in Kurtz's thesis, but is implicit in some of the proofs.

There is a nice martingale definition of Kurtz randomness:

Theorem 10.5.4 (Wang [318])**.** *A real α is Kurtz random iff there is no computable martingale F and nondecreasing function h, such that for* almost all n,

$$F(\alpha \upharpoonright n) > h(n).$$

Proof. (Downey, Griffiths, Reid [75]) We follow the proof of Downey, Griffiths and Reid [75]. First note that saying that there is some F, h that $F(x \upharpoonright n) \geqslant h(n)$ for almost all n is equivalent to saying that there is some h such that t $F(x \upharpoonright n) \geqslant h(n)$ for *all* n. For suppose that $F(x \upharpoonright n) \geqslant h(n)$ for almost all n. This means there is a number n_1, such that for all $n \geqslant n_1$, $F(x \upharpoonright n) \geqslant h(n)$. Replace $h(n)$ by the similarly computable, unbounded and nondecreasing function $\widehat{h}(n)$:

$$\widehat{h}(n) = \begin{cases} h(n) & \text{if } n \geqslant n_1 \\ 0 & \text{otherwise.} \end{cases}$$

Then for all n, $F(x \upharpoonright n) \geqslant \widehat{h}(n)$. Hence it can be assumed without loss of generality that $F(x \upharpoonright n) \geqslant h(n)$ for all n.

Suppose that x is not Kurtz random as witnessed by the Kurtz null test $\{W_i : i \in \omega\}$ where each W_i is a finite, nested, prefix-free set of extensions of strings $\{\sigma_{i,n}\}$ of length $g(i)$. Define

$$F(\tau) = \sum_{n \in \omega, \sigma_{n,m} \in W_n} w_{n,m}(\tau)$$

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

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

It is straightforward to check that $w_{n,m}$ satisfies the martingale property, and so F will also satisfy it. F is computable: to compute $F(\tau)$ to within 2^{-s}, we notice that if $l = |\tau|+s+1$ then all strings in U_l can increase $F(\tau)$ by at most $2^{-(s+1)}$ (this can be seen from the fact that the increase in $F(\lambda)$ due to U_l is $\mu(U_l) \leqslant 2^{-l}$). Similarly all sets U_l, U_l+1, $\ldots$ have a combined effect on $F(\tau)$ of no more than 2^{-s}. So let $F_s(\tau) = \Sigma_{n \leqslant |\tau|+s, \sigma_{n,m} \in W_n} w_{n,m}(\tau)$, this is within 2^{-s} of $F(\tau)$. Consider $x \upharpoonright n$, for any n. Since $(\forall i)(x \in W_i)$, for any j where $g(j) \leqslant n$ there is a string $\sigma_{j,p}$ in W_j such that $\sigma_{j,p} \preccurlyeq x \upharpoonright n$. This means that

$$\begin{aligned} F(x \upharpoonright n) &\geqslant \sum_{j : g(j) \leqslant n} 1 \\ &\geqslant |\{j : g(j) \leqslant n\}| \end{aligned}$$

Define $h(n) = |\{j : g(j) \leqslant n\}|$. Then $h(n)$ is defined for all n. $h(n)$ is computable, because for any n, $g(n) \geqslant n$. Hence it is only necessary to calculate the first $n+1$ (computable) values of g before $h(n)$ is known. $h(n)$ is non-decreasing and unbounded, and as $F(x \upharpoonright n) \geqslant h(n)$ for all n, as required.

Conversely, suppose that there is a computable martingale F and computable, unbounded nondecreasing function h such that $(\forall n)(F(x \upharpoonright n) \geqslant h(n))$. By Schnorr's Lemma 10.4.15, we can consider F to be a computable map directly into $\mathbb{Q}$, rather than be computably approximated with range a subset of the reals[9]. For each k, define c_k to be the least n such that $h(n) \geqslant 2^{k+1}$. Because of the nature of h, c_k exists and can be computed for each k. Define $U_k = \{\sigma : |\sigma| \leqslant c_k \text{ and } F(\sigma) > 2^k\}$. Each U_k is a finite set, is computable as F is a computable martingale, and has measure $\mu(U_k) \leqslant 2^{-k}$, so the set $\{U_k : k \in \omega\}$ is a Kurtz null test. We have that for all n, $F(x \upharpoonright n) \geqslant h(n)$, in particular, for all k:

$$\begin{aligned} F(x \upharpoonright c_k) &\geqslant h(c_k) \\ &\geqslant 2^{k+1}. \end{aligned}$$

Thus by the definition of U_k, $x \in U_k$ for all k, and so x is not Kurtz random, as required. □

Of course we can generalize Kurtz randomness to higher levels. Thus a real α is *weakly n-random,* or *Kurtz n-random* indexweakly n-random if it is a member of all Σ_n^0 classes of measure 1. The reader should have some caution here since now the distinction between *classes* and *open sets* is important. It is *not* true that being a member of every Σ_n class equates to being a member of every $\Sigma_1^{\emptyset^{(n-1)}}$-class. For instance, a 2-generic is in every $\Sigma_1^{\emptyset'}$-class, but cannot be 1-random and as we will see not Kurtz 2-random. This should be strongly contrasted with Section 9.7.1. We can do a little with classes.

Lemma 10.5.5 (Kurtz [165], Kautz [140]). *Let $n \geqslant 2$.*

(i) Then for any Σ_n^0 class C we can uniformly and computably obtain the index of a $\Sigma_2^{\emptyset^{(n-2)}}$-class $\widehat{C} \subseteq C$ with $\mu(\widehat{C}) = \mu(C)$.

(ii) For any Π_n^0 class V we can uniformly and computably obtain the index of a $\Pi_2^{\emptyset^{(n-2)}}$-class $\widehat{V} \supseteq V$ with $\mu(\widehat{V}) = \mu(V)$.

Proof. Let C be a Σ_n^0 class. Thus $C = \cup_i T_i$ with T_i Π_{n-1}^0. By Theorem 9.7.3, we can uniformly find, for each i and j, a a $\Pi_1^{\emptyset^{(n-2)}}$-class $\widehat{C}_{i,j} \subseteq T_i$ with $\mu(T_i) - \mu(\widehat{C}_{i,j}) \leqslant 2^{-j}$. Let $\widehat{C} = \cup_{j,i}\widehat{C}_{i,j}$. Then $\widehat{C} \subseteq C$, and $\mu(\widehat{C}) = \mu(C)$. The proof of (ii) is the same. □

[9] Since F is computable it is both c.e. and co-c.e. (meaning $-F$ is c.e.). Schnorr's result is that from any co-c.e. martingale $J : 2^{<\omega} \to \mathbb{R}$ we can effectively find a computable martingale $G : 2^{<\omega} \to \mathbb{Q}$ such that for all strings σ, $G(\sigma) > J(\sigma)$. In our proof we can use such a G in place of F, scaled down by a rational q if necessary to ensure $G(\lambda) \leqslant 1$ and use h scaled down by a similar factor.

Corollary 10.5.6. *Let $n \geqslant 2$. Then α is Kurtz n-random iff α is in every $\Sigma_2^{\emptyset^{(n-2)}}$-class of measure 1.*

Finally, we can have a "1-jump" characterization of Kurtz n-randomness via the analog of a Kurtz null test.

Theorem 10.5.7 (Kautz, Wang, after Kurtz [165]). *(i) A real α is Kurtz n-random iff for each $\emptyset^{(n)}$ computable sequence of Σ_{n-1}^0 classes $\{S_i : i \in \mathbb{N}\}$, with $\mu(S_i) \leqslant 2^{-i}$, $\alpha \notin \cap_i S_i$.*

(ii) Consequently, α is Kurtz n-random iff for every $\emptyset^{(n)}$ computable sequence of open $\Sigma_1^{\emptyset^{(n-2)}}$ classes $\{S_i : i \in \mathbb{N}\}$, with $\mu(S_i) \leqslant 2^{-i}$, $\alpha \notin \cap_i S_i$.

Proof. A real α is Kurtz n-random iff it avoids each Π_n^0 nullset T. Then a Π_n^0 nullset T is of the form $\cap_i U_i$, with the U_i a uniform sequence of Σ_{n-1}^0 classes. Then the U_i can be replaced by $\cap_{j \leqslant i} U_j$ we can assume the sequence is shrinking, and the measure goes to zero. Finally, using $\emptyset^{(n-1)}$ as an oracle, we can effectively find an index where $\mu(U_{k(i)}) < 2^{-k}$, giving the result. (ii) Follows by Theorem 9.7.3. □

It follows that Kurtz 2-randomness is very natural. In Chapter 9, we introduced the notion of Martin-Löf randomness. There the first possibility was simply a computable collection $\{U_n : n \in \mathbb{N}\}$ of c.e. open sets with $\mu(U_n) \to 0$, that is, without knowing an upper bound on the measure of the U_n. Notice that $\cap_{n \in \mathbb{N}} U_n$ in this case forms a Π_2^0 class of measure 0. Hence the complement is a Σ_2^0 class of measure 1. *Thus, Kurtz 2-randomness is the same as passing all such "generalized" Martin-Löf tests.* This type of randomness was apparently first studied by Gaifman and Snir [119].

Kurtz n-randomness and n-randomness are intertwined as follows.

Theorem 10.5.8 (Kurtz [165]). *(i) Every n-random real is Kurtz n-random.*

(ii) Every Kurtz $n+1$-random real is n-random.

Proof. (i) Assume that α is n-random. The proof of Theorem 10.5.3 shows that for each n, and each Kurtz n-test, there is an associated Σ_n^0 Kurtz null test. Then the real is in the Σ_n^0 open set of measure 1 iff it passes the n-test. As α is n-random, it passes all n-tests.

(ii) The complement of a Σ_n^0 Martin-Löf test is a Σ_{n+1}^0 Kurtz test. □

We will see in the next Chapter that neither of the implications in Theorem 10.5.8 can be reversed.

As we have remarked, in many ways weak 2-randomness is the first place where "typical" random behaviour happens. A good example of this is the following result saying that the behaviour of weak 2-random reals is very unlike random reals with high information like those given by the Kučera-Gács There.

Theorem 10.5.9 (Downey, Nies, Weber, Yu [?])**.** *Each weakly 2-random degree forms a minimal pair with* $\mathbf{0'}$.

Proof. Suppose not, so there is a non-computable Δ_2^0 set Z and a weakly 2-random set A so that $Z = \Phi_e^A$. Since Z is Δ_2^0, there is an effective approximation $Z[s]$ so that $\lim_s Z(n)[s] = Z(n)$ for all n. Define

$$S_e = \{X | (\forall n)(\forall s)(\exists t > s)(\Phi_e^X(n)[t] \downarrow = Z(n)[t])\}.$$

S_e is Π_2^0 and $A \in S_e$. Since A is weakly 2-random, $\mu(S_e) > 0$. Thus there is a finite set $\Sigma \subseteq 2^{<\omega}$ and an open set $U = \bigcup_{\sigma \in \Sigma} V_\sigma$ so that

$$\mu(U \cap S_e) > \frac{3}{4}\mu(U).$$

To effectively compute $Z(n)$, we simply need to search for a finite set $\Xi \subseteq 2^{<\omega}$ with

$$\mu(\bigcup_{\tau \in \Xi} V_\tau) > \frac{1}{2}\mu(U) \text{ and } \bigcup_{\tau \in \Xi} V_\tau \subseteq U$$

so that for any $\tau_0, \tau_1 \in \Xi$,

$$\Phi_e^{\tau_0}(n) \downarrow = \Phi_e^{\tau_1}(n) \downarrow .$$

Then $Z(n) = \Phi_e^{\tau_0}(n)$. Thus Z is computable, contradiction. □

Corollary 10.5.10 (Downey, Nies, Weber, Yu [?])**.** *There is no universal Generalized Martin-Löf test.*

Proof. Suppose there is a universal Generalized Martin-Löf test. Then there is a non-empty Π_1^0 class containing only weakly 2-random reals. Then, by the Kreisel Basis Theorem, there is a weakly 2-random set computed by $\mathbf{0'}$. This contradicts Theorem 10.5.9. □

We can use Theorem 10.5.9 together with an argument of the second author to obtain a nice charcaterization of weak-2-randomness.

Theorem 10.5.11 (Hirschfeldt, unpubl.)**.** *Suppose that* $\{U_n : n \in \omega\}$ *is a generalized Martin-Löf test. Then there is a computably enumerable noncomputable set* B *such that* $B \leqslant_T A$ *for every Martin-Löf random set* $A \in \cap_n U_n$.

Corollary 10.5.12. *A real* A *is weakly 2-random iff* A *is 1-random and its degree forms a minimal pair with* $\mathbf{0'}$.

Proof. (of Theorem 10.5.11) We define $c(n,s) = \mu(U_n)[s]$, here assuming that tests are nested and each U_n is presented by an antichain. We put x into $B[s]$ if $W_e \cap B = \emptyset[s]$, $x \in W_e[s]$ and $c(x,s) < 2^{-e}$. We define a functional Γ as follows. If $\sigma \in U_{n, \text{ at } s}$, declare $\Gamma^\sigma(n) = B(n)[s]$. Finally we will define a Solovay test by saying that if $x \in B_{\text{at } s}$, put $U_x[s]$ into S.

Then one verifies (i) $\mu(S) \leqslant 1$ (since we use the cost function 2^{-e}), (ii) B is noncomputable as $\mu(U_n) \to 0$, obtaining that if $A \in \cap_n U_n$ and A is 1-random, then since A will avoid S, $\Gamma^A =^* B$. □

10.5.2 A machine characterization of Kurtz randomness

We follow Downey, Griffiths and Reid [75] who gave a machine characterization of Kurtz randomness. This characterization was along the lines of the machine characterization for Schnorr randomness. That is, it uses initial segment complexity but measured with a more restrictive class of machines.

Definition 10.5.13 (Downey, Griffiths and Reid [75]). An *computably layered machine* is a prefix-free machine where there is a related computable function

$$f : \omega \to (2^{<\omega})^{<\omega}$$

such that

- $\bigcup_i f(i) = \text{dom}M$.
- If $\gamma \in f(i+1)$, then $(\exists \tau \in f(i))$, such that $M(\tau) \preccurlyeq M(\gamma)$.
- 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. In the next Theorem, the reader should keep in mind that if σ is not in the range of a machine V, then $K_V(\sigma) = \infty$.

Theorem 10.5.14. *x is Kurtz random iff for all computably layered machines M,*

$$(\exists d)(\forall n)(K_M(x \upharpoonright n) \geqslant n - d).$$

Proof. Suppose that x is not Kurtz random, as witnessed by a (nested) Kurtz null test $\{W_i : i \in \omega\}$. Recall that for all k, $W_k = [\sigma_{k,1}] \cup \ldots \cup [\sigma_{k,m_k}]$ for some finite, prefix-free set of strings. Consider the lengths $(|\sigma_{2n+2,i}| -$

$(n+1))$ for $n \in \omega$, $1 \leqslant i \leqslant m_n$:

$$\begin{aligned}
\sum_{n\in\omega}\sum_{i=1}^{m_n} 2^{-(|\sigma_{2n+2,i}|-(n+1))} &= \sum_{n\in\omega} 2^{n+1}\sum_{i=1}^{m_n} 2^{-|\sigma_{2n+2,i}|} \\
&= \sum_{n\in\omega} 2^{n+1}\mu(W_{2n+2}) \\
&\leqslant \sum_{n\in\omega} 2^{n+1}2^{-2n-2} \\
&\leqslant \sum_{n\in\omega} \frac{2^{-n}}{2} \\
&\leqslant 1.
\end{aligned}$$

Thus by the Kraft-Chaitin theorem there is a prefix-free machine mapping strings of length $(|\sigma_{2n+2,i}| - (n+1))$ to $\sigma_{2n+2,i}$ for $n \in \omega$, $1 \leqslant i \leqslant m_n$. 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 algorithm in the effective Kraft-Chaitin theorem that was used to produce M we can list the domain elements.

- By the definition of f,
$$\operatorname{dom}(M) = \bigcup_{n\in\omega} f(n).$$

- Since the Kurtz null test is nested, it follows that for all τ in $f(i+1)$, there exists a $\widehat{\tau}$ in $f(i)$ such that $[M(\tau)] \subseteq [M(\widehat{\tau})]$, that is, $M(\widehat{\tau}) \preccurlyeq M(\tau)$.

- 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 in fact a computably layered machine. Since for all n, x is in W_n, it follows that for all n, there is some i, such that $\sigma_{2n+2,i} = x \upharpoonright k$, where k is the length of $\sigma_{2n+2,i}$, and by the definition of M it follows that $(\forall n)(\exists k)(K_M(x \upharpoonright k) < k - n)$, as required.

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

$$W_k = \cup\{[\sigma] : K_M(\sigma) < |\sigma| - k)\}.$$

Suppose $\sigma \in W_k$. Then $K_M(\sigma) < |\sigma| - k$, so if $M(\tau) = \sigma$, then $|\tau| < |M(\tau)| - k$, so τ cannot be an element of $f(j)$ for any $j < k$. Thus

$$W_k = \bigcup_{\substack{j \geqslant k \\ \gamma \in f(j)}} [M(\gamma)].$$

However, suppose $\gamma \in f(j)$ where $j > k$. Then by the properties of f, it follows that there is a $\tau \in f(k)$ such that $M(\tau) \preccurlyeq M(\gamma)$. So the definition of W_k simplifies to:

$$W_k = \bigcup_{\gamma \in f(k)} [M(\gamma)].$$

Since $f(k)$ is finite and computable, and M is computable, W_k must be finite and computable, and since it is a standard result that for all k, $\mu(W_k) \leqslant 2^{-k}$, $\{W_k : k \in \omega\}$ 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. $\square$

10.5.3 Other characterizations of Kurtz randomness

In this section we give two more characterizations of Kurtz randomness. One is another machine one, this time in terms of the machines used to characterize Schnorr randomness. The other is a Solovay-type characterization.

We have already seen Schnorr randomness can be characterized in terms of *computable* machines. It is an indication of the close relationship between Schnorr and Kurtz randomness that this class of machines can be used to characterize Kurtz randomness also. In Theorem 10.3.3 we proved that a real x is *Schnorr* random iff for all computable machines M, there is a constant c such that for all n, $K_M(x \upharpoonright n) \geqslant n - c$. Computable machines characterize Kurtz randomness according to the following theorem:

Theorem 10.5.15 (Downey, Griffiths and Reid [75])**.** *A real x is* not *Kurtz random iff 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 $\{U_n : n \in \omega\}$, such that $(\forall n)(x \in U_n)$. Each U_n is a finite, computable set of extensions of strings: $U_n = \cup\{[\sigma_{n,i}] : i \leqslant m_n\}$. Assume that

$$(\forall n)(\forall i \leqslant m_n)(|\sigma_{n,i}| = g(n))$$

where $g : \omega \to \omega$ is a computable function. As noted in Theorem 10.5.14,

$$\sum_{n\in\omega}\sum_{i=1}^{m_n} 2^{-(|\sigma_{2n+2,i}|-(n+1))} \leqslant 1,$$

so by the effective Kraft-Chaitin theorem 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 $i \leqslant m_n$.

Let $\alpha = \sum_{\sigma\in\text{dom}M} 2^{-|\sigma|}$. Then

$$\begin{aligned}\alpha &= \sum_{n\in\omega}\sum_{i\leqslant m_n} 2^{-|\sigma_{2n+2,i}|}2^{n+1} \\ &= \sum_{n\in\omega} \mu(W_{2n+2})2^{n+1}.\end{aligned}$$

Note that

$$\begin{aligned}\sum_{n>s} \mu(W_{2n+2})2^{n+1} &\leqslant \sum_{n>s} 2^{-2n-2}2^{n+1} \\ &\leqslant \sum_{n>s} 2^{-n-1} \\ &< 2^{-s}\end{aligned}$$

So if α_s is defined to be

$$\alpha_s = \sum_{n\leqslant s} \mu(W_{2n+2})2^{n+1},$$

it follows that $|\alpha - \alpha_s| < 2^{-s}$, and so M is a computable machine.

Since $(\forall n)(x \in U_{2n+2})$, for each n there is obviously an $i \leqslant m_n$ such that $x \in [\sigma_{2n+2,i}]$, or put another way, $(\forall n)(x \restriction g(2n+2) = \sigma_{2n+2,i})$. By the definition of M, this means that

$$(\forall n)(K_M(x \restriction g(2n+2)) < g(2n+2) - n),$$

so M and $f(n) = g(2n+2$ are the required machine and computable function.

Conversely, suppose we have a computable machine M and a computable function f such that $(\forall d)(K(x \restriction f(d)) < f(d) - d)$, and we wish to show x is not Kurtz random. Let

$$V_d = \cup\{[\sigma] : |\sigma| = f(d), K_M(\sigma) < f(d) - d\}$$

It is a standard result that $\mu(V_d) \leqslant 2^{-d}$. Furthermore, V_d is a finite set of strings - as there are $2^{f(d)}$ strings of length $f(d)$, V_d cannot contain more than this many strings - and 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 $x \in \bigcap_d V_d$ and therefore x is not Kurtz random. $\square$

We turn to a Solovay-type characterization. A variant of the Martin-Löf characterization of random numbers is the Solovay characterization. This can be adapted to Kurtz randomness as follows:

Definition 10.5.16 (Downey, Griffiths, and Reid [75]). 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. We say that a 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 10.5.17 (Downey, Griffiths, and Reid [75]). *A real x is Kurtz random iff 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 $x \in \bigcap_i W_i$. This null test is finite and computable, and since

$$\sum_{i=0}^{\infty} \mu(W_i) \leqslant \sum_{i=0}^{\infty} 2^{-i},$$

the sum of the measures is finite and a computable real. 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 a prefix-free set by $S_k = \{y : y \text{ is in } 2^k \text{ of } V_0, \ldots, V_{f(2^k)}\}$ 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 at least 2^k sets V_n, we know that

$$\begin{aligned} \sum_{n=0}^{f(2^k)} \mu(V_n) &\geqslant \sum_{y \in S_k} 2^k 2^{-|y|} \\ &\geqslant 2^k \mu(W_k). \end{aligned}$$

Since, by assumption, $\sum_{n=0}^{\infty} \mu(V_n) \leqslant 1$, it follows that

$$1 \geqslant 2^k \mu(W_k),$$

and hence that

$$\mu(W_k) \leqslant 2^{-k},$$

as required of a Kurtz null test. 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 also 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, …, $V_{f(n)}$. In particular this means that for all k, x is at least 2^k of V_0, V_1, …, $V_{f(2^k)}$. Hence for all k, $x \in W_k$, and so x is not Kurtz random, as required. $\square$

We observe that given any Kurtz-Solovay test, (f, V), a new Kurtz-Solovay test $(id, \widehat{V})$ can be defined via

$$\widehat{V}_k = V_{f(k)} \cup \ldots \cup V_{f(k+1)-1}.$$

10.5.4 Schnorr randomness via Kurtz randomness

Schnorr randomness can also be characterized in terms of Kurtz null tests.

Definition 10.5.18 (Downey, Griffiths and Reid [75]). A *Kurtz array* is a uniform collection of Kurtz null tests

$$\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geqslant 1\}$$

with the property that

$$(\forall j)(\forall n)(\mu(W_{j,n}) \leqslant 2^{-j}2^{-n}).$$

Theorem 10.5.19. *x is Schnorr random iff for all Kurtz arrays $\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geqslant 1\}$, $(\exists n)(\forall j)(x \notin W_{j,n})$.*

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 $\mu(U_n^s) \geqslant \frac{2^{j+1}-1}{2^{j+1}}\mu(U_n)$. 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 $j \geqslant 1$. 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

$$\begin{aligned}\mu(W_{j,n}) &\leqslant \mu(U_n^{s_j}) - \frac{2^j-1}{2^j}\mu(U_n)\\ &\leqslant \mu(U_n) - \frac{2^j-1}{2^j}\mu(U_n)\\ &\leqslant \mu(U_n)[1-(1-2^{-j})]\\ &\leqslant \mu(U_n)2^{-j}\\ &\leqslant 2^{-n}2^{-j},\end{aligned}$$

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 $\{\{W_{j,n} : n \in \omega\} : j \in \omega, j \geqslant 1\}$ and x such that $(\forall n)(\exists j)(x \in W_{j,n})$. We define $U_n = \cup_{j\in\omega} W_{j,n+1}$. 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

$$\begin{aligned}\mu(U_n) &= \sum_{j\in\omega}\mu(W_{j,n+1})\\ &\leqslant \sum_{j\in\omega}2^{-j}2^{-n+1}\\ &\leqslant 2^{-n},\end{aligned}$$

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 $\{W_{j,n+1} : j \leqslant s\}$. Hence U_n is a Schnorr test, and x is not Schnorr random. □

10.5.5 Computably enumerable Kurtz random reals

No Kurtz random real can be a computably enumerable *set*. In fact:

Theorem 10.5.20 (Jockusch, see Kurtz [165])**.** *If α is Kurtz random then α is bi-immune.*

Proof. Let W_e be an infinite computably enumerable set. Let $U_e = \{x : n \in W_e \wedge x(n) = 0\}$. Then U_e has measure 1, it is a c.e. open set of reals, and any Kurtz random set must be in U_e. The same argument applies to the complement of any Kurtz random set. □

Corollary 10.5.21 (Jockusch, see Kurtz [165])**.** *There are $2^{\aleph_0}$ degrees that contain no Kurtz random reals.*

Proof. Jockusch [130] showed that there are $2^{\aleph_0}$ bi-immune free degrees. □

While no computably enumerable *set* can be Kurtz random, as with Martin-Löf randomness, the same is not true for computably enumerable degrees. Kurtz [165], Corollary 2.3a proved that every computably enumerable degree contains a Kurtz random real. The following improves this to computably enumerable *reals*.

Theorem 10.5.22 (Downey, Griffiths, and Reid [75])**.** *There is a Kurtz random c.e. real in each non-zero c.e. degree*

Proof. We combine a technique of avoiding test sets with permitting and coding of a non-computable set. We use an enumeration of Kurtz null tests (including "finite tests") and ensure that if a test really is a Kurtz null test, not a finite test, then our real is not in the null set of the test. Let B be an arbitrary non-computable c.e. set; we will build our real z below this set. In fact the permitting will produce the property that $B_s \upharpoonright n = B \upharpoonright n \rightarrow z_s \upharpoonright n = z \upharpoonright n$.

Notation: (i) for any string τ let τ^+ represent the string obtained by changing the last bit of τ from 0 to 1 or vice versa and (ii) let $\langle .,. \rangle$ represent a computable bijective map $\omega \times \omega \rightarrow \omega$ which is increasing in both variables.

The requirement R_e will deal with a test set U_k^i, where $e = \langle i, j \rangle$ for some j, and k will be determined during the construction to make sure the maximum possible measure of the test set is sufficiently small. The c.e. approximation z_s to z will be defined as $\inf(X_s)$, where the set $X_s \subseteq [0,1]$ has no intersection with U_k^i once R_e has acted. For all s we ensure $X_{s+1} \subseteq X_s$.

R_e : If U^i is a Kurtz test then $(\exists k, s) X_s \cap U_k^i = \emptyset$, where $e = \langle i, j \rangle$.

P_j : $\exists \pi = z \upharpoonright n_j$ such that z codes $B \upharpoonright j$ in bits $|\pi| - (j-1)$ to $|\pi|$ of z, ie.

$$z((n_j - j) + i) = B(i-1), \ \ 0 \leqslant i < j$$

Strategy for a single requirement R_e.

If τ_e if it is not defined, let $\tau_e = \tau_{e-1}0$ (where τ_{e-1} is a previously defined string), and choose k large enough so that $\mu(U_k^i) < \frac{1}{4}\mu([\tau_e]) = \frac{1}{4}2^{-|\tau_e|}$. If R_e never acts and is never initialized then the entire construction will take place within $[\tau_e]$ producing a real z extending τ_e. Each set $[\tau_e] \subseteq X_s$ at stage s, but τ_e are not "permanent" like X_s in the sense that $X_{s+1} \subseteq X_s$, but sometimes $\tau_e[s+1]$ does not extend $\tau_e[s]$ (here $[s]$ and $[s+1]$ represent stage s and stage $s+1$ definitions of τ_e, *not* bits s and $s+1$ of the string).

We say R_e *requires attention* if U_k^i becomes non-empty, that is, it appears U^i is a genuine Kurtz test. Once it requires attention it will act iff $B_s \upharpoonright |\tau_e|$ changes. The action then is to set $X_{s+1} = X_s \cap \overline{U_k^i} \cap \overline{[\tau_e]}$.

Thus if R_e acts the construction *will* move out of $[\tau_e]$.

Since U_k^i can eliminate a set of measure less than the measure of $[\tau_e]$, there will be a string σ_e extending τ_e^+ such that $[\sigma_e] \cap X_s = [\sigma_e]$. The construction will continue within such a cone $[\sigma_e]$.

Strategy for a single requirement P_h. Given τ_h extend it by h bits to code $B(0)$,..., $B(h-1)$. This extends the string τ_h defined by R_h. Requirement R_{h+1}, when defining τ_{h+1}, will use the extended version of τ_h.

Combining several strategies.

Initialization and Satisfaction: in the construction we use the additional notion of satisfaction: if R_e acts, where $e = \langle i,j \rangle$, then declare R_f *satisfied* for all $f = \langle i, j' \rangle$, $j \in \omega$. A requirement R_f remains satisfied for the rest of the construction, and never requires attention or acts. When a requirement R_e is *initialized* by higher priority action it means it's string τ_e is canceled, as is its choice k for Kurtz null test set U^i_k. When a requirement P_e is initialized it simply ceases to have any impact in updating strings in the construction.

The technical content of some strings τ_e will become permanently subsumed into strings of higher priority requirements. This happens only when the requirement R_e has been satisfied, and if τ_e is declared to be *subsumed* then thereafter $\tau_e = \tau_{e-1}$.

CONSTRUCTION:

Stage 0— let $X_0 = [0,1]$, $\tau_0 = 0$, choose $k = 4$ for R_0.

Stage s>0 — (i) Define in sequence all τ_0 to τ_s that are undefined: if τ_e is subsumed then let $\tau_e = \tau_{e-1}$, otherwise let $\tau_e = \tau_{e-1}0$. Choose k associated with these requirements, k must be large enough so that $\mu(U^i_k) < \frac{1}{4}2^{-|\tau_e|}$, where $e = \langle i,j \rangle$ some $j \in \omega$.

(ii) If there is a requirement P_h that needs to update its string because of a $B \restriction h$ change, *and* for all R_e that require attention e is greater than h, then let P_h act (for least such h). Redefine τ_{h+1} through to τ_s as extensions of the new τ_h.

(iii) Act on all R_t, $t \leqslant s$, that require attention and for which $B \restriction p \neq B_{s-1} \restriction p$ where p is the length of τ_t when R_t was seen to require attention. The action means $X_{s+1} = X_s \backslash ((\cup_m [\tau_{e_m}] \cup (\cup_m U^{i_m}_{k_m}))$ where m indexes the requirements acting, $e_m = \langle i_m, j_m \rangle$.

Let h be the least e_m such that R_{e_m} acts. All τ_{e_m} acting will be subsumed into τ_h; specifically, if R_{e_m} has acted and $e_m > h$ then hereafter $\tau_{e_m} = \tau_{e_m - 1}$. Initialize all lower priority R_e, P_e ($e > h$).

(iv) Let $h =$ the least such that R_h has acted. Allow P_h to act — P_h codes the membership of 0, 1, ... ,$h-1$ in B. P_h extends τ_h by h bits to contain the current information based on B_s.

VERIFICATION:

We note that when P_h first acts, at step (iv), R_h has defined τ_h and all lower priority R_e, P_e are initialized at that stage. So τ_h gets extended by h bits, but both the old and new intervals $[\tau_h]$ have no intersection with $\overline{X_s}$. If P_h acts again at step (ii), not having been initialized, there is enough space in the resultant interval $[\tau_h]$ to redefine τ_{h+1} through to τ_s with the *same* lengths as before, and the same k values for R_{h+1} to R_s.

Lemma 10.5.23. *Each requirement R_e is initialized at most finitely often, and acts at most once.*

Proof. R_e is initialized at most e times; R_e acts at most once. $\square$

Lemma 10.5.24. $(\forall s) X_s \neq \emptyset$.

Proof. We will establish two main points: (i) when any requirement R_e acts X_s remains non-empty and (ii) when a requirement P_h acts it is possible to redefine $[\tau_{h+1}]$, $[\tau_{h+2}]$, ..., $[\tau_s]$ as subsets of both $[\tau_h]$ and X_s.

For (i) first consider the total measure of sets U_k^i used in the construction. We have $\mu([\tau_0]) = \frac{1}{2}$, and for its associated null test set $\mu(U_4^{i_0}) = \frac{1}{16}$. Also $\mu([\tau_1]) \leqslant \frac{1}{4}$; in general $\mu([\tau_n]) \leqslant 2^{-(n+1)}$ and the associated null test set satisfies $\mu(U_{k_n}^{i_n}) \leqslant 2^{-(n+4)}$. Each requirement R_e acts at most once so the total measure of null test sets removed from X_s is less than $\Sigma_{k=4}^{\infty} 2^{-k} = \frac{1}{8}$. Thus X_s is non-empty throughout the construction; next we consider the more specific intervals in which the construction occurs, $X_s \cap [\tau_e]$.

Considering $[\tau_0]$ (measure $\frac{1}{2}$) and R_0 initially, we note that if R_0 acts the construction loses at most all of $[\tau_0]$ and all of $U_4^{i_0}$ from X_s, because $X_{s+1} = X_s \cap U_4^{i_0} \cap \overline{[\tau_0]}$. Thus the new $[\tau_0]$ will have measure as small as $1 - \frac{1}{2} - \frac{1}{8} = \frac{1}{8}$ or less, but there will be intervals of positive measure to choose from which have no intersection with $U_4^{i_0}$ or *any* null test set used in the construction so far (the amount of $\frac{1}{8}$ subtracted from X_s represents $U_4^{i_0}$ and all other null test sets combined).

There will be no further action from R_0; lower priority requirements can be restarted with small null test sets. The interval $[\tau_1]$ will be no larger than half the measure of the new $[\tau_0]$, and the size of null test sets U_k^i will be reduced to fit the proportions of $[\tau_1]$, initially this means reduction by a factor of $\mu([\tau_1])/\frac{1}{2}$. When any requirement R_e acts the size of null test sets of lower priority requirements is similarly reduced by the appropriate factor to ensure that $X_s \cap [\tau_e]$ remains non-empty.

Turning to point (ii), when P_h first acts it makes sure that all lower priority intervals $[\tau_{h+j}]$, $j \in \omega$, are reduced by a factor 2^{-h}. This means there is enough space (measure) to define this sequence of intervals extending *any* of the 2^h possible strings τ_h that now include a suffix of h bits of information on B. These intervals do not need to change size if $B \upharpoonright h$ causes as τ_h change. $\square$

Lemma 10.5.25. $z \leqslant_T B$

Proof. Immediate from the simple permitting technique: once B stops changing up to n, so does z_s. $\square$

Lemma 10.5.26. $(\forall e)$ R_e *is met.*

Proof. Notice that if R'_e is met for any $e' = \langle i, j' \rangle$ then R_e is met for all $e = \langle i, j \rangle$, $j \in \omega$. We are concerned only with requirements R_e for which the associated test U^i is truly a Kurtz test. We will show that $[(\forall j) R_{\langle i,j \rangle}$ not met $] \to B \leqslant_T \emptyset'$.

Fix i. Consider $e = \langle i, j \rangle$, $j \in \omega$. We claim that if all such R_e are unmet, then there is a computable sequence of strings τ_e, $e = \langle i, j \rangle$, increasing in length and a computable sequence of stages s_e such that $B \upharpoonright |\tau_e| = B_{s_e} \upharpoonright |\tau_e|$.

R_e can be initialized only if a higher priority strategy R_f, $f < e$, acts; if this happens after R_e requires attention then R_e is met at the same time as R_f.

So if $e = \langle i_e, j_e \rangle$, consider stage s_0 when $R_{e_0} = R_{\langle i_e, 0 \rangle}$ requires attention, stage s_1 when $R_{e_1} = R_{\langle i_e, 1 \rangle}$ requires attention, and generally stage s_j when $R_{e_j} = R_{\langle i_e, 0 \rangle}$ requires attention. For all j we have $B_{s_j} \upharpoonright |\tau_{e_j}| = B \upharpoonright |\tau_{e_j}|$, as each $R_{\langle i_e, j \rangle}$, as it requires attention, becomes impossible to initialize or satisfy (by hypothesis) or even *move* as by a requirement P_h $(h < e)$ as any such P_h in acting would indicate that $R_{\langle i_e, j \rangle}$ receives permission and it satisfied. □

Lemma 10.5.27. $B \leqslant_T z$

Proof. To determine $B(x)$ find P_h in the construction coding B up to at least x that remains, throughout the construction, uninitialized. Such a requirement P_h must exist, and z can be used to eliminate contenders $P_{h'}$ that will be initialized in the future of the construction since then z will increase above the value $P_{h'}$ would code even if $B \upharpoonright h = \{0, 1, ..., h-1\}$. Once P_h has been determined $B(x)$ can be read off the appropriate bit of z (which extends the final value of τ_h). □

□

10.5.6 Kurtz randomness and hyperimmunity

We remark that in the next Chapter, we will see that each hyperimmune degree contains a Kurtz random real. Indeed, one which is not Schnorr random. On the hyperimmune-free degrees, all of the randomness notions coincide. The precise classification of the Kurtz random degrees remains open.

10.6 Decidable machines : A unifying class

Recently Bienvenu and Merkle [30] introduced a class of machines which can be used to classify most of the randomness notions met so far; in particular, Kurtz, Schnorr and Martin-Löf randomness.

Definition 10.6.1 (Bienvenu and Merkle [30]). We say that a machine M is a decidable machine if $\mathrm{dom}(M)$ is a computable set of strings.

Evidently any computable machine is decidable, as is any bounded machine. It has long been realized that a Martin-Löf test could be a computable set of strings, by taking all the extensions of σ of length s should $\sigma \in U_{n,s}$, and hence the universal Martin-Löf test can be taken as a computable set of strings. Then if we follow Schnorr's Theorem converting tests to machines, (computable tests correspond to decidable machines) it is not hard to see the following.

Theorem 10.6.2 (Bienvenu and Merkle [30]). *A real x is Martin-Löf random iff for all decidable prefix-free machines M, $K_M(x \upharpoonright n) >^+ n$. Moreover, there is a fixed decidable prefix-free machine N such that x is random iff $K_N(x \upharpoonright n) >^+ n$.*

For Schnorr randomness we have the following.

Theorem 10.6.3 (Bienvenu and Merkle [30]). *A real x is Schnorr random iff for all decidable machines M and for all (computable) orders g, we have*

$$K_M(x \upharpoonright n) \geqslant^+ n - g(n).$$

Proof. Suppose that x is not Schnorr random. Then by Theorem 10.4.4, there is a computable martingale d and a computable order g such that $d(x \upharpoonright n) \geqslant g(n)$ for infinitely many n. Let g' be a computable order with $g'(n) = o(g(n))$, taking, say $g'(n) = \log g(n)$. For each k let A_k denote the minimal set of strings σ with $d(\sigma) \geqslant 2^{k+1} g'(|\sigma|)$. Then by construction, $x \in \cap_k [A_k]$. By Kolmogorov's inequality, we have

$$2^{k+1} \sum_{\sigma \in A_k} 2^{-|\sigma| + \log(g'(|\sigma|))} = \sum_{\sigma \in A_k} 2^{-|\sigma|+k+1+\log(g'(|\sigma|))} \leqslant \sum_{\sigma \in A_k} 2^{-|\sigma|} d(\sigma) \leqslant 1.$$

Thus

$$\sum_k \sum_{\sigma \in A_k} 2^{-|\sigma| + \log(g'(|\sigma|))} \leqslant 1.$$

Therefore, we can apply KC to get a prefix-free machine M for the axioms $\langle |\sigma| - \log(g'(|\sigma|)), \sigma \rangle$. By definition of A_k, we know that $\sigma \in A_k$ implies $|\sigma| - \log(g'(|\sigma|)) \geqslant k$. Hence the KC machine is a decidable machine, and since, for all $\sigma \in A_k$, $K_M(\sigma) \leqslant |\sigma| - \log(g'(|\sigma|))$, it follows that $\exists^\infty k K_M(x \upharpoonright k) < k - \log(g'(k)) + \mathcal{O}(1)$.

For the converse, take a computable order g and a prefix- free decidable machine M such that for all k, $K_M(x \upharpoonright n) < n - g(n) - k$ for infinitely many n, and here we suppose that g is $o(n)$ without loss of generality. For each k let A_k be the set of strings σ minimal amongst those words τ with $K_M(\tau) < |\tau| - g(|\tau|) - k$. We note that $\{A_k : k \in \mathbb{N}\}$ is a uniformly computable family of prefix-free subsets of $2^{<\omega}$, such that, for all k,

$$\sum_{\sigma \in A_k} 2^{-|\sigma|+g(|\sigma|)} \leqslant \sum_{\sigma \in A_k} 2^{-K(\sigma)-k} \leqslant 2^{-k}.$$

We can then define a martingale d as follows.

$$d(\tau) = \sum_{k \in \mathbb{N}} \sum_{\sigma \in A_k} 2^{-|\sigma|+g(|\sigma|)} d_\sigma^{\frac{g(|\sigma|)}{2}}(\tau),$$

where d_u^p is the normed martingale which plays the doubling strategy on σ except for the first p bits:

$d_u^p(\sigma) = 1$, if $|\sigma| \leqslant p$, $d^p(\sigma) = 2^{\min(|\sigma|,|u|)}$, if $p < |\sigma|$ and $\sigma_p \ldots \sigma_{\min(|\sigma|,|u|)-1} = u_p \ldots u_{\min(|\sigma|,|u|)-1}$, and $d_u^p(\sigma) = 0$ otherwise.

Then as d is a weighted sum of martingales, it is a martingale. Note that

$$d(\lambda) = \sum_{k \in \mathbb{N}} \sum_{\sigma \in A_k} 2^{-|\sigma|+g(|\sigma|)},$$

and since $\sum_{\sigma \in A_k} 2^{-|\sigma|+g(|\sigma|)} \leqslant 2^{-k}$ we can computably approximate $d(\lambda)$ which is thus a computable real. For an induction, to if we can compute $d(\tau)$ to compute $d(\tau 0)$, say, we see that

$$d(\tau 0) - d(\tau) = \sum_{k \in \mathbb{N}} \sum_{\sigma \in A_k} 2^{-|\sigma|+g(|\sigma|)} (d_\sigma^{\frac{g(|\sigma|)}{2}}(\tau 0) - d_\sigma^{\frac{g(|\sigma|)}{2}}(\tau)).$$

This is a finite sum as $d_\sigma^{\frac{g(|\sigma|)}{2}}(\tau) = 1$ for all σ with $g(|\sigma|) \geqslant 2|\tau|$. To finish the proof note that if $w \in \cup_{k \in \mathbb{N}} A_k$,

$$d(w) \geqslant 2^{-|w|+g(|w|)} d_w^{\frac{g(|w|)}{2}}(w) \geqslant 2^{\frac{g(|w|)}{2}}.$$

But $2^{\frac{g(|w|)}{2}}$ is an order, and there are infinitely many prefixes of x in $\cup_{k \in \mathbb{N}} A_k$. Thus x is not Schnorr random by theorem 10.4.4. □

For Kurtz randomness we have the following.

Theorem 10.6.4 (Bienvenu and Merkle [30]). *The following are equivalent for a real x.*

(i) x is not Kurtz random.

(ii) There exists a decidable prefix-free machine M and a computable order $f : \omega \to \omega$ such that for all n,

$$K_M(x \upharpoonright f(n)) \leqslant f(n) - n.$$

(iii) There exists a decidable machine M and a computable order $f : \omega \to \omega$ such that, for all n,

$$K_M(x \upharpoonright f(n)) \leqslant f(n) - n.$$

Proof. (i) implies (ii) and (iii). To see this, by Theorem 10.5.15 A real x is *not* Kurtz random iff there is a computable machine M and a computable increasing function $f : \omega \to \omega$ such that:

$$(\forall d)(K_M(x \upharpoonright f(d)) < f(d) - d).$$

This implies (ii) and (iii). To see that (iii) implies (i), let $B_n = \{\sigma : |\sigma| = h(2n) \wedge C_M(\sigma) \leqslant |\sigma| - 2n\}$. Define the martingale

$$d(w) = \sum_{n \in \mathbb{N}} \sum_{\sigma \in B_n} 2^{-f(2n)+n} d_\sigma(w).$$

For all n, $\sum_{\sigma \in B_n} d_\sigma(w) \leqslant |B_n| 2^{|w|} \leqslant 2^{f(2n)-2n+|w|}$. Hence for all n, $\sum_{\sigma \in B_n} d_\sigma(w) \leqslant 2^{-n+|w|}$, and hence d is computable. Also for all n, $x \upharpoonright f(2n) \in B_n$ and therefore for all $m \geqslant f(2n)$, we have

$$\sum_{\sigma \in B_n} 2^{-f(2n)+n} d_\sigma(x \upharpoonright m) \geqslant 2^n.$$

Thus we need only set $g(n) = 2^{f^{-1}(2n)}$, to have an order showing that x is not Kurtz random. □

Using similar methods, Bienvenu and Merkle also obtained the following characterization of Kurtz randomness.

Theorem 10.6.5 (Bienvenu and Merkle [30])**.** *The following are equivalent for a real x.*

(i) x is not Kurtz random.

(ii) There exists a decidable prefix-free machine M and computable order h such that for all n,

$$K_M(x \upharpoonright n) \leqslant n - h(n).$$

(iii) There exists a decidable machine M and computable order h such that, for all n,

$$K_M(x \upharpoonright n) \leqslant n - h(n).$$

11
Randomness in Cantor space and Turing reducibility

In this chapter, we look at the distribution of 1-random (and n-random) sets among the Turing (and other) degrees.

11.1 Π^0_1 classes of random sets

For each constant c, the class $\{x : \forall n\, K(x \upharpoonright n) \geqslant n-c\}$ is clearly a Π^0_1 class, so basic facts about Π^0_1 classes gives us several interesting results about the 1-random sets.

Proposition 11.1.1 (Kučera)**.** *The collection of* 1*-random sets is a* Σ^0_2 *class.*

Proposition 11.1.2. *There exist low* 1*-random sets.*

Proof. Apply the Low Basis Theorem (Theorem 5.16.7). □

We have already seen the following result in the proof of Corollary 9.10.2, which states that there is a 1-random degree $\mathbf{a}$ such that every nonzero degree below $\mathbf{a}$ is also 1-random.

Proposition 11.1.3. *There exist* 1*-random sets of hyperimmune-free degree.*

Proof. Apply the Hyperimmune-Free Basis Theorem (Theorem 5.16.9). □

Proposition 11.1.2 is obviously particular to 1-randomness, since no 2-random set can be Δ^0_2, let alone low. In Corollary 11.16.20, we will see that this is also the case for Proposition 11.1.3, since every 2-random set has hyperimmune degree.

11.2 Computably enumerable degrees

Kučera [155] completely answered the question of which c.e. degrees contain 1-random sets, by showing that the only such degree is the complete one.

Theorem 11.2.1 (Kučera [155])**.** *If A is 1-random, B is a c.e. set, and $A \leqslant_{\mathrm{T}} B$, then $B \equiv_{\mathrm{T}} \emptyset'$. In particular, if A is 1-random and has c.e. degree, then $A \equiv_{\mathrm{T}} \emptyset'$.*

Proof. Kučera's original proof used Arslanov's Completeness Criterion (Theorem 5.19.3). We give a new direct proof.

Suppose that A is 1-random, B is c.e., and $\Psi^B = A$. Let $\{A_s\}_{s\in\omega}$ and $\{B_s\}_{s\in\omega}$ be computable approximations to A and B, respectively. Let c be such that $K(A \restriction n) > n - c$ for all n. We will enumerate a KC set, with constant d given by the Recursion Theorem. Let $k = c+d+1$. We want to ensure that if $B \restriction \psi_s^B(n+k) = B_s \restriction \psi_s^B(n+k)$, then $\emptyset'_s(n) = \emptyset'(n)$, which clearly ensures that $\emptyset' \leqslant_{\mathrm{T}} B$. We can assume that our approximations are sufficiently sped-up so that $\Psi_s^B(n+k)\downarrow$ for all s and all $n \leqslant s$.

If n enters $\emptyset'$ at some stage $s \geqslant n$, we wish to change B below $\psi_s^B(n+k)$. We do this by forcing A to change below $n+k$. That is, we enumerate a KC request $\langle n+1, A_s \restriction n+k\rangle$. This request ensures that $K(A_s \restriction n+k) \leqslant n+d+1 = n+k-c$, and hence that $A \restriction n+k \neq A_s \restriction n+k$. Thus $B \restriction \psi_s^B(n+k) \neq B_s \restriction \psi_s^B(n+k)$, as required. (Note that we enumerate at most one request, of weight $2^{-(n+1)}$, for each $n \in \omega$, and hence our requests indeed form a KC set.) $\square$

In the above proof, if Ψ is a wtt-reduction, then so is the reduction from B to $\emptyset'$ that we build. Hence, we have the following result.

Corollary 11.2.2. *If A is 1-random, B is a c.e. set, and $A \leqslant_{\mathrm{wtt}} B$, then $B \equiv_{\mathrm{wtt}} \emptyset'$. In particular, if A is 1-random and has c.e. wtt-degree, then $A \equiv_{\mathrm{wtt}} \emptyset'$.*

11.3 The Kučera-Gács Theorem

In this section we present a basic result, proved independently by Kučera [155] and Gács [117], about the distribution of 1-random sets and the relationship between 1-randomness and Turing reducibility. Specifically, we show that every set is computable from some 1-random set. We begin with an auxiliary result.

Lemma 11.3.1 (Space Lemma, see Merkle and Mihailović [204]). *Given a rational $\delta > 1$ and $k > 0$, we can compute a length $l(\delta, k)$ such that, for any martingale d and any σ,*

$$|\{\tau \in 2^{l(\delta,k)} : d(\sigma\tau) \leqslant \delta d(\sigma)\}| \geqslant k.$$

Proof. By Kolmogorov's Inequality (Theorem 10.1.4), for any l and σ, the average of $d(\sigma\tau)$ over strings τ of length l is $d(\sigma)$. Thus

$$\frac{|\{|\tau| = l : d(\sigma\tau) > \delta d(\sigma)\}|}{2^l} < \frac{1}{\delta}.$$

Let $l(\delta, k) = \lceil \log \frac{k}{1-\delta^{-1}} \rceil$, which is well-defined since $\delta > 1$. Then

$$|\{|\tau| = l(\delta, k) : d(\sigma\tau) > \delta d(\sigma)\}| < \frac{2^{l(\delta,k)}}{\delta},$$

so

$$|\{\tau \in 2^{l(\delta,k)} : d(\sigma\tau) \leqslant \delta d(\sigma)\}| > 2^{l(\delta,k)} - \frac{2^{l(\delta,k)}}{\delta} = 2^{l(\delta,k)}(1 - \delta^{-1}) \geqslant \frac{k}{1-\delta^{-1}}(1 - \delta^{-1}) = k.$$

□

One should think of the Space Lemma as saying that, for any σ and any martingale d, there are at least k many extensions τ of σ of length $l(\delta, k)$ such that d cannot increase its capital at σ by more than a factor of δ while betting along τ.

We wish to show that a given set X can be coded into a 1-random set R. Clearly, unless X itself is random, there is no hope of doing such a coding unless the reduction does not allow for recovery of X as an identifiable subsequence of R. For instance, it would be hopeless to try to ensure that $X \leqslant_{\mathrm{m}} R$. As we will show, however, we can get $X \leqslant_{\mathrm{wtt}} R$. The key idea in the proof below, which we take from Merkle and Mihailović [204], is to use the intervals provided by the Space Lemma to do the coding.

Theorem 11.3.2 (Kučera [155], Gács [117]). *Every set is wtt-reducible to a 1-random set.*

Proof. Let d be a universal c.e. martingale. Let $r_0 > r_1 > \cdots$ be a collection of positive rationals such that, letting

$$\beta_i = \prod_{j \leqslant i} r_j,$$

the sequence $\{\beta_i\}_{i\in\mathbb{N}}$ converges to some value β. Let $l_s := l(r_s, 2)$ be as in the Space Lemma, which means that for any σ there are at least two words τ of length l_s with $d(\sigma\tau) \leqslant r_s d(\sigma)$. Partition $\mathbb{N}$ into consecutive intervals $\{I_s\}_{s\in\mathbb{N}}$ with $|I_s| = l_s$.

Fix a set X. We construct a 1-random set R that wtt-computes X. At stage s, we specify R on the elements of I_s. We denote the part of R specified before stage s by σ_s. (That is, $\sigma_s = R \upharpoonright \sum_{i<s} l_i$.) If $s > 0$, then we assume by induction that $d(\sigma_s) \leqslant \beta_{s-1}$. We say that a string τ of length l_s is *s-admissible* if $d(\sigma_s\tau) \leqslant \beta_s$. Since $\beta_s = r_s\beta_{s-1}$ (when $s > 0$) and $l_s = l(r_s, 2)$, there are at least two s-admissible strings. Let τ_0 and τ_1 be the lexicographically least and greatest among such strings, respectively. Let $\sigma_{s+1} = \tau_i$, where $i = X(s)$.

Now $\liminf_n d(R \upharpoonright n) \leqslant \beta$, so R is 1-random. We now show how to compute $X(s)$ from $\sigma_{s+1} = R \upharpoonright \sum_{i\leqslant s} l_i$. We know that σ_{s+1} is either the leftmost or the rightmost s-admissible extension of σ_s, and being s-admissible is clearly a co-c.e. property, so we wait until either all extensions of σ to the left of σ_{s+1} are seen to be not s-admissible, or all extensions of σ to the right of σ_{s+1} are seen to be not s-admissible. In the first case, $s \notin X$, while in the second case, $s \in X$. □

11.4 Kučera coding

One of the most basic questions we might ask about the distribution of 1-random sets is which Turing degrees contain 1-random sets. Kučera [155] gave the following partial answer, which was later extended by Kautz [140] to n-random sets using the jump operator, as we will see in Theorem 11.10.8.

Theorem 11.4.1 (Kučera [155])**.** *If* $\mathbf{a} \geqslant \mathbf{0}'$ *then* $\mathbf{a}$ *contains a* 1*-random set.*

Proof. Let $\mathbf{a} \geqslant \mathbf{0}'$ and let $X \in \mathbf{a}$. Let R be as in the proof of Theorem 11.3.2. Then $X \leqslant_T R$. Furthermore, we can compute R if we know X and can tell which sets are s-admissible for each s. The latter question can be answered by $\emptyset'$. Since $X \geqslant_T \emptyset'$, it follows that $R \leqslant_T X$. Thus R is a 1-random set of degree $\mathbf{a}$. □

We now give a different proof of Theorem 11.4.1, along the lines of the original proof of Kučera [155]. This version is due to Jan Reimann [244] (Also see Reimann and Slaman [246]). We will use some of its ideas when we consider FPF degrees later. We begin with Kučera's useful construction of a universal Martin-Löf test.

Kučera's universal Martin-Löf test. For this construction, we think of the elements of the W_i as strings.

Fix $n \in \mathbb{N}$. For each $e > n$, enumerate all elements of $W_{\Phi_e(e)}$ into a set U_n (where we take $W_{\Phi_e(e)}$ to be empty if $\Phi_e(e)\uparrow$) as long as

$$\sum_{\sigma \in W_{\Phi_e(e)}} 2^{-|\sigma|} < 2^{-e}.$$

Then the U_n are uniformly c.e. and

$$\sum_{\sigma \in U_n} 2^{-|\sigma|} \leqslant \sum_{e>n} 2^{-e} = 2^{-n}.$$

Thus $\{U_n\}_{n\in\mathbb{N}}$ is a Martin-Löf test. To see that it is universal, let $\{V_n\}_{n\in\mathbb{N}}$ be a Martin-Löf test. Let e be such that $V_i = W_{\Phi(e)(i)}$ for all i. Every computable function possesses infinitely many indices, so for each n there is an $i > n$ such that $\Phi_e = \Phi_i$. For such an i, we have $W_{\Phi_i(i)} = W_{\Phi_e(i)} = V_i$, which means that every element of V_i is enumerated into U_n. Thus $\bigcap_m V_m \subseteq U_n$. So $\bigcap_m V_m \subseteq \bigcap_n U_n$. Since $\{V_n\}_{n\in\mathbb{N}}$ is an arbitrary Martin-Löf test, $\{U_n\}_{n\in\mathbb{N}}$ is universal.

We next need a lemma similar in spirit to the Space Lemma of the previous section.

Lemma 11.4.2. *Let $\{U_n\}_{n\in\mathbb{N}}$ be Kučera's universal Martin-Löf test, and let P_n be the complement of U_n. Let C be a Π^0_1 class. Then there exists a computable function $\gamma : 2^{<\omega} \times \mathbb{N} \to \mathbb{Q}^+$ such that for any $\sigma \in 2^{<\omega}$ and $n \in \mathbb{N}$,*

$$P_n \cap C \cap [\sigma] \neq \emptyset \implies \mu(C \cap [\sigma]) \geqslant \gamma(\sigma, n).$$

Proof. It is easy to check that if $P_n \cap C \cap [\sigma] \neq \emptyset$ then $\mu(C \cap [\sigma]) > 0$, but we need to show that we can give an *effective* positive lower bound for $\mu(C \cap [\sigma])$.

Since C is a Π^0_1 class, there is an index k such that $\overline{C} = \bigcup_{\tau\in W_k}[\tau]$. Fix σ and n, and define the partial computable function Φ as follows. On input j, search for an s such that

$$\mu([\sigma] - \bigcup_{\tau\in W_k[s]} [\tau]) < 2^{-j}.$$

If such an s is ever found, then let $\Phi(j)$ be such that $V_{\Phi(j)}$ is a finite cover of $[\sigma] - \bigcup_{\tau\in W_k[s]}[\tau]$ of measure at most 2^{-j}. Clearly, $\{V_{\Phi(j)}\}_{j\in\mathbb{N}}$ is a Martin-Löf test. Let $e > n$ be an index such that $\Phi_e = \Phi$, obtained effectively from σ and n. Let $\gamma(\sigma, n) = 2^{-e}$.

Suppose that $\Phi_e(e)\downarrow$. Then $V_{\Phi_e(e)} \subseteq U_n$, so there is an s such that $[\sigma] - \bigcup_{\tau\in W_k[s]}[\tau] \in U_n$. But

$$C \cap [\sigma] = [\sigma] - \bigcup_{\tau\in W_k} \subseteq [\sigma] - \bigcup_{\tau\in W_k[s]} [\tau],$$

so $C \cap [\sigma] \in U_n$, and hence $P_n \cap C \cap [\sigma] = \emptyset$.

Now suppose that $\Phi_e(e)\uparrow$. Then for every s,

$$\mu([\sigma] - \bigcup_{\tau\in W_k[s]} [\tau]) \geqslant 2^{-e},$$

and hence

$$\mu(C \cap [\sigma]) = \mu([\sigma] - \bigcup_{\tau \in W_k} [\tau]) = \lim_s \mu([\sigma] - \bigcup_{\tau \in W_k[s]} [\tau]) \geqslant 2^{-e} = \gamma(\sigma, n).$$

□

We can now give an alternate proof of Theorem 11.4.1. Let $B \geqslant_{\mathrm{T}} \emptyset'$. We need to show that there is a 1-random set A such that $A \equiv_{\mathrm{T}} B$. Let $\{U_n\}_{n\in\mathbb{N}}$ be Kučera's universal Martin-Löf test. We describe how to code B into an element A of $\overline{U_0}$. Let T be a computable tree such that $[T] = \overline{U_0}$ and let E be the set of extendible nodes of T. Note that E is co-c.e., and hence $\emptyset'$-computable.

Let γ be as in Lemma 11.4.2 with $C = \overline{U_0}$, and let $b(\sigma) = \lceil -\log \gamma(\sigma, 0) \rceil$. Then we know that, for any σ, if $\overline{U_0} \cup [\sigma] \neq \emptyset$, then $\mu(\overline{U_0} \cup [\sigma]) \geqslant 2^{-b(\sigma)}$. Thus, if $\sigma \in E$ then σ has at least two extensions of length $b(\sigma)+1$ in E.

Suppose that, at stage n, we have defined $A \upharpoonright m_n$ for some m_n. Let $\sigma = A \upharpoonright m_n$. We assume by induction that $\sigma \in E$. Let τ_0 and τ_1 be the leftmost and rightmost extension of σ of length $b(\sigma)+1$ in E, respectively. As pointed out in the previous paragraph, $\tau_0 \neq \tau_1$. Let $A \upharpoonright b(\sigma) = \tau_i$, where $i = B(n)$.

Since $A \in [T] = \overline{U_0}$, we know that A is 1-random. We now show that $A \equiv_{\mathrm{T}} B$.

The construction of A is computable in B and $\emptyset'$. Since we are assuming that $B \geqslant_{\mathrm{T}} \emptyset'$, we have $A \leqslant_{\mathrm{T}} B$. For the other direction, first note that the function $n \mapsto m_n$ is computable in A. To see that this is the case, assume that we have already computed m_n. Then $m_{n+1} = b(A \upharpoonright m_n) + 1$. Now, to compute $B(n)$ using A, let $\sigma = A \upharpoonright m_n$ and let $\tau = A \upharpoonright m_{n+1}$. We know that τ is either the leftmost or the rightmost extension of σ of length m_{n+1} in E. Since E is co-c.e., we can wait until either all extensions of σ of length m_{n+1} to the left of τ leave E, or all extensions of σ of length m_{n+1} to the right of τ leave E. In the first case, $s \notin B$, while in the second case, $s \in B$.

11.4.1 A proof of the Reimann-Slaman Theorem

We are now in a position to prove Theorem 9.9.11 which was the following:

Theorem 11.4.3 (Reimann and Slaman [246])**.** *Suppose that x is a non-computable real. Then there is a computable measure λ such that x is random relative to λ.*

The following proof is drawn from Reimann and Slaman [246]. Relativizing Theorem 11.4.1, we know that

If B is a real with $C' \leqslant_T B \oplus C$ then $B \oplus C \equiv_T X \oplus C$ for some C-random real X.

Reimann and Slaman then used the following basis theorem.

Theorem 11.4.4 (Reimann and Slaman [246]). *Suppose that $C \in 2^\omega$ and $T \subseteq \omega^{<\omega}$ is a an infinite C-computable tree withe has finite C-computable branching width. Then for each C-random real A, there is a path X of T with A $C \oplus X$-random.*

Proof. Given C and $\tau \in \omega^{<\omega}$, let $\{U_n^{C,\tau} : n \in \mathbb{N}\}$ denote the universal Martin-Löf test relative to C, τ, enumerable in C. Then we enumerate a C-Martin-Löf test $\{V_n : n \in \mathbb{N}\}$ as follows: Put $[\sigma]$ into V_n if $[\sigma]$ is contained in $U_n^{C,\tau}$ for all $\tau \in T$ with $|\tau| = |\sigma|$. As A is C-random, there exists an n with $A \notin V_n$. Therefore for all m, $[A \upharpoonright m] \notin V_n$ and hence there is $\tau \in T$ of length m with $[\tau] \notin U_n^{C,\tau}$. Therefore there is an infinite subtree of T of nodes τ which don't enumerate any $[\eta]$ into $U_n^{C,\tau}$ for any $\eta \prec A$. By König's Lemma, there is an infinite path X through this subtree. Finally A is $C \oplus X$-random. Otherwise, by the use principle, for each n there would be a an initial segment τ of X such that $A \in U_n^{C,\tau}$, a contradiction. $\square$

Proof. (of Theorem 9.9.11) Suppose that A is noncomputable. Using the relativized version of Theorem 11.4.1, and the Posner-Robinson Join Theorem, (Theorem 5.13.3), there is a C-random real X with $X \oplus C \equiv_T A \oplus C$. Therefore there are C-computable functionals Ψ and Φ with

$$\Phi(X) = A \text{ and } \Psi(A) = X.$$

The idea is to use these functionals to define a class of measures relative to which A is random. For $\sigma \in 2^{<\omega}$, let

$$\mathrm{pre}(\sigma) = \{\tau : \tau \text{ shortest with } \sigma \preccurlyeq \Phi(\tau) \wedge \tau \preccurlyeq \Psi(\sigma)\}.$$

We wish to define a measure λ relative to which A is random. To do this we make sure that λ will dominate an umage measure induced by Φ, thereby making sure that a 1-random real is mapped to a λ-random one, and secondly making sure that λ is not atomic. To do this, we will restrict the values of λ so that

$$\mu([\mathrm{pre}(\sigma)]) \leqslant \lambda([\sigma]) \leqslant \mu([\Psi(\sigma)]).$$

The first inequality ensures domination and the second ensured that λ is not atomic on the doman of Ψ.

The next step is to show that there is a Π_1^C class each element of which represents a measure relative to which A is random. Note that the restriction on $\mathrm{pre}(\sigma)$ above is Σ_1^0 and it kills off branches of the computably bounded tree of measures T_M. Because of the c.e. nature of the restrcition, there will be a computable subtree consisting of admissable measures, and we can generate the Π_1^0 class M of paths through this tree.

Notice that this class is nonempty. To see this, as $\Psi(A) = X$ there are nfinitely many $\sigma_n = A \upharpoonright \ell_n$ such that $\Psi(\sigma_n) \prec X$ and $|\Psi(\sigma_n)| \to \infty$. Furthermore, for each n, $[\mathrm{pre}(\sigma_n)] \subseteq [\Psi(\sigma_n)]$, and $\mathrm{pre}(\sigma_n)$ is not empty for almost all n. Since $\mu([\Psi(\sigma_n)]) \to 0$, and since $\mathrm{pre}(\sigma_n)] \subseteq [\Psi(\sigma_n)]$, there exists a set of infinitely many mutially different compatible requirement of the form $\mu([\mathrm{pre}(\sigma)]) \leqslant \lambda([\sigma]) \leqslant \mu([\Psi(\sigma])$ which can be enumerated into the Σ_1^0 class described above. Thereofre $M \neq \emptyset$. Also it follows by this reasoning for any λ encoded by M, $\lambda(\{A\}) = 0.$, and hence A is not a λ-atom.

By the basis theorem, Theorem 11.4.4, there is a λ encoded by a path of M such that X is $C \oplus \lambda$-random.

To comlee the proof, we need to show that A is λ-random. For the sake of a cntradiction, suppose that there was a λ-ML test $\{V_n : n \in \mathbb{N}\}$ covering A. There must be infinitely many $\sigma_n = A \restriction \ell_n$ such that $\sigma_n \imath V_n$ for all n. We ca then define a new test $\{U_n : n \in \mathbb{N}\}$ by putting $\mathrm{pre}(\sigma)$ for each $\sigma \in V_n$. By definition of pre, the U_n covers X. Furtehrmore since λ satsfies $\mu([\mathrm{pre}(\sigma)]) \leqslant \lambda([\sigma]) \leqslant \mu([\Psi(\sigma])$, the Lebesgue measure of U_n is bounded by $\lambda(V_n)$. Thus $\{U_n : n \in \mathbb{N}\}$ is a $C \oplus \lambda$-ML test, a contradiction, since it covers X. □

11.5 Randomness and PA degrees

In light of the results of the previous two sections, it is natural to ask whether the collection of 1-random degrees is closed upwards. In this section we give a negative answer to this question by examining the relationship between randomness and the PA degrees introduced in Section 5.18, a connection first explored by Kučera [155].

As we saw in Section 5.18, one of the characterizations of the PA degrees (which we can take as a definition), is that **a** is PA iff each partial computable $\{0,1\}$-valued function has an **a**-computable total extension. As we saw in Section 5.19, this fact implies that if **a** is PA then it computes a diagonally concomputable (DNC) function, that is, there is a total function g such that $g(n) \neq \Phi_n(n)$ for all n. The same is true of 1-random degrees.

Lemma 11.5.1 (Kučera [155]). *Suppose that A is 1-random. Then A computes a DNC function.*

Proof. Let $f(n)$ be the number of $A \restriction n$ in some effective listing of finite binary strings. Since A is 1-random, $K(f(n)) = K(A \restriction n) \pm O(1) \geqslant n - O(1)$. On the other hand, if $\Phi_n(n)\downarrow$, then $K(\Phi(n)) \leqslant 2\log n + O(1)$, so there are at most finitely many n such that $f(n) = \Phi_n(n)$. By altering f at these finitely many places, we obtain an A-computable DNC function. □

A further similarity between PA degrees and 1-random degrees is that, since there are Π_1^0 classes consisting entirely of 1-random sets (as we saw in Section 11.1), every PA degree computes a 1-random set. On the other hand, Kučera [155] showed that the class of sets of PA degree has measure 0, and hence there are 1-random degrees that are not PA. He also showed that there are PA degrees that are not 1-random, and hence the collection of 1-random degrees is not closed upwards. The exact relationship between

PA degrees and 1-random degrees was clarified by the following remarkable result of Stephan [292].

Theorem 11.5.2 (Stephan [292])**.** *If* $\mathbf{a}$ *is both PA and* 1*-random then* $\mathbf{a} \geqslant \mathbf{0}'$.

Proof. Let $A \not\geqslant_{\mathrm{T}} \emptyset'$ have PA degree. We first construct a partial computable function ψ such that, for each e, the class of all B such that Φ_e^B is a total extension of ψ has small measure. Then we use ψ to show that A is not 1-random, using the fact that, since A has PA degree, A can compute a total extension of ψ.

We proceed as follows for all e simultaneously. Let $I_e = [2^{e+2}, 2^{e+3} - 1]$. Within I_e, define ψ as follows. At stage s, let a_s be the least element of I_e such that $\psi(a_s)$ has not yet been defined. (We will show below that such a number always exists.) For $i = 0, 1$, let $P_{e,s,i}$ be the Δ_1^0 class of all B such that, for all $n \in I_e$,

1. $\Phi_e^B(n)[s]\downarrow$ (and hence has use less than or equal to s),
2. if $n < a_s$ the $\Phi_e^B(n) = \psi(n)$, and
3. if $n = a_s$ then $\Phi_e^B(n) = i$.

Let $d_{e,s,i} = \mu(P_{e,s,i})$. Note that $d_{e,s,i}$ is a rational number, and that we can compute this number.

If $d_{e,s,0} + d_{e,s,1} > 2^{-e-1}$, then choose $i \in \{0, 1\}$ such that $d_{e,s,i} \leqslant d_{e,s,1-i}$ and let $\psi(a_s) = i$. Otherwise, do nothing at this stage.

Clearly, ψ is partial computable. If s is a stage at which ψ is defined on a new input in I_e, then at stage s we ensure that Φ_e^B does not extend ψ for all B in a class C_s of measure at least 2^{-e-2}. Furthermore if $s \neq t$ then $C_s \cap C_t = \emptyset$, so we cannot define ψ on all of I_e. Thus there is a stage t such that $d_{e,s,0} + d_{e,s,1} \leqslant 2^{-e-1}$ for all $s > t$.

Since A is PA, A can compute a total $\{0,1\}$ valued extension Ψ of ψ. Choose an infinite collection $\varphi_{e_0}, \varphi_{e_1}, \dots$ $e_i \leqslant e_{i+1}$ all with $\varphi_{e_i}^A = \Psi$, and let $k_0, k_1, \dots$ be a 1-1 enumeration of the halting set K.

Let $e(s) = e_{k_s}$, and $r(s)$ be the first stage $t > s$ where $P_{e(s),t,0} \cup P_{e(s),t,1}$ has at most measure $2^{-e(s)-1}$. Noting that membership of B in $P_{e(s),t,0} \cup P_{e(s),t,1}$ depends only on values of B up to $r(s)$, means that this class is a Σ_1^0 class. We can define

$$U_n = \cup_{\{s:e(s) \geqslant n\}} P_{e(s),t,0} \cup P_{e(s),t,1}.$$

The measure of U_n is bounded by $\sum_{e(s) \geqslant n} 2^{-e(s)-1}$. Hence, since the mappings $s \mapsto k_s$ and $i \mapsto e_i$ are injective, $s \mapsto e(s)$ is injective also, and hence $\mu(U_n) \leqslant 2^{-n}$. Thus the $\{U_n : n \in \mathbb{N}\}$ is a Martin-Löf test.

The function $f(p) = \max\{u(A, x, e_p) : x \in I_{e_p}\}$ is A-computable. Since $A \not\leqslant_{\mathrm{T}} \emptyset'$, there are infinitely many $p \in K$ such that $f(p) < s$ where $p = k_s$.

It follows that for these e, p and s and all $x \in I_{e_p}$,

$$\varphi_{e_p}^A(x) \downarrow \wedge u(A, e_p, x) \leqslant r(s).$$

Hence $A \in P_{e(s),t,0} \cup P_{e(s),t,1}$. Since A is in infinitely many of these classes, it is not Solovay random and hence not 1-random. □

Corollary 11.5.3 (Kučera [155]). *There are PA degrees that are not* 1-*random.*

Proof. Apply Theorem 11.5.2 to a low PA degree. □

Corollary 11.5.4 (Kučera [155]). *The collection of* 1-*random degrees is not closed upwards.*

Proof. As mentioned above, every PA degree computes a 1-random set. Now apply the previous corollary. □

Stephan [**?**] noted the following improvement of this result.

Corollary 11.5.5 (Stephan [292]). *Let* $\mathbf{a} \not\geqslant \emptyset'$. *Then there is a degree* $\mathbf{b} \geqslant \mathbf{a}$ *that is not* 1-*random.*

Proof. It is easy to check, since $\mathbf{a} \not\geqslant \emptyset'$, there is no $\mathbf{a}$-computable function that majorizes the modulus function $c_{\emptyset'}$ of (a fixed enumeration of) $\emptyset'$. By the relativized Hyperimmune-Free Basis Theorem, there is a PA degree $\mathbf{b} \geqslant \mathbf{a}$ that is hyperimmune-free relative to $\mathbf{a}$, which means that every $\mathbf{b}$-computable function is majorized by a $\mathbf{a}$-computable function. Thus $c_{\emptyset'}$ is not $\mathbf{b}$-computable, and hence $\mathbf{b} \not\geqslant \mathbf{0}'$. Since $\mathbf{b}$ is PA, it cannot be 1-random. □

Stephan[292] has the following interesting conclusion regarding Theorem 11.5.2: "[Theorem 11.5.2] says that there are two types of Martin-Löf sets: the first type are the computationally powerful sets which permit the solving of the halting problem; the second type of random set are computationally weak in the sense that they are not [PA]. Every set not belonging to one of these two classes is not Martin-Löf random."

We could also note that since no n-random set for $n \geqslant 2$ is above $\emptyset'$ it follows that while the n-random sets are incompressible and hence complicated, they are computationally weak in the sense above. We will see more reflections of this in later material when we consider things like van Lambalgen reducibility.

11.6 Independence results

Van Lambalgen and others looked at the extent to which independence and other properties held of subsequences. The intuition is that no piece of information for a random real should be able to help elsewhere.

Theorem 11.6.1 (van Lambalgen, Kautz). *(i) If $A \oplus B$ is n-random so is A.*

(ii) If A is n-random so is $A^{[n]}$, the n-th column of A.

Proof. (i) For simplicity let $n = 1$. If A is not random, then $A \in [\sigma]$ for infinitely many $[\sigma]$ in some Solovay test V. Then $A \oplus B$ would be in $\widehat{V}$, where $[\sigma \oplus \tau] \in \widehat{V}$ for all τ with $|\tau| = |\sigma|$ and $\sigma \in V$. The measure of $\widehat{V}$ is the same as V. (ii) is similar. □

A stronger form of Theorem 11.6.1 result is the following due to van Lambalgen. The proof is more or less the same.

Theorem 11.6.2 (van Lambalgen [314]). *If $A \oplus B$ is n-random, then A is $n - B-$random.*

Proof. Again let $n = 1$ and then relativize. Suppose B is not random over A. We have $B \in \bigcap_n V_n^A$ where V_n^A is uniformly $\Sigma_1^{0,A}$ of measure $\leqslant 1/2^n$. Put

$$W_n = \{X \oplus Y \mid X \in 2^\omega, Y \in \widetilde{V}_n^X\}$$

where $\widetilde{V}_n^X$ is V_n^X enumerated so long as its measure is $\leqslant 1/2^n$. Note that W_n is uniformly Σ_1^0 of measure $\leqslant 1/2^n$. Moreover $\widetilde{V}_n^A = V_n^A$, hence $A \oplus B \in \bigcap_n W_n$, contradicting the assumption that $A \oplus B$ is random. □

Another result known to Kučera and van Lambalgen is the following

Theorem 11.6.3 (Kučera, van Lambalgen). *If $A \oplus B$ is random, then $A \nleqslant_{\mathrm{T}} B$.*

Proof. Suppose that $A = \Phi^B$. We cover $A \oplus B$. Put into V_e any string $\sigma \oplus \tau$, $|\sigma| = e$ and $\sigma = \Phi^\tau$, (note that, by convention, $|\tau| \geqslant |\sigma|$). Now note that $\mu(V_e) \leqslant 2^{-e}$, and $A \oplus B \in \cap_e V_e$. □

Corollary 11.6.4 (Kurtz [165]). *No random real has minimal degree.*

The converse of Theorem 11.6.2 is also true. The next result will prove to be extremely important when we attempt to understand things like $\leqslant_C$ and $\leqslant_K$, and should be seen as a *central* result in algorithmic information theory.

Theorem 11.6.5 (van Lambalgen [315]). *If A n-random and B is $n - A-$random, then $A \oplus B$ is n-random.*

Proof. Again we specialize to $n = 1$. Then we would relativize for higher n. Suppose $A \oplus B$ is not random. We have $A \oplus B \in \bigcap_n W_n$ where W_n is uniformly Σ_1^0 with $\mu(W_n) \leqslant 1/2^n$. By passing to a subsequence we may assume that $\mu(W_n) \leqslant 1/2^{2n}$.

Put

$$U_n = \{X \mid \mu(\{Y \mid X \oplus Y \in W_n\}) > 1/2^n\}.$$

Note that U_n is uniformly Σ_1^0. Moreover $\mu(U_n) \leqslant 1/2^n$ for all n, because otherwise we would have

$$\mu(W_n) > \mu(U_n) \cdot \frac{1}{2^n} > \frac{1}{2^n} \cdot \frac{1}{2^n} = \frac{1}{2^{2n}},$$

a contradiction.

Since A is random, it follows that $\{n \mid A \in U_n\}$ is finite. Thus for all but finitely many n we have $A \notin U_n$, i.e.,

$$\mu(\{Y \mid A \oplus Y \in W_n\}) \leqslant 1/2^n.$$

Put $V_n^A = \{Y \mid A \oplus Y \in W_n\}$. Then $\mu(V_n^A) \leqslant 1/2^n$ for all but finitely many n, and V_n^A is uniformly $\Sigma_1^{0,A}$. Moreover $B \in \bigcap_n V_n^A$, contradicting the assumption that B is random over A. □

We will use this result extensively in Chapter 14, where we look at the K-degrees of random reals. Indeed, we base a reducibility on the result: To wit, we will define *van Lambalgen* reducibility by saying that $A \leqslant_{vL} B$ iff for all X, $A \oplus X$ is random if $B \oplus X$ is random. As we see in Chapter 14, $\leqslant_K$ implies $\leqslant_{vL}$. As an example of the use of van Lambalgen's Theorem, we give a very short and elegant proof of a fact which says that $\leqslant_{\mathrm{T}}$ says a lot about randomness. This fact was extremely surprising when it was first found, and the original proof by Miller and Yu was rather complex. The following result is also true for arbitary $n \geqslant 2$ as we see in Chapter 14.

Theorem 11.6.6 (Miller and Yu [216])**.** *Suppose that A is random and B is 2-random. Suppose also that $A \leqslant_{\mathrm{T}} B$. Then A is 2-random.*

Proof. If B is 2-random, then B is 1-Ω-random (as $\Omega \equiv_{\mathrm{T}} \emptyset'$.) Hence by van Lambalgen's Theorem, $\Omega \oplus B$ is random. Thus, again by van Lambalgen's Theorem, Ω is 1-B-random. But $A \leqslant_{\mathrm{T}} B$. Hence, Ω is 1-A-random. Hence $\Omega \oplus A$ is random, again by van Lambalgen's Theorem. Thus, A is 1-Ω-random. That is, A is 2-random. □

We remark that van Lambalgen's Theorem is known to fail for other randomness notions, such as Schnorr and computable randomness. Interestingly, as observed by Downey, Mileti, Hirschfeldt, and by Yu (and no doubt others), the "hard" direction of van Lambalgen's Theorem, namely Theorem 11.6.5 actually holds, more or less by the same proof. That is, for instance, *if A is Schnorr random, and B is A-Schnorr random, then $A \oplus B$ is Schnorr random.* It is also true that if $A \oplus B$ is either Schnorr or computably random, then so are A and B. It is the analog of Theorem 11.6.2 which fails.

Theorem 11.6.7 (Merkle, Miller, Nies, Reimann, and Stephan [207], Yu [**?**])**.** *Van Lambalgen's Theorem fails for both Schnorr and computable randomness.*

Proof. The (earlier) [207] proof cycles through some material on Kolmogorov-Loveland stochasticity and will be treated in Chapter 12, when we discuss that concept. The following proof is due to Yu [**?**]. The proof filters cleverly through the theorem of Nies, Stephan, and Terwijn [232], Theorem 10.4.8, classifying degrees containing computably and Schnorr random reals. We will need this also in relativized form, as in the following lemma.

Lemma 11.6.8. *Suppose that A_0 is A_1-Schnorr random, and $A_1'' \not\leqslant_T (A_0 \oplus A_1)'$. Then A_0 is A_1-random.*

Proof. Otherwise $A_0 \in \cap_n U_n^{A_1}$ for an A_1 Martin-Löf test $\{U_n^{A_1} : n \in \mathbb{N}\}$. Now we use the familiar trick from 10.4.8. Let f be the $A_0 \oplus A_1$ computable function where $A_0 \in [U_n^{A_1}[f(n)]]$. Then by Martin's highness characterization of highness, there is an A_1-computable function g so that $g(n) > f(n)$ infinitely often. Now we define a Schnorr Solovay test $\{V_n^{A_1} : n \in \mathbb{N}\}$, by making $V_n^{A_1} = U_n^{A_1}[g(n)]$, so that $A_0 \in V_n^{A_1}$ for infinitely many n, and hence A_0 is nor Schnorr $A-1$-random, a contradiction. □

Returning to the proof it will follow from the next result.

Theorem 11.6.9 (Yu [**?**])**.** *Let $B <_T \emptyset'$ be a c.e. set. Then the following hold.*

(i) If $A = A_0 \oplus A_1 \leqslant B$ is Schnorr random but not random, then A_i is not A_{1-i}-Schnorr random for $i \in \{0, 1\}$.

(ii) If $A = A_0 \oplus A_1 \leqslant B$ is computably random but not random, then A_i is not A_{1-i}-computably random for $i \in \{0, 1\}$.

Proof. (i) S uppose not. Suppose that A_0 is A_1-Schnorr random. We already know that both A_0 and A_1 are Schnorr random. By Theorem 10.4.8, both A_0 and A_1 are high. Since each random set computes a diagonally noncomputable function, by Theorem 11.5.1[1]. Thus by Arslanov's completeness criterion, niether of the A_i's (nor A) can be random. Thus by Theorem 10.4.8, $A_i' \equiv_T \emptyset''$ for $i \in \{0, 1\}$. Hence

$$A_1'' \equiv_T \emptyset''' >_T \emptyset'' \equiv_T A' \equiv_T (A_0 \oplus A_1)'.$$

Now we can apply Lemma 11.6.8 to get that A_0 is random, contradicting the fact that $B <_T \emptyset'$.

(ii) Relativizing Schnorr implications about randomness, we see that for each set X, each X-computably random set is X-Schnorr random. Now we can apply (i). If A is computably random, certainly it is Schnorr random, and ny (i), A_i is not $A_{1-i}-$Schnorr random for $i \in \{0, 1\}$. In particular, A_i in also not $A_{1-i}-$computably random. □

[1]We could have delayed the proof of Theorem 11.6.9 until Kučeraa's Theorem saying all 1-randoms bound DNC degrees, but it seemed more appropriate to include it here within our discussion of indepensence results.

□

11.7 Measure Theory and Turing reducibility

The first applications of measure theory to computability theory were due to Spector [287] (1958), who showed that almost all pairs of hyperdegrees are incomparable, and even earlier (1956) to de Leeuw, Moore, Shannon, and Shapiro [60]. Their result, which we now discuss, was not well-known in the computability theory community until recently, and has as an immediate corollary another fundamental theorem first explicitly stated by Sacks [260].

Recall that an index e is universal if, for each index i, there is a string σ_i such that $W_e^{\sigma_i X} = W_i^X$ for all sets X. Fix a universal index e and define the *enumeration probability* of A as

$$P(A) := \mu(\{X \in 2^\omega : W_e^X = A\}).$$

If A is c.e. then there is an i such that $W_i^X = A$ for all X, so $P(A) > 0$. The following result shows that the converse also holds.

Theorem 11.7.1 (de Leeuw, Moore, Shannon, and Shapiro [60])**.** *If $P(A) > 0$ then A is c.e.*

Proof. We use the Lebesgue Density Theorem (Theorem 4.2.3). Recall that a measurable set $S \subseteq 2^\omega$ has density d at X if $\lim_n 2^n \mu(S \cap [X \upharpoonright n]) = d$. Letting $\Xi(S) = \{X : S \text{ has density 1 at } X\}$, the Lebesgue Density Theorem states that if S is measurable then so is $\Xi(S)$, and furthermore, the measure of the symmetric difference of S and $\Xi(S)$ is zero, so $\mu(\Xi(S)) = \mu(S)$.

Suppose that $P(A) > 0$. Then $S := \{X : W_e^X = A\}$ has positive measure, so $\Xi(S)$ has positive measure, and hence there is an X such that S has density 1 at X. Thus, there is an n such that $2^n \mu(S \cap [X \upharpoonright n]) > \frac{1}{2}$. Let $\sigma = X \upharpoonright n$.

We can now enumerate A by "taking a vote" among the sets extending σ. More precisely, $n \in A$ iff

$$2^n \mu(\{Y : \sigma \prec Y \wedge n \in W_e^Y\}) > \tfrac{1}{2}, \tag{11.1}$$

and the set of n for which (11.1) holds is clearly c.e. □

Theorem 11.7.1 has a very interesting corollary, which was not explicitly stated in [60], but later independently formulated by Sacks [260]. For a set A, let

$$A^{\leqslant_T} = \{B : A \leqslant_T B\}.$$

It is natural to ask whether there is a noncomputable set A such that $A^{\leqslant_T}$ has positive measure. Such a set would have a good claim to being "almost computable". Furthermore, the existence of a noncomputable set A and an i such that $\mu(\{B : \Phi_i^B = A\}) = 1$ could be considered a counterexample to the Church-Turing Thesis, since a machine with access to a random source would be able to compute A. The following result lays such worries to rest.

Corollary 11.7.2 (Sacks [260]). *If $\mu(A^{\leqslant_T}) > 0$ then A is computable.*

Proof. If $\mu(A^{\leqslant_T}) > 0$ then there is an i such that $\{B : \Phi_i^B = A\}$ has positive measure. It is now easy to show that there are j and k such that $\{B : W_j^B = A\}$ and $\{B : W_k^B = \overline{A}\}$ both have positive measure. Thus $P(A) > 0$ and $P(\overline{A}) > 0$, and hence A and $\overline{A}$ are both c.e. $\square$

It is not hard to adapt the proof of Theorem 11.7.1 to show that if $\mu(\{B : A \text{ is } \Delta_2^B\}) > 0$ then A is Δ_2^0. From this result it follows that if $\mathbf{d} > \mathbf{0}'$, then $\mu(\{B : \deg(B') \geqslant \mathbf{d}\}) = \mathbf{0}$. In particular, the collection of high sets has measure 0. We will later improve this result by showing that in fact almost all sets are GL_1.

The following is a useful consequence of Corollary 11.7.2

Corollary 11.7.3. *If $D >_T \emptyset$ and A is weakly 2-random relative to D, then $D \not\leqslant_T A$.*

Proof. Suppose that $\emptyset <_T D \leqslant_T A$. Let e be such that $\Phi_e^A = D$, and let $S = \{X : \Phi_e^X = D\}$. Then S is a Π_2^D class, and by Corollary 11.7.2, it is null. Since $A \in S$, it follows that A is not weakly 2-random relative to D. $\square$

Corollary 11.7.4 (Kautz, [140]). *If A and B are relatively 2-random, then their degrees form a minimal pair.*

Corollary 11.7.4 does not hold for 1-randomness.

Theorem 11.7.5 (Kučera [156]). *If A and B are 1-random with $A, B <_T \emptyset'$ then A and B do not form a minimal pair.*

Proof. By Theorem 11.5.1 (to follow), if a degree is 1-random then it is diagonally noncomputable and hence fixed point free. By Theorem **??**, if a degree below $\mathbf{0}'$ is fixed point free it is not half of a minimal pair. $\square$

Randomness is linked to properties of "almost all" degrees. Here is one example which is a nice generalization of Sacks's Theorem.

Theorem 11.7.6 (Stillwell [296]). *Suppose that $\mu(\{C : D \leqslant_T A \oplus C\}) > 0$. Then $C \leqslant_T A$.*

Proof. This again uses the "majority vote" technique of Theorem 11.7.1. By countable additivity there must be an e such that

$$\mu(\{C : D = \Phi_e^{A \oplus C}\}) > 0.$$

Considering those m and C for which $D \upharpoonright m = \Phi_e^{A \oplus C} \upharpoonright m$, the probability $\to 1$ that this is a C with $D = \Phi_e^{A \oplus C}$ as $m \to \infty$. Let m_0 be a m for which this probability is $> \frac{3}{4}$.

Given $D \upharpoonright m_0$, we compute $D(x)$ for $x > m_0$ as follows: Suppose that $\mu(\{C : D \upharpoonright m_0 = \Phi_e^{A \oplus C} \upharpoonright m_0\} = d$. Take a rational r with $\frac{d}{2} < r < \frac{3d}{4}$. List the finite sequences τ_i for which there is an initial segment $\sigma_i \preccurlyeq A$ so that $\Phi_e^{\sigma_i \oplus \tau_i} \upharpoonright m_0 = D \upharpoonright m_0$. Those τ_i which also give $\Phi_e^{A \oplus \tau_i} = D(x)$ will determine neighborhoods of measure at least $\frac{3d}{4}$ out of a possible d. Once we enumerate neighborhoods with measure at least r, all having the same value on x, we will know that this is the correct answer and hence know $D(x)$. $\square$

An immediate corollary to this result (which we improve later) is the following.

Corollary 11.7.7 (Stillwell [296])**.** *For any* $\mathbf{a}, \mathbf{b}$, $(\mathbf{a} \cup \mathbf{b}) \cap (\mathbf{a} \cup \mathbf{c}) = \mathbf{a}$, *for almost all* $\mathbf{c}$.

Proof. Take $D \leqslant_{\mathrm{T}} A \oplus B$. Then by Theorem 11.7.6, if $D \leqslant_{\mathrm{T}} A \oplus C$ for more than a measure 0 set of C, $D \leqslant_{\mathrm{T}} A$. Hence for almost all C,

$$D \leqslant_{\mathrm{T}} A \oplus B \wedge D \leqslant_{\mathrm{T}} A \oplus C \to D \leqslant_{\mathrm{T}} A.$$

$\square$

11.8 Stillwell's Theorem

We will use the material from the previous section to prove Stillwell's Theorem on the "almost all" theory of the Turing degrees. Here the usual quantifiers $\forall$ and $\exists$ are interpreted to mean "for almost all". By Fubini's Theorem, $A \subset (2^\omega)^n$ has measure 1 means that almost all section of A with first coordinate fixed have measure 1. This implicitly allows us to deal with nested quantifiers.

Theorem 11.8.1 (Stillwell [296])**.** *The "almost all" theory of degrees is decidable.*

Proof. Variables $\mathbf{a}, \mathbf{b}, \mathbf{c}, \ldots$ vary over arbitrary degrees. Terms are built from $'$ (jump), $\cup, \cap$. An atomic formula is one of the form $t_1 \leqslant t_2$ for terms t_1, t_2, and formulae in general are built from atomic ones and $\wedge, ^-$ and the quantifier $\forall$ interpreted to mean "for almost all."

Note that Corollary 11.7.7 allows us to compute the meet of any two terms of the form $\mathbf{a}_1 \cup \mathbf{a}_2 \cup \cdots \cup \mathbf{0}^{(f)}$ and $\mathbf{a}_1 \cup \mathbf{b}_2 \cup \cdots \cup \mathbf{0}^{(k)}$ as $\mathbf{c}_1 \cup \mathbf{c}2 \cup \cdots \cup \mathbf{0}^{(\min\{f,k\})}$, where $\mathbf{c}_i$ are variables common to both terms. For example $(\mathbf{a}_1 \cup \mathbf{a}_3 \cup \mathbf{0}^{(4)}) \cap (\mathbf{a}_1 \cup \mathbf{a}_5 \cup \mathbf{a}_7 \cup \mathbf{0}^{(6)}) = \mathbf{a}_1 \cup \mathbf{0}^{(4)}$. Note that also for almost all $\mathbf{c}$, $\mathbf{a} \cap \mathbf{c} = \mathbf{0}$.

Lemma 11.8.2 (Stillwell [296])**.** *For all A, $(A \oplus B)' \equiv_{\mathrm{T}} A' \oplus B$ for almost all B.*

Proof. We delay the proof till the next section, where it is established with sharper bounds in Kautz' Theorem 11.10.7. □

The following corollary is needed, and will later be extended by the work of Kautz.

Corollary 11.8.3 (Stillwell [296])**.** *For almost all* $\mathbf{a}$, $\mathbf{a}^{(n)} = \mathbf{a} \cup \mathbf{0}^{(n)}$.

Proof. By induction on n. We suppose that $\mathbf{a}^{(n)} = \mathbf{a} \cup \mathbf{0}^{(n)}$ for almost all $\mathbf{a}$. Now $\mathbf{a}^{(n+1)} = (\mathbf{a} \cup \mathbf{0}^{(n)})' = \mathbf{a} \cup \mathbf{0}^{(n+1)}$, for almost all $\mathbf{a}$, by Lemma 11.8.2. □

Now we can give a normal form for terms:

We can use Corollary 11.8.3 to compute the jump of any term of the form $\mathbf{a}_1 \cup \mathbf{a}_2 \cup \ldots \mathbf{a}_m \cup \mathbf{0}^{(f)}$. That is, as $(\mathbf{a}_1 \cup \mathbf{a}_2 \cup \mathbf{a}_m \cdots \cup \mathbf{0}^{(f)}) = (\mathbf{a}_1 \cup \mathbf{a}_2 \cup \ldots \mathbf{a}_m)^{(f)}$, and hence

$$(\mathbf{a}_1 \cup \mathbf{a}_2 \cup \ldots \mathbf{a}_m \cup \mathbf{0}^{(f)})' = \mathbf{a}_1 \cup \mathbf{a}_2 \cup \ldots \mathbf{a}_m \cup \mathbf{0}^{(f+1)}.$$

Hence, using the rule for $\cap$, the jump rule, and the join rule $(\mathbf{a}_1 \cup \mathbf{a}_2 \cup \mathbf{a}_m \cdots \cup \mathbf{0}^{(f)}) \cup (\mathbf{b}_1 \cup \mathbf{b}_2 \cup \mathbf{b}_n \cdots \cup \mathbf{0}^{(k)}) = (\mathbf{a}_1 \cup \mathbf{a}_2 \cup \mathbf{b}_n \cdots \cup \mathbf{0}^{(\max\{f,k\})})$, we can reduce any term t to one of the form

$$(\mathbf{a}_1 \cup \mathbf{a}_2 \cup \mathbf{a}_m \cdots \cup \mathbf{0}^{(f)}),$$

for almost all $\mathbf{a}_1 \cup \mathbf{a}_2 \cup \mathbf{a}_m$.

The proof is completed by giving the decision procedure. Following Stillwell we show every formula with free variables is satisfied by a set of instances of measure 0 or 1, (the property being called being 0-1 valued), and we can effectively calculate the value. This is done by induction on the structure of the formulae.

To see that formulae $t_1 \leqslant t_2$ are effectively 0-1 valued, first put the terms t_i in normal form, and calculate $t_1 \cap t_2$. If $t_1 \cap t_2 = t_1$ then $t_1 \leqslant t_2$ for almost all instances and hence the formula has value 1. Otherwise we must have $t_1 \cap t_2 < t_1$ either because t_1 is not a variable in the intersection, or the $\mathbf{0}^{(f)}$ term in t_1 is too big. In any case $t_1 \leqslant t_2$ will have value 0.

If ψ_1, ψ_2 are effectively 0-1 valued then so are $\psi_1 \wedge \psi_2$ and $\overline{\psi_i}$ by elementary considerations. Fubini's theorem then does this for $\forall a_i \psi(a_1, \ldots, a_n)$. That completes the proof. □

11.9 Effective 0-1 Laws

Stillwell's Theorem suggest that there is a lot of 0-1 behavior in the degrees. In this section we will prove another effective 0-1 law, this time for classes. Classically this the 0-1 law is that any class of reals closed under finite translations has measure 0 or measure 1. (Such as Kolmogorov's 0-1 Law, Theorem 4.2.4. For more, see e.g. Oxtoby [235])

Lemma 11.9.1 (Kučera-Kurtz, [140], Lemma IV.2.1). *Let D be a real, and $n \geqslant 1$. Let T be a Π^D_n class of positive measure. Then T contains a member of every $D-n-$random degree. Moreover, if A is any $D-n-$random, then there is some string σ and real B such that $A = \sigma B$ and $B \in T$.*

Proof. Assume that T is a $\Pi_1^{D^{(n-1)}}$-class, with $S = \overline{D}$. Let $r \in \mathbb{Q}$ be such that $\mu(S) \leqslant r < 1$. Suppose that for every B with $A = \sigma B$, $B \in S$. Let E be a set of strings computably enumerable relative to $D^{(n-1)}$ with $S = [F]$. We assume that all the strings in E are disjoint. Let $E_0 = F$ and $E_{s+1} = \{\sigma\tau : \sigma \in E_s \wedge \tau \in E\}$. Then each E_s is c.e. relative to $D^{(n-1)}$. It follows that $A \in [E_s]$ for all s since for every B with $A = \sigma B$, $B \in S$. Also $\mu([E_{s+1}] \leqslant \mu([E_s]) \cdot \mu([E]) \leqslant r^{s+1}$. Therefore A is $\Sigma_1^{D^{(n-1)}}$-approximable, and hence not $D-n$-random. Therefore it must be the case that for some strings σ, and some B, $A = \sigma B$ and $B \notin S$, meaning that $B \in T$. □

Corollary 11.9.2 (Kurtz [165], also Kautz [140]). *(i) Every degree invariant Σ^0_{n+1}-class or Π^0_{n+1} either contains all n-random sets or no n-random sets.*

(ii) In fact the same is true for any such class closed under translations, and such that for all A, if $A \in S$, then for any string σ, $\sigma A \in S$.

Proof. Let S be as described. Then S either has measure 0 or measure 1. If S has measure 1, then it is a union of Π^0_n classes at least one of which has positive measure. By Lemma 11.9.1, it must have a representative of every n-random degree. By degree invariance, it must have every n-random real. If S has measure 0, then $\overline{S}$ is a Σ^0_{n+1} class of measure 1. It must contain every weakly n-random set, and hence every n-random set. □

Later we will have several important applications of the $0-1$-law. They include Martin's Theorem 11.16.19, that almost every degree is hyperimmune, and Kurtz's results that almost every degree has a 1-generic predecessor, and almost every degree is computably enumerable in some strictly lesser one. (Theorems 11.16.21 and krg2.) These last theorems have proofs that use the method of "betting measure" and are very interesting.

Corollary 11.9.3 (Kurtz and others). *(i) The class $\{A : A$ has non-minimal degree $\}$ has measure 1, and includes every 1-random set.*

(ii) The class $\{A \oplus B : A, B$ form a minimal pair $\}$ has measure 1, and includes all 2-random but not every 1-random set.

(iii) The class $\{A : \deg(A)$ is hyperimmune $\}$ has measure 1 and includes all 2-random but not every 1-random set.

Proof. (i) By Corollary 11.6.4, no random set has minimal degree.

(ii) Stillwell [296] showed that the sets of joins of minimal pairs has measure 1. This is because every non-GL_2 degree is the top of a minimal pair. We saw in Corollary 11.7.4, that every n-random set is the joint of a minimal pair. (This in fact also shows that the set has measure 1.) Kućera (Theorem 11.7.5) proved that if A and B are random and $<_{\mathrm{T}} \emptyset'$ then they do not form a minimal pair.

(iii) This follows by Theorem 11.16.19. $\square$

11.10 n-randomness and Kurtz n-randomness

In the last Chapter, in Theorem 10.5.8, we observed that every n-random real was Kurtz n-random, and that every Kurtz $n+1$-random real was n-random. In this section, we will show that neither implication can be reversed.

Theorem 11.10.1 (Kurtz, [165])**.** *For every $n \geqslant 1$, there is an n-random real that is not of Kurtz $n+1$-random degree.*

Proof. By relativizing the Kriesel basis theorem, there is a n-random real of degree computable from $\emptyset^{(n)}$. (Actually one can see this by considering $\Omega^{\emptyset^{(n-1)}}$.) Suppose that $A \leqslant_{\mathrm{T}} \emptyset^{(n)}$. Consider, for each e,

$$P = \{\alpha : \Phi_e^\alpha = A\} = \{\alpha : \forall x \forall s (\Phi_e^\alpha(x)\downarrow \rightarrow \Phi_e^\alpha(x) = A(x)\}.$$

The by Corollary 11.7.2, P is a $\Pi_1^{\emptyset^{(n)}}$-class, of measure 0, and hence $\overline{P}$ is a Σ_{n+1}^0 class of measure 1, which any Kurtz $n+1$-random real must be in. $\square$

Corollary 11.10.2 (to the proof)**.** *(i) No Kurtz $n+1$-random set is computable from $\emptyset^{(n)}$.*

(ii) There is a n-random set computable from $\emptyset^{(n)}$ of Σ_n^0 degree.

We remark that it is also easy to construct an arithmetically random set computable from $\emptyset^{(\omega)}$.

Theorem 11.10.3 (Kurtz [165] for $n = 1$, Kautz [140])**.** *Let $n \geqslant 1$. Then there is a Kurtz n-random set that is not n-random.*

Proof. This proof was originally given by Kautz, and the result stated without proof by Gaifman and Snir [119]. Kautz' proof was fairly complicated, but we can get this pretty easily using material from Chapter 10. Assume that we have a version of Theorem 10.5.22, but in relativized form, so that we know that for every $n \geqslant 1$, every set X which is $\Sigma_1^{\emptyset^{(n-1)}}$ and above $\emptyset^{(n-1)}$, there is a Kurtz n-random set S such that $S \oplus \emptyset^{(n-1)} \equiv_{\mathrm{T}} X$. Then, the other hand, we know that by relativizing Theorem 13.3.4, that if a Q is of degree c.e. relative to $\emptyset^{(n-1)}$, and there is an n-random set R with $R \oplus \emptyset^{(n-1)} \equiv_{\mathrm{T}} Q$, then $Q \equiv_{\mathrm{T}} \emptyset^{(n)}$. $\square$

To complete the proof above, we need the analog of Theorem 10.5.22 to hold. This is by not means obvious since, as we recall, Kurtz $n+1$ randomness is *not* Kurtz 1-randomness relative to $\emptyset^{(n)}$. Nevertheless, the analog still holds.

Theorem 11.10.4. *Every degree* **a** *computably enumerable in and above $\emptyset^{(n-1)}$ contains a Kurtz n-random set S such that $S \oplus \emptyset^{(n-1)}$ has degree* **a**.

Proof. We recall Theorem 10.5.7. Namely, a real α is Kurtz n-random iff for each $\emptyset^{(n)}$ computable sequence of Σ^0_{n-1} classes $\{S_i : i \in \mathbb{N}\}$, with $\mu(S_i) \leqslant 2^{-i}$, $\alpha \notin \cap_i S_i$. And consequently, α is Kurtz n-random iff for every $\emptyset^{(n)}$ computable sequence of open $\Sigma_1^{\emptyset^{(n-2)}}$ classes $\{S_i : i \in \mathbb{N}\}$, with $\mu(S_i) \leqslant 2^{-i}$, $\alpha \notin \cap_i S_i$. We will use this characterization for the proof.

In fact we will deal with the case of Kurtz 2-randomness and then relativize. Our construction assumes that we have a Σ^0_2 set $X > \emptyset'$, and will be an oracle construction relative to $X \oplus \emptyset' \equiv_{\mathrm{T}} X$. Since X is c.e. relative to $\emptyset'$ we can assume that we have a $\emptyset'$ enumeration $X = \cup_s X_s$ of it. Let $\{V_e^i : i, e \in \mathbb{N}\}$ be a $\emptyset'$-listing of all $\emptyset'$ c.e. open Σ^0_1 classes, with $V_e^i \supseteq V_e^{i+1}$. (That is, V_e^i is of the form $\cup\cup\{[\sigma] : \sigma \in W_{f(e,i)}\}$, and prefix-free.) We must meet a requirement of the type:

$$R_e : \mu(V_e^i) \to 0 \text{ implies } S \notin \cap_i V_e^i.$$

additionally we must code X into $S \oplus \emptyset'$. Of course, $\emptyset'$ cannot know if $\mu(V_e^i) \to 0$. Notice that *if* $\emptyset'$ could enumerate indices $i_0, i_1, \ldots, i_e, \ldots$ such that $\mu(V_e^{i_e}) < 2^{-(e+1)}$, then we in fact $\emptyset'$ could construct S in the same way that it can construct a 1-random real. However, we can now use a finite injury argument with permitting over $\emptyset'$. That is because $\emptyset'$ (and hence X) can *recognize* uniformly in e, i and k if $\mu(V_e^i) \not\geqslant 2^{-k}$.

Thus without coding, the argument for each e, is to try to define a collection of permitting markers attempting to show that $X \leqslant_{\mathrm{T}} \emptyset'$. For example for $e = 0$, we might have $k(1) = 2$. we do nothing for the sake of R_0 until we see a stage s where and some i, $\mu(V_0^i) \not\geqslant 2^{-2}$. Whilst we are doing this, we are pursuing the R_q-strategies for $q > 0$ in some cone, say [0000] of measure at most 2^{-4}. At this stage s_0, we would set $i_1 = i$, and declare that if $X \upharpoonright 2 \neq X_s \upharpoonright 2$, then we will allow ourselves to avoid $V_0^{i_1}$ using strings of length $\geqslant 2$. Whislt we are waiting for this to happen, we would initialize lower priority R_q so that they would work with much smaller measure, and also reset $k(2) = s + 2$, say.

The point is that since $\emptyset' <_{\mathrm{T}} X$, some $k(j)$ will be X-permitted and hence then we can use $\emptyset'$ to avoid the relevant V_e^i. It all fits easily together using a finite injury argument.

Coding is easy. For each f, to code $f \in X$ we need some marker $\gamma(f, s)$ which has a limit, and additionally, $\alpha(f, s+1) \neq \alpha(f, s)$ only if $S_{s+1} \upharpoonright \alpha(f, s) \neq S_{s+1} \upharpoonright \alpha(f, s)$ (the construction only moves left, so it constructs a ce-relative to $\emptyset'$-real). Then to decide if $f \in X$ we will force decide this at the stage t where $S_t \upharpoonright \alpha(f, t) + 1 = S \upharpoonright \alpha(f, t) + 1$. We need to have $S_s(\alpha(f,s)+1) = 0$, and we will change to 1 if f enters X. As with Friedberg coding, we only mode the marker when some R_k for $k < f$ receives attention. We can clearly arrange measures so that the coding locations are safe from being upset by lower priority R_q. The details are straightforward and familiar. $\square$

Corollary 11.10.5. *(i) If an n-random set A is computably enumerable in $\emptyset^{(n-1)}$ then $A \oplus \emptyset^{(n-1)}$ is of degree $\mathbf{0^n}$.*

(ii) For every degree $\mathbf{a}$ computably enumerable in and above $\emptyset^{(n-1)}$ and not equal to $\emptyset^{(n)}$, there is a Kurtz n-random S S such that $S \oplus \emptyset^{(n-1)}$ has degree $\mathbf{a}$.

We remark that we can suitably relativize the notions of Schnorr and computable randomness.

11.10.1 Most degrees are GL_1

The fact that the only random c.e. real is Ω gives the impression that randomness is a notion "like" the halting problem and is a highness notion. This is not true. We will also see reflections of this fact in the work of Miller and of Miller and Yu who proves that as reals become more random, they become more like trivial reals. We see this in Chapter 14. Recall that the class $\mathrm{GL}_1 = \{X : X' \equiv_{\mathrm{T}} \emptyset' \oplus X\}$ is the collection of *generalized low* sets. The following generalizes Stillwell's Theorem 11.8.1.

Theorem 11.10.6 (Kautz [140])**.** *The class $\{X : X^{(n-1)} \equiv_{\mathrm{T}} X \oplus \emptyset^{(n-1)}\}$ has measure 1* (Sacks)*, and includes every n-random set, but not every $n-1$ random set.*

Proof. Kautz' main idea to build a particular procedure Φ so that

$$\{A : A^{(n)}(e) \neq \Phi^{A \oplus \emptyset^{(n)}}(e)\}$$

is Σ^0_{n+1} approximable. Then if A is $n+1$-random, $A^{(n)}(e) = \Phi^{A\oplus\emptyset^{(n)}}(e)$ for almost all e, and hence $A^{(n)}(e) \leqslant_{\mathrm{T}} \Phi^{A\oplus\emptyset^{(n)}}(e)$.

Without loss of generality, suppose that $n \geqslant 1$. For each e, define a class

$$C_e^n = C_e = \{A : e \in A^{(n)}\}.$$

We claim that C_e is a Σ^0_n class. Inductively, suppose that C_e^{n-1} is a Σ^0_{n-1} class. Then the predicate $\sigma \prec \alpha$ for $\alpha \in C_e^{n-1}$ is Δ^0_n. Hence,

$$A \in C_e \text{ iff } \exists\sigma(\sigma \subset A^{(n-1)} \wedge \varphi_e^\sigma(e)\downarrow),$$

is Σ^0_n. Furthermore, an index for C_e can be found uniformly from e. Hence by Theorem 9.7.3, uniformly in $\emptyset^{(n)}$, we can find a $\Sigma_n^{\emptyset^{(n-1)}}$ class U_e with $C_e \subseteq U_e$ and $\mu(U_e) - \mu(C_e) \leqslant 2^{-(e+1)}$. Since U_e is $\Sigma_1^{\emptyset^{(n-1)}}$, we have that $U_{e,s} = \cup\{[\sigma] : \sigma W_{z,s}^{\emptyset^{(n-1)}}$ for some c.e. set W_z, and uniformly in $\emptyset^{(n)}$ we can find a stage $s(e)$ such that $\mu(U_e) - \mu(U_{e,s(e)}) \leqslant 2^{-(e+1)}$. We define $U_{e,s(e)}$ by defining a partial computable Ψ such that for each e, if a set B happens to be $\emptyset^{(n)}$ then $\Psi^B(e)$ converges and its output is the index of a finite set of strings such that $U_{e,s(e)}$ is the collection of the extensions of these strings, that is $U_{e,s(e)} = \cup\{[\sigma] : \sigma \in \Psi^B(e)\}$. We are ready to define Φ Let

$$\Phi^{A \oplus B}(e) = \begin{cases} 1 & \text{if } \Psi^B(e) \downarrow \wedge A \in \cup\{[\sigma] : \sigma \in \Psi^B(e)\} \\ 0 & \text{if } \Psi^B(e) \downarrow \wedge A \notin \cup\{[\sigma] : \sigma \in \Psi^B(e)\} \\ \uparrow & \text{otherwise.} \end{cases}$$

Thus,

$$\Phi^{A \oplus \emptyset^{(n)}}(e) = \begin{cases} 1 & \text{if } A \in U_{e,s(e)} \\ 0 & \text{otherwise.} \end{cases}$$

To complete the proof we will show that the class $\{A : A^{(n)}(e) \neq \Phi^{A \oplus \emptyset^{(n)}}(e)\}$ is Σ^0_{n+1} approximable. We see that $A^{(n)}(e) \neq \Phi_e^{A \oplus \emptyset^{(n)}}(e)$ iff either $A \in C_e$ and $A \notin U_{e,s(e)}$; or $A \notin C_e$ and $A \in U_{e,s(e)}$. Since $\mu(C_e - U_{e,s(e)}) \leqslant 2^{-(e+1)}$, and $\mu(U_{e,s(e)} - C_e) \leqslant 2^{-(e+1)}$, the class $S_e = (C_e - U_{e,s(e)}) \cup (U_{e,s(e)} - C_e))$ has measure at most 2^{-e}. Since $U_{e,s(e)}$ is clopen and computable in $\emptyset^{(n)}$, we see that S_e is a Σ^0_{n+1} class, and we are done. □

By relativizing to A-n-random sets, we see:

Theorem 11.10.7. *The class* $\{X : (X \oplus A)^{(n-1)} \equiv_{\mathrm{T}} X \oplus A^{(n-1)}\}$ *has measure 1* (Sacks)*, and includes every n-A-random set, but not every* $(n-1) - A$ *random set.*

We have seen that Kučera [155] proved that for every degree $\mathbf{a}$ above $\mathbf{0}'$ there is a 1-random set A with $A' \in \mathbf{a}$. Kautz generalized this as follows.

Theorem 11.10.8 (Kautz [140])**.** *Suppose that* $\mathbf{a} \geqslant \mathbf{0}^{(n)}$*. Then there is an n-random real A with* $A^{(n)} \in \mathbf{a}$*.*

Proof. We will use the proof of Kučera's Theorem 11.4.1 in relativized form. Recall that the basic idea of the proof consisted of constructing a perfect tree $T \leqslant_{\mathrm{T}} \emptyset'$ such that $[T] \subseteq \overline{U_0}$ so that contained only random sequences; and every path codes a set $B \subseteq \mathbb{N}$. This coding was effective due to Lemma 11.4.2, which allowed us to compute an effective lower bound for the measure of $\overline{U_0}$.

Now this can all be relativized to $\emptyset^{(n-1)}$. We can construct this tree computable from $\emptyset^{(n)}$ and have the lower bound computable from $\overline{U_0^{\emptyset^{(n-1)}}}$. Now since A is n-random we would have $A^{(n)} \equiv_{\mathrm{T}} \emptyset^{(n)} \oplus A$, by the previous Theorem. Hence $A^{(n)}$ can sort out the coding, thus if we mimic the same proof, as observed by Kautz we would get $B \equiv_{\mathrm{T}} A^{(n)}$. □

Notice that since every weakly $n+1$-random real is n random, it must have $A^{(n-1)} \equiv_{\mathrm{T}} A \oplus \emptyset^{(n-1)}$. Hence the class $\{A : A \oplus \emptyset^{(n-1)} \geqslant_{\mathrm{T}} \emptyset^{(n)}\}$ has measure zero and contains no weakly $n+1$-random real.

11.11 DNC degrees and autocomplex reals

The reader might well wonder exactly what reals are diagonally noncomputable. Kjos-Hanssen, Merkle and Stephan extended Kučera's Theorem (Theorem 11.5.1) to characterize the reals computing diagonally noncomputable functions in terms of Kolmogorov complexity.

Definition 11.11.1 (Kjos-Hanssen, Merkle, Stephan [**?**])**.** (i) We say that a real x is *order complex* if there is an order h such that $C(x \upharpoonright n) > h(n)$. We will say that A is *h-complex*

(ii) We say that a real x is *autocomplex* if there is a x-computable order g with $C(x \upharpoonright n) \geqslant g(n)$ for all n.

Notice that we could have used K in place of C in the definition above since we know that K is a 2-approximation for C.

Clearly by Schnorr's Theorem, we can choose $f(n) = n$ for a 1-random x. We will now prove some interesting characterizations of autocomplexity and order complexity.

Lemma 11.11.2 (Kjos-Hanssen, Merkle, Stephan [**?**])**.** *The following are equivalent.*

(i) x is autocomplex.

(ii) There is an x-computable function h such that, for all n, $C(A \upharpoonright h(n)) \geqslant n$.

(iii) There is an x-computable f such that $C(f(n)) > n$ for all n.

(iv) For all orders $h(n) \geqslant n$, there is an x-computable f such that $C(f(n)) > h(n)$ for all n.

Proof. To show that (i) implies (ii), let x be autocomplex and choose the x-computable order g with $C(x \upharpoonright n) \geqslant g(n)$ for all n. Then we get (ii) by taking

$$h(n) = \min\{p : g(p) \geqslant n\}.$$

To see that (ii) implies (iii) let $f(n)$ be the encoding of $x \upharpoonright h(n)$. To see that (iii) implies (i), let f be givan as in (iii). Let $q(n)$ be an x-computable order ssuch that some fixed oracle machine M computes f with oracle x such that M queries $x \upharpoonright q(n)$ when computing $f(n)$. Then for any $m \geqslant q(n)$, the value of $f(n)$ can be computed from n and $x \upharpoonright m$. Therefore

$$n \leqslant C(f(n)) \leqslant C(x \upharpoonright m) + 2\log n + \mathcal{O}(1).$$

Hence, for almost all n and all $m \geqslant q(n)$, $\frac{n}{2} \leqslant C(x \upharpoonright m)$. Therefore a finite variation of the x-computable order $g : n \mapsto \frac{1}{2}\max\{m : q(m) \leqslant n\}$ will witness the autocomplexity of x. Finally (iii) is equivalent to (iv) as follows. First suppose (iii). Then let $h(n) \geqslant n$ be an order. Take f as in (iii), and let $g(n) = f(h(n))$. Then $C(g(n)) \geqslant C(f(h(n)) > h(n) \geqslant n$. The other direction is clear. □

Similar methods a allow us to characterize order complex reals[2].

Theorem 11.11.3 (Kjos-Hanssen, Merkle, Stephan [**?**])**.** *For any real x, the following are equivalent.*

(i) x is complex.

(ii) There is a computable function h such that for all n, $C(x \upharpoonright h(n)) \geqslant n$.

(iii) x tt-computes a function f such that for all n, $C(f(n)) \geqslant n$.

(iv) x wtt-computes a function f such that for all n, $C(f(n)) \geqslant n$.

We next need a lemma relating DNC and autocomplexity. It can be viewed in some sense as the ultimate generalization of Kučera's Theorem that randoms are DNC.

Lemma 11.11.4 (Kjos-Hanssen, Merkle, Stephan [**?**])**.** *x is autocomplex iff x is DNC.*

Proof. The following proof is taken from [**?**]. Assume that x is autocomplex and let f be as in Lemma 11.11.2 (iii). Then we will have for some constant c and almost all n,

$$C(\{n\}(n)) \leqslant C(n) + c \leqslant \log n + 2c < n \leqslant C(f(n).$$

Thus we can use a finite variation of f as a DNC function.

Conversely, suppose that x is not autocomplex. Suppose that x computes a DNC function φ_r. For each z there is an index $e(z)$ such that for every input x, $\varphi_{e(z)}(x)$ is computed as follows.

First assume that z is the code of some prefix w of an oracle (i.e. the target being an initial segment of x), and then try to decode this prefix by simulating the universal Turing machine used to define C on input z. If this succeeds, then simulate $\varphi_r(x)$ with the oracle w as an oracle. If this halts output the value.

Now consider the x-computable function h where $h(n)$ is the maximum of the uses of all values of $\varphi_r^x(e(z))$ with $|z| < n$ on oracle x. As x is not autocomplex, by Lemma 11.11.2, there exist infinitely many n such that

[2]We remark that order complex sets were shown to be the same as the *hyperavoidable* sets of Miller [**?**]. Here a set A is called hyperavoidable iff there is some order h such that for all $m \leqslant h(n) - 1$, $A \upharpoonright m \neq \varphi_m(m)$. Miller showed that these are exactly the sets that are not weak truth table reducible to a hyperimmune set. Thus order complex sets are exactly those sets that are not wtt-reducible to any hyperimmune set.

the complexity of $x \upharpoonright h(n)$ is below n. Let this be witnessed by a code z_n. Then for all such n and z_n,

$$\varphi_r(e(z_n)) = \varphi_{e(z_n)}(e(z_n)).$$

But then φ_r^x is *not* DNC, a contradiction. □

We remark that again similar methods yield results for order complex sets.

Corollary 11.11.5 (Kjos-Hanssen, Merkle, Stephan [**?**]). *The following are equivalent.*

(i) x is order complex.

(ii) x tt-computes a DNC function.

(ii) x wtt-computes a DNC function.

In particular, the sets that wtt-compute DNC functions and to compute computable bounded DNC functions coincide.

We have seen that initial segment complexity and Turing complexity are strongly related in the presence of enumeration. For instance, by Theorem 13.9.3, we know that no c.e. real which is not sw above all c.e. sets can have its complexity bounded away from $K(n)$. We have the following.

Theorem 11.11.6 (Kjos-Hanssen, Merkle, Stephan [**?**]). *(i) A is c.e. and order complex iff A is wtt-complete.*

(iii) If A is c.e. and autocomplex iff A is Turing complete.

Proof. Each direction above follows from the Lemmas above and the Arslanov completeness criterion. □

We remark that one of the points of the paper [**?**] was to show how to avoid the use of Arslanov's completeness criterion in results like the above.

A direct proof of Theorem 11.11.6 was given in [**?**]. We also could give a direct proof based on the methods of Theorem 11.2.1 and 13.9.3. That is if we suppose that $K(A \upharpoonright n) \geqslant h(n)$ for an order h, then we can work where $h(n)$ is above $2 \log n + c$, where c is our coding constant. Then we can assume enumerations where this is all true at stage s. If n enters $\emptyset'[s]$ then we can lower the complexity of $A_s \upharpoonright n$ to below $h(n)$ and after that $A - A_s \upharpoonright n$ must change.

The following result will be important when we consider lowness for Kurtz randomness in Chapter 16.

Theorem 11.11.7 (Kjos-Hanssen, Merkle, Stephan [**?**]). *The following are equivalent*

(i) x is DNC or of high degree.

(ii) x is autocomplex or or high degree.

(iii) *x computes a function g which dominates all computable functions, or for every partial computable function h, $g(n) \neq h(n)$ for almost all n.*

(iv) *There is a $f \leqslant_T x$ such that for every computable function h, $f(n) \neq h(n)$ for almost all n.*

(v) *There is no weak computable tracing of x in that there is no computable h such that for all $f \leqslant_T x$ there is a computable function g with $|D_{g(n)}| \leqslant h(n)$ for all n and $\exists^\infty n f(n) \in D_{g(n)}$.*

Furthermore, if y is a real which is of hyperimmune free degree and which is not DNC, then for each $g \leqslant_T y$, there are computable functions h and $\widehat{h}$ such that

$$\forall n \exists m \in \{n, n+1, \ldots, \widehat{h(n)}\}(h(m) = g(m).$$

Proof. Now (i) is equivalent to (ii) by Lemma 11.11.4. To see that (iv) implies (iii), if x is high then by domination, (iii) follows. So suppose that x is not high. Let $f \leqslant_x$ be such that for every computable function h, $f(n) \neq h(n)$ for almost all n. Suppose that φ_d is a partial computable function with $f(n) = \varphi_d(n)$ for infinitely many $ni \in \mathrm{dom}\varphi_d$. Let p be a function such that, for each n, there are $n+1$ many k with $f(k) = \varphi_d^{p(n)}(k)$. Then $p \leqslant_T x$. Since x is not high, there is a computable function q such that for infinitely many n, $q(n) > p(n)$. We then define a computable function ψ where $\psi(n) = \varphi_d^{p(n)}(n)$ if this halts, and 0 otherwise. If $q(n) \geqslant p(n)$, then for some $m \geqslant n$, $\varphi_d^{q(n)}(m) = f(m)$. Hence, $\varphi_d^{q(m)}(m) = f(m) = \psi(m)$. Therefore there are infinitely many m where $f(m)$ agrees with a total computable function, contradiction.

To see that (iii) implies (i), suppose that $g \leqslant_T x$ satisfies (iii). If g is dominant then x is high. If g is eventually different from all partial computable functions, then consider the partial computable $\psi(x) = \varphi_x(x)$. Then a finite variation of g will be DNC.

To show that (i) implies (iv), if x is high it is dominant, and hence computes a function eventually different from all computable ones. If x is DNC, let $g \leqslant_T x$ be DNC. We show that there is a function $h \leqslant_T g$ such that for all e, for all $m, n \leqslant e$, $h(e) \neq \varphi_n(m)$ (and so $h(e)$ differs from $\varphi_n(e)$ for all $e \geqslant n$).

Let V_ω be the collection of hereditarly finite sets and let $\pi\colon \omega \to V_\omega$ be an effective bijection. Let $D_e = \{(n,m) : n, m \leqslant e\}$. We define $h(e)$ so that $\pi(h(e))$ is a function from $D_e \to \omega$. For $n, m \leqslant e$, if $\varphi_n(m)\uparrow$ or if $\pi(\varphi_n(m))$ is not a function from D_e to ω, then obviously $h(e)$ is different from $\varphi_n(m)$. If $\pi(\varphi_n(m))$ is a function from D_e to ω, then we ensure that $h(e) \neq \varphi_n(m)$ by making sure that $\pi(h(e))(n,m) \neq \pi(\varphi_n(m))(n,m)$. To do that, we simply define $\pi(h(e))(n,m) = g(t(e,n,m))$, where $t(e,n,m)$ is chosen such that for all y,

$$\varphi_{t(e,n,m)}(y) = \pi(\varphi_n(m))(n,m).$$

To see that (i) implies (v) Suppose that (v) fails, and x is autocomplex or high. Let h be the relevant order h. (Without loss of generality, h can be taken as an order.) Now choose an x-computable p such that for all n,

$$C(x \upharpoonright p(n)) > 2^{h(n)},$$

by Lemma 11.11.2. Now take the strong array $\{D_{g(n)} : n \in \mathbb{N}\}$ weakly tracing $p(n)$, and with $|D_{g(n)}| < h(n)$. Then to calculate $p(n)$, we need only the information $h(n)$, $D_{g(n)}$ and $\log h(n)$ many bits to say which of $D_{g(n)}$ $p(n)$ is. This shows that $C(p(n)) < h(n)^2 = \mathcal{O}(1) < 2^{h(n)}$, a contradiction, if x is autocomplex. If x is high, then choose $p \leqslant_T x$ dominant, and then take any strong array $W_{g(n)}$ for computable g. Then p dominates $\widehat{q}(n) = \max W_{g(n)}$, and hence p cannot be infinitely often traced. Thus x can't be high either. Thus (i) implies (v).

Finally, suppose that (v) holds and x is not high. We first claim that x computes a function f such that for every computable function p with domain $2^{<\omega}$ and almost all n, $f(n) \notin \{p(\sigma) : \sigma \in 2^{<\omega} \wedge \sigma < n\}$. Otherwise, there is a fixed f where for each computable p, infinitely many n, $f(n) \in P_n = \{p(\sigma) : \sigma \in 2^{<\omega} \wedge \sigma < n\}$. Since the size of P_n is bounded by $2^n - 1$, x, (v) fails, a contradiction. Now we will use this f from claim to show that x computes a function which is almost always different from all computable ones. That is, (iii) holds. Given a computable function h, for each string σ with $|\sigma| = n - 1$, define $p(\sigma) = h(n)$. Then clearly f differs from h almost always. Thus (v) implies (iii).

Finally, we verify the last part of the Theorem. So suppose that y is a real which is of hyperimmune free degree and which is not DNC. Then, of course y is not of high degree. We know that there is a computable function h such that $h(n) = g(n)$ infinitely often. Let $f(n) = \mu m \geqslant n(h(m) = g(m)$. Then $f \leqslant_T y$. Since y is hyperimmune free, there is a computable function $\widehat{h}$ majorizing f. This pair, $\widehat{h}, h$ will work. $\square$

11.12 n-random reals compute n-FPF functions

Much of the material in this book demonstrates that being random involves being computationally feeble. The results of the last section show that if A is n-random and $n \geqslant 2$ then A cannot even compute a $\{0,1\}$-valued fixed point free function. We have seen that randomness is also a lowness property.

However, the results of this section will show that n-random reals have some computational power. In particular we cannot replace "$\{0,1\}$-valued fixed point free function" by "fixed point free."

Definition 11.12.1 (Jockusch, Lerman, Soare, and Solovay [134]). We define a relation $A \sim_n B$ as follows.

(i) $A = B$ if $n = 0$.

(ii) $A =^* B$ if $n = 1$.

(iii) $A^{(n-2)} \equiv_{\mathrm{T}} B^{(n-2)}$, if $n \geqslant 2$.

Here, of course, $A^{(n-2)}$ denotes the $(n-2)$-th Turing jump.

Definition 11.12.2 (Jockusch, C. G. , M. Lerman, R. I. Soare, and R. Solovay [134]). A total function f is called *n-fixed point free* (n-FPF) iff for all x,

$$W_{f(x)} \not\sim_n W_x.$$

We note that the usual fixed point free functions are just the 0-FPF functions. We have seen that all 1-random reals compute DNC and hence FPF functions in Lemma 11.5.1. We prove the following definative result.

Theorem 11.12.3 (Kučera [158]). *Suppose that A is $n+1$ random. Then A computes an n-FPF function.*

Proof. We begin with the case $n = 1$, proving that any 2-random rals computes a $*$-FPF function. Then we will indicate how to modify that proof to get the general case.

We begin by recalling Kučera's construction of a universal Martin-Löf test:

Given $n \in \mathbb{N}$, consider all indices $e > n$. For each such e, enumerate all elements of $W_{\{e\}(e)}$ in to U_n (where we understand that $W_{\{e\}(e)}$ is empty if $\{e\}(e)$ is undefined) as long a s the condition

$$\sum_{w \in W_{\{e\}(e)}} 2^{|w|} < 2^{-e}$$

is satisfied. Then

$$\sum_{w \in U_n} 2^{|w|} \leqslant \sum_{e>n} 2^{-e} = 2^{-n}.$$

Naturally this construction relativizes, and we would write U_n^X for the version relative to X. Let

$$D_n^X = \overline{U_n^X}.$$

Notice that each D_n^X is a class of positive measure.

Now suppose that $\mathbf{a}$ is 2-random. Then $\mathbf{a}$ contains a 2-random set A in the class $D_0^{\emptyset'}$, by the Effective 0-1 Law, Theorem 11.9.2. We will denote the i-th column of a set Y as $Y^{[n]} = \{\langle a, n\rangle : \langle a, n\rangle \in Y\}$.

Let h be a computable function define via

$$W_{h(x)}^{\emptyset'} = \{i : W_x^{[i]} \text{ is finite}\}.$$

Now we let e be the index of a computable function with $e(x) > 0$ for all x and $\{e(x)\}(j)$ is an index of a $\Sigma_1^{\emptyset'}$ class V_j^x defined as follows.

(i) $V_{\langle x,j\rangle} = \emptyset$ if $|W_{h(x)}^{\emptyset'}| < j+1$.

(ii) $V_{\langle x,j\rangle} = \cup\{[\tau] : \tau(i) = 1 \text{ if } i \in S_j\}$, if $|W_{h(x)}^{\emptyset'} \geqslant j+1$, and S_j denotes the first $j+1$ elements of $W_{h(x)}^{\emptyset'}$.

Clearly, V_j^x is a $\Sigma_1^{\emptyset'}$ class. It is also cleat that $\mu(V_j^x) < 2^{-j}$. Now by the way that we construct Kučera's universal Martin-Löf test, we see that

$$V_{e(x)}^x \notin D_0^{\emptyset'}.$$

This means that $V_{e(x)} \cap D_0^{\emptyset'} = \emptyset$ for all x. But, by choice of A, $A \in D_0^{\emptyset'}$. For each x, let σ_x denote the string which is the initial segment of A coding the first $x+1$ elements of A. Then, by the construction,

$$W_{h(x)}^{\emptyset'} \neq \{i : \sigma_x(i) = 1\},$$

for all x. We are ready to define or FPF function f. Let

$$W_{f(x)} = \{\langle i, y\rangle : i \geqslant |\sigma_x|\} \cup \{\langle j, y\rangle : j < |\sigma_x| \wedge \sigma_x(j) = 0\}.$$

Notice that A can compute σ_x and hence $f \leqslant_{\mathrm{T}} A$. We claim that $W_{f(x)} \neq^* W_x$. If W_x has finite columns, then $W_{f(x)}$ will differ on at least one of $\{i : \sigma_x(i) = 1\}$, as we have seen, and if all the columns of W_x are infinite, we still know that $W_{h(x)}^{\emptyset'} \neq \{i : \sigma(i) = 1\}$.

Now we turn to $n \geqslant 2$. Iterating the proof of the Friedberg-Muchnik Theorem (in relativized form), we may choose a set B which is is c.e. relative to $\emptyset^{(n-2)}$, $\emptyset^{(n-2)} \leqslant_{\mathrm{T}} B$, and for all j,

$$B^{[j]}|_{\mathrm{T}} \oplus_{j\neq i} B^{[i]}.$$

By analogy with the $*$-case, let h be a computable function such that

$$W_{h(x)}^{\emptyset^{(n)}} = \{i : W_x^{(n-2)} \leqslant_{\mathrm{T}} \oplus_{j\neq i} B^{[i]}\}.$$

This time we can let e be an index of a computable function such that for all x, $e(x) > 0$ and $\{e(x)\}$ is an index for a $\Sigma_1^{\emptyset^{(n)}}$ class V_j^x defined as follows.

(i) $V_j^x = \emptyset$ if $|W_{h(x)}^{\emptyset^{(n)}}| < j+1$.

(ii) $V_j^x = \cup\{[\tau] : \tau(i) = 1 \text{ if } i \in S_j\}$, if $|W_{h(x)}^{\emptyset^{(n)}}| \geqslant j+1$, and S_j denotes the first $j+1$ elements of $W_{h(x)}^{\emptyset^{(n)}}$.

Again we see that $D_0^{\emptyset^{(n)}} \cap V_{e(x)}^x = \emptyset$. Thus, again we choose A of $n+1$-random degree with $A \in D_0^{\emptyset^{(n)}}$, by the effective 0-1 law. We define σ_x as in the $*$-case, and again we see that

$$W_{h(x)}^{\emptyset^{(n)}} \neq \{i : \sigma_x(i) = 1\}.$$

Now we define define an A-computable function g and a set $W_{g(x)}^{\emptyset^{(n-2)}}$ in an analogous way, namely,

$$W_{g(x)}^{\emptyset^{(n-2)}} = \{\langle i, y\rangle : i \geqslant |\sigma_x| \wedge \langle i, y\rangle \in B\} \cup \{\langle j, y\rangle : j < |\sigma_x| \wedge \sigma_x(j) = 0\}.$$

We claim that for all x,

$$W_x^{(n-2)} \not\equiv_{\mathrm{T}} W_{g(x)}^{\emptyset^{(n-2)}}.$$

Again this follows lest $W_{g(x)}^{\emptyset^{(n-2)}} = \{i : \sigma_x(i) = 1\}$.

Finally, the set $W_{g(x)}^{\emptyset^{(n-2)}}$ is computably enumerable in and above $\emptyset^{(n-2)}$. Thus, since g is A-computable, by the uniformities of the Sacks' Jump Theorem, Theorem 5.13.5, there is a A-computable function f such that, for all x,

$$W_{f(x)}^{(n-2)} \equiv_{\mathrm{T}} W_{g(x)}^{\emptyset^{(n-2)}}.$$

Therefore $W_{f(x)}^{(n-2)} \not\equiv_{\mathrm{T}} W_x^{(n-2)}$, for all x, as required. □

Actually, an analysis of the proof above shows that we only need members of the class $B_0^{\emptyset^{(n)}}$. Thus we actually only need Kurtz $n+1$-randomness. Thus we get the following corollary.

Corollary 11.12.4. *Suppose that A is Kurtz $n+1$-random. Then A computes an n-FPF degree.*

11.13 Jump Inversion

It is natural to seek a combination of the Kučera and Kučera-Gács Theorems. We know that every degree is computable in a 1-random one, and every degree above $\mathbf{0}'$ is 1-random. However, in the last section we saw that there is no common generalization. There are 1-random Δ_2^0 reals α and degrees between the degree of α and $\mathbf{0}'$ which are not 1-random. Thus the 1-random degrees are not closed upwards even for degrees below $\mathbf{0}'$.

In this section we will show that the jump operator can be used to address the distribution of 1-random (Δ_2^0) degrees.

We will do this by proving a new basis theorem for Π_1^0 classes of positive measure.

We would like to prove that if P is a nontrivial Π_1^0 class then it must have members of all possible jumps. This is of course false, since we could be dealing with a countable Π_1^0 class, which might have only one However, as we recall from Chapter 5, (Theorem 5.16.16), we can prove the following

Lemma 11.13.1 (Folklore see [41])**.** *If $\mathcal{P}$ is a nonempty Π^0_1 class with no computable members and $S \geqslant_{\mathrm{T}} \emptyset'$, then there is an $A \in \mathcal{P}$ such that $A' \equiv_{\mathrm{T}} A \oplus \emptyset' \equiv_{\mathrm{T}} S$.*

Applying the Lemma to a Π^0_1 class of 1-random reals, we see the following.

Corollary 11.13.2. *Suppose that $S \geqslant_{\mathrm{T}} \emptyset'$. Then there is a 1-random set A such that $A' \equiv_{\mathrm{T}} A \oplus \emptyset' \equiv_{\mathrm{T}} S$.*

The situation for Δ^0_2 1-randoms is less clear. That is because there is no Lemma 11.13.1 where we *additionally* ask that $A \leqslant \emptyset'$. In fact, Cenzer [41] observed that there are Π^0_1 classes with no computable members[3] such that every member is GL_1, and hence all the Δ^0_2 members are low.

To prove jump inversion for Δ^0_2 random reals, we replace Lemma 11.13.1 with is a similar basis theorem for *fat* Π^0_1 classes.

Theorem 11.13.3 (Kučera [157], Downey and Miller [90][4])**.** *If $\mathcal{P}$ is a Π^0_1 class such that $\mu(\mathcal{P}) > 0$, then $S \geqslant_{\mathrm{T}} \emptyset'$ is Σ^0_2 iff there is a Δ^0_2 real $A \in \mathcal{P}$ such that $A' \equiv_{\mathrm{T}} S$.*

Corollary 11.13.4. *For every Σ^0_2 set $S \geqslant_{\mathrm{T}} \emptyset'$, there is a 1-random real $A \in \Delta^0_2$ such that $A' \equiv_{\mathrm{T}} S$.*

Proof. The proof of Theorem 11.13.3 can be viewed as a finite injury construction relative to $\emptyset'$. In that sense, it is similar to Sacks' construction of a minimal degree below $\mathbf{0}'$ [259]. We require two additional ideas from the literature. The first is the method of forcing with Π^0_1 classes, which was introduced by Jockusch and Soare [136] to prove the low basis theorem. This method is used to ensure that $A' \leqslant_{\mathrm{T}} S$. The second is a version of a lemma of Kučera [155] which allows us to recursively bound the positions of branchings in a Π^0_1 class with nonzero measure. The lemma allows us to code S into A' using a variation of a process known as Kučera coding. (We give a proof of this result as it is a variation of the Kučera coding technique.)

Lemma 11.13.5 (Kučera [155])**.** *Let $\mathcal{P}$ be a Π^0_1 class such that $\mu(\mathcal{P}) > 0$. Then there a Π^0_1 subclass $\mathcal{Q} \subseteq \mathcal{P}$ and a computable function $g\colon \omega \to \omega$ such that $\mu(\mathcal{Q}) > 0$ and*

$$(\forall e)\ \mathcal{Q} \cap \mathcal{P}_e \neq \emptyset \implies \mu(\mathcal{Q} \cap \mathcal{P}_e) \geqslant 2^{-g(e)}.$$

Proof. Let g be any computable function such that $\sum_{e \in \omega} 2^{-g(e)} < \mu(\mathcal{P})$. Let $\mathcal{Q}$ be the Π^0_1 subclass of $\mathcal{P}$ obtained by removing the reals in $\mathcal{P}_e[s]$ (th e stage s approximation to $\mathcal{P}_e$) whenever $\mathcal{P}_e[s] \cap \mathcal{Q}[s]$ has measure less than $2^{-g(e)}$. The choice of g guarantees that $\mu(\mathcal{Q}) > 0$. □

[3] Specifically, a *thin* perfect class (see Chapter 19).

[4] This results was stated without proof in Kučera [157], where he had constructed a high incomplete 1-random real.

Even though randomness is not explicit in the statement of Kučera's lemma, it is worth pointing out its implicit presence. Lemma 11.13.5 is proved using concepts from randomness and its main application has been to the study of the Turing degrees of 1-randoms.

Proof of Theorem 11.13.3. We are given a Π^0_1 class $\mathcal{P} \subseteq 2^\omega$ with nonzero measure and a Σ^0_2 set $S \geqslant_{\mathrm{T}} \emptyset'$. Take the Π^0_1 class $\mathcal{Q} \subseteq \mathcal{P}$ and computable function $g \colon \omega \to \omega$ guaranteed by Lemma 11.13.5. We will construct a Δ^0_2 real $A \in \mathcal{Q}$ such that $A' \equiv_{\mathrm{T}} S$.

Before describing the construction we must give a few preliminary definitions. For every $\sigma \in 2^{<\omega}$, define a Π^0_1 class

$$\mathcal{F}_\sigma = \{B \in \mathcal{Q} \mid (\forall e < |\sigma|)\ \sigma(e) = 0 \implies \varphi^B_e(e)\uparrow\}.$$

At each stage $s \in \omega$ of the construction, we will define a string $\sigma_s \in 2^s$ which is intended to approximate $A' \upharpoonright s$. We will tentatively restrict A to the class $\mathcal{F}_{\sigma_s}$ in order to *force its jump.* It is important to note that this restriction may be injured at a later stage by the enumeration of an $e < s$ into S.

Next we define a computable function $f \colon \omega \to \omega$ which grows fast enough to ensure that it (eventually) bounds the branchings between elements of $\mathcal{F}_\sigma$, for every $\sigma \in 2^{<\omega}$. We will use f to code elements of S into A (or more precisely, into A'). Let $h \colon 2^{<\omega} \times 2^{<\omega} \to \omega$ be a computable function such that $\mathcal{P}_{h(x,\sigma)} = [x] \cap \mathcal{F}_\sigma$, for all $x, \sigma \in 2^{<\omega}$. Set $f(0) = 0$. For $s \in \omega$, inductively define

$$f(s+1) > \max\{g(h(x,\sigma)) \mid x \in 2^{f(s)} \text{ and } \sigma \in 2^s\}.$$

Now take $x \in 2^{f(t)}$ and $\sigma \in 2^s$, for $t \geqslant s$, such that $[x] \cap \mathcal{F}_\sigma \neq \emptyset$. We claim that x has distinct finite extensions $y_0, y_1 \in 2^{f(t+1)}$ which extend to reals in $\mathcal{F}_\sigma$. Assume not. Let $\widehat{\sigma} = \sigma 1^{t-s}$ and note that $\mathcal{F}_{\widehat{\sigma}} = \mathcal{F}_\sigma$. Then $\mu(\mathcal{Q} \cap \mathcal{P}_{h(x,\widehat{\sigma})}) = \mu([x] \cap \mathcal{F}_\sigma) \leqslant 2^{-f(t+1)} < 2^{-g(h(x,\widehat{\sigma}))}$, which contradicts the lemma.

Kučera used the fact that we can bound branchings in a Π^0_1 class with nonzero measure to code information into members of such a class. The most basic form of Kučera coding constructs a real by extensions, choosing the leftmost or rightmost permissible extension to encode the next bit. For our construction, we only distinguish between the rightmost extension and any other permissible extension. Let $\mathcal{R}$ be a Π^0_1 class and let $x \in 2^{f(s+1)}$, for some $s \in \omega$. Define

$$H_f(\mathcal{R}; x) \text{ if and only if } (\exists n)[\text{if } y \in 2^{f(s)} \text{ is to the right of } x \upharpoonright f(s), \text{ then } \mathcal{R}[n] \cap [y] = \emptyset],$$

where $\mathcal{R}[n]$ is the approximation to $\mathcal{R}$ at stage n. Note that $H_f(\mathcal{R}; x)$ is a Σ^0_1 condition. By compactness, if $\mathcal{R} \cap [y] = \emptyset$, then there is a $n \in \omega$ such that $\mathcal{R}[n] \cap [y] = \emptyset$. This implies that if $\mathcal{R} \cap [x] \neq \emptyset$, then $H_f(\mathcal{R}; x)$ is true iff x is the rightmost length $f(s+1)$ extension of $x \upharpoonright f(s)$ which extends to an element of $\mathcal{R}$.

It will be useful to understand the interaction between f and H_f. Assume that we have $x \in 2^{f(t)}$, for some $t \geqslant s$, and $\sigma \in 2^s$ such that $[x] \cap \mathcal{F}_\sigma \neq \emptyset$. Let $\widehat{x} \in 2^{f(t+1)}$ be the leftmost extension of x such that $[\widehat{x}] \cap \mathcal{F}_\sigma \neq \emptyset$. By the definition of f, there are multiple extensions to choose from, so $H_f(\mathcal{F}_\sigma, \widehat{x})$ is false. In fact, if $\tau \preccurlyeq \sigma$, then $\mathcal{F}_\sigma \subseteq \mathcal{F}_\tau$. Therefore, $H_f(\mathcal{F}_\tau, \widehat{x})$ is also false.

We are now ready to describe the construction. Let $\{S_s\}_{s\in\omega}$ be a $\emptyset'$-computable enumeration of the Σ^0_2 set S. We may assume that $S_0 = \emptyset$ and $|S_{s+1} \smallsetminus S_s| = 1$ for all $s \in \omega$. We construct A by initial segments using a $\emptyset'$ oracle. At each stage $s \in \omega$, we find a string $x_s \in 2^{f(t)}$, for some $t \geqslant s$. Define $A = \bigcup_s x_s$. Each stage also produces a string $\sigma_s \in 2^s$, which is an approximation to A', although not necessarily an initial segment of it. For each $s \in \omega$, we require that

1. $[x_s] \cap \mathcal{F}_{\sigma_s} \neq \emptyset$.
2. If $\tau = \sigma_s \upharpoonright e+1$, for $e < s$, and $B \in [x_s] \cap \mathcal{F}_\tau$, then $B'(e) = \tau(e)$.

The Construction.

Stage 0. Let x_0 and σ_0 both be the empty string. Then $[x_0] \cap \mathcal{F}_{\sigma_0} = \mathcal{Q} \neq \emptyset$, so (1) is satisfied. Note that (2) is vacuous.

Stage $s+1$. Assume that we have already constructed $x_s \in 2^{f(t)}$, for some $t \geqslant s$, and $\sigma_s \in 2^s$ satisfying (1) and (2). Let $e \in S_{s+1} \smallsetminus S_s$ (this element is unique).

Case 1. If $e > s$, then let $x_{s+1} \in 2^{f(t+1)}$ be the leftmost extension of x_s such that $[x_{s+1}] \cap \mathcal{F}_{\sigma_s} \neq \emptyset$. Note that $\emptyset'$ can determine if $[y] \cap \mathcal{F}_{\sigma_s} = \emptyset$, for each $y \in 2^{<\omega}$, so $\emptyset'$ can find x_{s+1}. If $[x_{s+1}] \cap \mathcal{F}_{\sigma_s 0} \neq \emptyset$, then let $\sigma_{s+1} = \sigma_s 0$. Otherwise, let $\sigma_{s+1} = \sigma_s 1$. Again, this can be determined using the $\emptyset'$ oracle. Note that (1) and (2) are satisfied by our choices of x_{s+1} and σ_{s+1}.

Case 2. If $e \leqslant s$, then let $\tau_e = \sigma_s \upharpoonright e$. Consider the least number $m \in \omega$ such that $f(\langle e, m\rangle) \geqslant |x_s|$. First define $\widehat{x}_s \in 2^{f(\langle e,m\rangle)}$ be the leftmost extension of x_s such that $[\widehat{x}_s] \cap \mathcal{F}_{\tau_e} \neq \emptyset$. Next let $x_{s+1} \in 2^{f(\langle e,m\rangle+1)}$ be the rightmost extension of $\widehat{x}_s$ such that $[x_{s+1}] \cap \mathcal{F}_{\tau_e} \neq \emptyset$. Now inductively define τ_k, for each $e < k \leqslant s+1$. If we have already defined τ_k, then determine if $[x_{s+1}] \cap \mathcal{F}_{\tau_k 0} \neq \emptyset$. If so, let $\tau_{k+1} = \tau_k 0$. Otherwise, let $\tau_{k+1} = \tau_k 1$. Finally, set $\sigma_{s+1} = \tau_{s+1}$. Again, the construction is computable relative to $\emptyset'$ and we have ensured that (1) and (2) continue to hold.

End Construction.

We turn to the verification. The construction is computable from a $\emptyset'$ oracle, so A is Δ^0_2. Furthermore, (1) tells us that $[x_s] \cap \mathcal{F}_{\sigma_s} \neq \emptyset$, for each $s \in \omega$. Because $\mathcal{F}_{\sigma_s} \subseteq \mathcal{Q} \subseteq \mathcal{P}$, this implies that every x_s can be extended to an element of $\mathcal{P}$. But $\mathcal{P}$ is closed, so $A = \bigcup_s x_s \in \mathcal{P}$. All that remains to verify is that $A' \equiv_{\mathrm{T}} S$.

First we prove that $A' \leqslant_{\mathrm{T}} S$. To determine whether $e \in A'$, use S and $\emptyset'$ to find a stage $s > e$ such that $S_s \upharpoonright e+1 = S \upharpoonright e+1$. Let $\tau = \sigma_s \upharpoonright e+1$. We claim that $\sigma_t \upharpoonright e+1 = \tau$, for all $t \geqslant s$. This is because the only way that $\sigma_s \upharpoonright e+1$ can be injured during the construction is in Case 2, when an element $i \leqslant e$ is enumerated into S. But this will never happen after stage s. Therefore, for all $t \geqslant s$, we have $\tau \preccurlyeq \sigma_t$ and hence $\mathcal{F}_{\sigma_t} \subseteq \mathcal{F}_\tau$. So $[x_t] \cap \mathcal{F}_\tau \neq \emptyset$, for all $t \geqslant s$, which implies that $A \in \mathcal{F}_\tau$. By (2), we have $A'(e) = \tau(e)$. This proves that we can uniformly decide if $e \in A'$ using only $S \oplus \emptyset' \equiv_{\mathrm{T}} S$. Therefore, $A' \leqslant_{\mathrm{T}} S$.

Now we must show that $S \leqslant_{\mathrm{T}} A'$. Assume by induction that we have determined $S \upharpoonright e$, for some $e \in \omega$. Find the least $s \geqslant e$ such that $S_s \upharpoonright e = S \upharpoonright e$. Let $\tau = \sigma_s \upharpoonright e$ and note, as above, that $\tau \preccurlyeq \sigma_t$, for all $t \geqslant s$. Find the least $m \in \omega$ such that $f(\langle e, m\rangle) \geqslant |x_s|$. Of course, both s and m can be found by $\emptyset'$. We claim that $e \in S$ iff either $e \in S_s$ or $(\exists n \geqslant m)\ H_f(\mathcal{F}_\tau; A \upharpoonright f(\langle e, n\rangle + 1))$. If $e \in S \setminus S_s$, then Case 2 ensures that $H_f(\mathcal{F}_\tau; A \upharpoonright f(\langle e, n\rangle + 1))$ holds for some $n \geqslant m$. So, assume that $e \notin S$. Then for every $n \geqslant m$, the construction chooses the leftmost extension of $A \upharpoonright f(\langle e, n\rangle)$ which is extendible in the appropriate Π^0_1 class. This class is of the form $\mathcal{F}_{\widehat{\tau}}$, where $\widehat{\tau} \preccurlyeq \sigma_t$ for some $t \geqslant s$ and $|\widehat{\tau}| \geqslant e$. This implies that $\tau \preccurlyeq \widehat{\tau}$, so $\mathcal{F}_{\widehat{\tau}} \subseteq \mathcal{F}_\tau$. The definition of f ensures that there are distinct length $f(\langle e, n\rangle + 1))$ extensions of $A \upharpoonright f(\langle e, n\rangle)$ which can be extended to elements of $\mathcal{F}_{\widehat{\tau}}$. Therefore, the leftmost choice consistent with $\mathcal{F}_{\widehat{\tau}}$ must be left of the rightmost choice consistent with $\mathcal{F}_\tau$. Hence $H_f(\mathcal{F}_\tau; A \upharpoonright f(\langle e, n\rangle + 1))$ is false. Finally, note that A' can decide if $(\exists n \geqslant m)\ H_f(\mathcal{F}_\tau; A \upharpoonright f(\langle e, n\rangle + 1))$, because H_f is Σ^0_1. Therefore A' can determine if $e \in S$, proving that $S \leqslant_{\mathrm{T}} A'$. □

□

11.14 Psuedo-jump Inversion

In the same way that the construction of a low c.e. set was generalized by Jochusch and Shore [**?**, **?**], Nies realized that Theorem 11.13.3 can be extended to a similar pseudo-jump theorem. Recall that an operator of the form $V^Y = Y \oplus W^Y$ for a c.e. set W with $Y < V^Y$ for all oracles Y is called a nontrivial pseudo-jump operator. The following is a randomness version of Theorem 5.10.3.

Theorem 11.14.1 (Nies [**?**])**.** *For any nonttivial pseudo-jump operator V there is a 1-random set A with $V^A \equiv_T \emptyset'$.*

11.15 Randomness and genericity

11.15.1 Genericity and weak genericity

Schnorr's Theorem gives the classical interpretation of randomness, as a notion of *typicalness in terms of measure.* Another notion of typicalness is in terms of Baire category. These are the *generic* set which are members of every dense open (suitably definable) class.

In this section, we will look at the such a notion of typicalness, and see how it relates to measure theoretical typicalness. The answer, surprisingly, is that genericity and randomness are more or less orthogonal especially when we look at arithmetical classes with more than one quantifier. One the other hand, there is a notion of forcing, Solovay forcing, which is related. This in turn gives insight into the both randomness and genericity itself.

The original definition of genericity was in terms of forcing. For completeness let $\mathcal{L}$ be the usual first order language for number theory together with a set constant X and the membership relation $\in$.

Definition 11.15.1 (Cohen, Feffermann)**.** Let φ be a sentence of $\mathcal{L}$ and $\sigma \in 2^{<\omega}$. We say that σ *forces* φ, written $\sigma \Vdash \varphi$, if:

(i) φ is atomic and does not contain X, and φ is true in arithmetic.

(ii) $\varphi \equiv (n \in X)$ and $\sigma(n) = 1$.

(iii) $\varphi \equiv (\varphi_1 \vee \varphi_2)$ and $\sigma \Vdash \varphi_1$ or $\sigma \Vdash \varphi_2$,

(iv) $\varphi \equiv \overline{\psi}$ and $\forall \widehat{\sigma} \supseteq \sigma(\widehat{\sigma} \nVdash \varphi)$, and

(v) $\varphi \equiv (\exists z(\psi(z))$ and there is an $n \in \mathbb{N}$ such that $\sigma \Vdash \psi(n)$.

If A is a real and φ is a sentence of $\mathcal{L}$, then we say that $A \Vdash \varphi$ if some initial segment of A forces φ.

Definition 11.15.2. A set A is called *n-generic* iff for all Σ^0_n sentence φ of $\mathcal{L}$, either $A \Vdash \varphi$ or $A \Vdash \overline{\varphi}$.

The Σ^0_n subsets of $\mathbb{N}$ have Σ^0_n definitions in $\mathcal{L}$, by Matijacevic's[223] Theorem. A new direction in these studies was initiated by Jockusch and Posner:

Theorem 11.15.3 (see Jockusch [133])**.** *A is n-generic iff for all Σ^0_n sets of strings V either*

(i) there is a $\sigma \in V$ such that $\sigma \prec A$, or

(ii) there is an initial segment τ of A such that for all $\sigma \in V$, $\tau \nsubseteq \sigma$.

Proof. For (ii) implies (i), let φ be the relevant sentence, and let $A = \{\sigma : \sigma \Vdash \varphi\}$. Then A is Σ^0_n.

For the reverse direction, given A let $Q = \{B \subseteq \mathbb{N} : \exists \sigma(\sigma \in A \wedge \sigma \subset B\}$. Let φ be the sentence of $\mathcal{L}$ that defines Q. Let σ be a string which forces φ or forces $\overline{\varphi}$. If $\sigma \Vdash \varphi$ then $A \in Q$. If $\sigma \Vdash \overline{\varphi}$, then then no extension of σ is in A. $\square$

This allows one to easily construct n-generic Δ^0_{n+1} sets, and arithmetically generic sets (i.e. n-generic for each n) computable from $\emptyset^($\omega$)$. One defines a *degree* to be n-generic iff it contains a n-generic set. A very interesting fact about generic degrees (i.e. arithmetical generic degrees), is that *for* $\mathbf{a}$ *and* $\mathbf{b}$ *generic, the structures of the degrees below* $\mathbf{a}$ *and* $\mathbf{b}$ *are elementary equivalent* (Jockusch [133]).

Theorem 11.15.4 (Jockusch [133])**.** *If A is n-generic then $A^{(n)} \equiv_{\mathrm{T}} A \oplus \emptyset^{(n)}$.*

Proof. Consider the Σ^0_n formula $\varphi(x, A)$ saying "$x \in A^{(n)}$." Then $k \in A^{(n)}$ iff $\exists\sigma(\sigma \prec A \wedge \sigma \Vdash \varphi(n, A))$. Thus $A^{(n)}$ is computably enumerable in $A \oplus \emptyset^{(n)}$. Using $\overline{\varphi}$ in place of φ shows that $A^{(n)} \leqslant_{\mathrm{T}} A \oplus \emptyset^{(n)}$. $\square$

We will mainly use the characterization of Theorem 11.15.3 henceforth, as it is more suited to our purposes.

There are many results known to hold for n-generic reals. The comprehensive survey Jockusch [133] list many classical results and Kumabe's Thesis [**?**] has a wealth of results. As is well known, suffiently generic reals charcaterize amny finite extension arguments. For instance the following is the well known generalization of Friedberg's completeness criterion.

Theorem 11.15.5. *Suppose that $y \geqslant x \oplus \emptyset'$. Then there is a 1-generic z such that $z \oplus x \equiv_T y$.*

We wish to weaken the definition of genericity.

Definition 11.15.6. A set Q of strings is called *dense* iff for all strings σ there is a strings $\tau \in Q$ such that $\sigma \subseteq \tau$.

Kurtz noted that is Q is a dense Σ^0_n set of strings and A is n-generic then A meets Q in the sense that

$$\exists\sigma \prec A(\sigma \in Q).$$

Definition 11.15.7 (Kurtz [165])**.** A real is called *weakly* n-generic iff it meets all dense Σ^0_n sets of strings.

It is also easy to show that A is weakly $n+1$-generic iff A meets all dense Δ^0_{n+1} sets of strings iff A meets all dense Π^0_n sets of strings. (Jockusch, see [165])

Kurtz remarked that the above definition of forcing was a direct analog of the set theoretical definition.

Theorem 11.15.8 (Kurtz [165])**.** *If A is weakly $n+1$-generic then A is n-generic.*

Proof. This is similar to the proof that Kurtz $n+1$-randomness implies n-randomness. Let S be a Σ^0_n set of strings. Now consider

$$S' = \{\sigma : \sigma \in S \vee \forall\tau(\tau \supseteq \sigma \rightarrow \tau \notin S)\}.$$

Then S' is a dense Σ^0_{n+1} set of strings, which A will meet if it is weakly $n+1$-generic. $\square$

We remark that many theorems can bee seen as manifestations of general results on forcing and hence analogs in both randomness and genericity. For instance, the following is well known.

Theorem 11.15.9 (see Jockusch [133])**.** *Suppose that $x \oplus y$ is n-generic. The x and y are n-generic.*

Proof. Suppose that $x \oplus y$ is n-generic. Suppose that S is a Σ^0_n set of strings which does not meet x yet is dense in x. Now let $\widehat{S} = \{\sigma \oplus \tau : \tau \in 2^{<\omega}\}$. Then $\widehat{S}$ would be a Σ^0_n set in which $x \oplus y$ is dense and does not meet $x \oplus y$ lest S meet x. □

One nice example is the following.

Theorem 11.15.10 (Yu [**?**])**.** *van Lambalgen's Theorem holds for genericity. That is, for any $n \geqslant 1$, $x = x_1 \oplus x_2$ is n-generic iff x_1 is $n - x_2-$generic and x_2 is $n - x_1-$generic.*

Proof. Suppose frst that x_2 is $n - x_1-$generic. Let S be a Σ^0_1 class and define $\widehat{S} = \{\tau : \exists \sigma \preccurlyeq \tau(\sigma \oplus \tau \in S)\}$. Then $\widehat{S}$ is a $\Sigma^{x_1}_1$ class. Since x_2 is $n - x_1-$generic, either $\exists \tau \prec x_2$ with no extension in $\widehat{S}$ or $\exists \tau \prec x_2$ with $\tau \in \widehat{S}$. In the latter case, $\sigma \oplus \tau \in S$. In the former, take such a τ. Let $T = \{\sigma : (x \restriction 2|\tau|) \prec \sigma \oplus \tau' \text{ and } \sigma \oplus \tau' \in S\}$. Then no member of T is an initial segment of x_1. For if $\sigma' \prec x_1$ then there is a τ' with $\tau \prec \tau'$ and $\sigma' \oplus \tau' \in T$. Thus $\tau' \in \widehat{S}$, a contradiction. Therefore there is a $\sigma' \prec x_1$ with no extension in T. Then $x \restriction 2|\sigma'|$ has no extension in S, as required. Hence x is $n-$generic.

Conversely suppose that $x = x_1 \oplus x_2$ is $n-$generic. We already know that x_1 is $n-$generic. Suppose that x_2 is not $x_1 - n-$generic. Choose a $\Sigma^{x_1}_1$ set S such that no member of S is an initial segment of x_2 and every initial segment of x_2 has an extension in S. There is a Σ^0_n formula Φ with $S = \{\sigma : \exists n \Phi(x_1 \restriction n, \sigma)\}$. Then we can define

$$\widehat{S} = \{\sigma \oplus (\tau 0^{(|\sigma| - |\tau|)} : |\tau| \leqslant |\sigma| \wedge \Phi(\sigma, \tau)\}.$$

Then $\widehat{S}$ is Σ^0_n. Notice that, for all $\sigma \oplus \tau \prec x(\sigma \oplus \tau \notin \widehat{S})$. If not then there would be some $\tau \prec x_2(\tau \in S)$, a contradiction. However, because $S = \{\sigma : \exists n \Phi(x_1 \restriction n, \sigma)\}$, we see that for all $\sigma \oplus \tau \prec x$, there exists a $\sigma' \oplus \tau'$ extending $\sigma \oplus \tau$ with $\sigma' \oplus \tau' \in \widehat{S}$. This contradicts the fact that x is $n-$generic. □

One final example of analogues is an analog to Theorem **??**

Theorem 11.15.11 (Csima, Downey, Greenberg, Hirschfeldt, Miller [**?**])**.** *Let B be 2-generic, computable in A, which is 1-Z-generic. Then B is 1-Z-generic over Z.*

Proof. Let A be 1-generic over Z and let $B \leqslant_T A$ be 2-generic. Let Φ be a Turing functional such that $\Phi(A) = B$. Let $W \subset 2^{<\omega}$ be c.e. in Z; we may assume that W is closed upwards.

Suppose that $B \notin \mathcal{W}$. Let $\tilde{W} = \{\sigma \in 2^{<\omega} : \Phi(\sigma) \in W\}$. Certainly $\tilde{W}$ is c.e. in Z. Since $A \notin \tilde{\mathcal{W}}$, we know that there is some $\sigma^* \subset A$ with no extension in $\tilde{W}$.

Let $U = \{\tau \in 2^{<\omega} : \neg\exists\sigma \supset \sigma^* \, [\tau \subset \Phi(\sigma)]\}$. U is co-c.e., and $B \notin \mathcal{U}$, so there is some $\tau^* \subset B$ with no extension in U. Then τ^* has no extension in W either. □

Choosing $Z = \emptyset^k$ in the above yields the following corollary.

Corollary 11.15.12 (Csima, Downey, Greenberg, Hirschfeldt, Miller [**?**]). *Suppose that $n \geqslant 2$ and $A \leqslant_T B$ with A 2-generic and B n-generic. Then A is n-generic.*

The reader might wonder if the full analog of Theorem **??** holds, namely if a 1-generic below a 2-generic is automatically 2-generic. The answer is no, no matter how generic the top set may be!

Theorem 11.15.13 (Csima, Downey, Greenberg, Hirschfeldt, Miller [**?**]). *Every 1-generic real computes a properly 1-generic real. In fact, Every 1-generic real computes a 1-generic real that is not weakly 2-generic.*

Proof. To do this, we construct a Turing functional Γ which has the following properties:

1. There is a dense $\Pi^0_2(\emptyset')$ class which is contained in the domain of Γ, and whose range by Γ consists of 1-generic sets.
2. The range of Γ is contained in a nowhere dense $\Pi^0_1(\emptyset')$ class.

To see that this suffices, let X be weakly 2-generic (if X is properly 1-generic we are of course done.) Let $\mathcal{A}$ be the class guaranteed by (1). X is the element of any dense $\Sigma^0_1(\emptyset')$ class, hence in any countable intersection of such (recall Baire's theorem that states that an intersection of countably many dense, open sets is dense); so $X \in \mathcal{A}$. Then $\Gamma(X)$ (which is computable by X) is 1-generic by (1). It is not weakly 2-generic because by (2), it misses a dense open set enumerable by $\emptyset'$.

Getting property 1 To make the range of $\mathcal{A}$ under Γ consist of 1-generic sets, for every e, we must construct a dense, open set $\mathcal{S}_e$ such that for all $\sigma \in S_e$, $\Gamma(\sigma)$ decides W_e; and further we must make the sequence $\mathcal{S}_0, \mathcal{S}_1, \dots$ uniformly enumerable by $\emptyset'$.

Consider W_0. A simple plan for meeting it would be setting $\mathcal{S}_0 = 2^\omega$ and acting as follows: if W_0 is empty, do nothing; if there is some $\tau \in W_0$, let $\Gamma(\langle\rangle) = \tau$. Now move to W_1. Of course, this is not effective, so we must use the priority method for our construction. Again, a naïve approach would be as follows: While W_0 is empty, do nothing, and let weaker requirements $(W_1, W_2, \dots)$ act if they want to. If some string τ enters W_0 then injure the weaker requirements and set $\Gamma(\langle\rangle) = \tau$. The problem here is that we cannot cancel the axioms that work for $W_1, W_2, \ldots$ had us enumerating into Γ, so if we want to keep Γ consistent, we cannot make the definition we like. The solution is to break up the playing ground into pieces, let weaker requirements work on some of the pieces, and make sure that there is sufficiently much room for the stronger requirement to act if necessary.

Here is the strategy for W_0. In the beginning, we mark the interval $2^\omega = [\langle\rangle]$ to work on W_0. We break the interval up into infinitely many disjoint subintervals whose union is dense in 2^ω, say $[1], [01], [001], [0001], \ldots$. For the time being, each such subinterval believes it has met the W_0-requirement by forcing into the complement of $\mathcal{W}_0$, simply because $\mathcal{W}_0$ is still empty. So we can be generous and let each subinterval work for the next requirement W_1.

At a later stage, some string τ enters W_0. Only finitely many subintervals have been spoiled for W_0; so we can define Γ to be τ everywhere else. On the spoiled intervals, we need to work again for W_0; since definitions of Γ have been made on possibly small subsubintervals, we need to break the spoiled region into small intervals on which we individually work on W_0.

We let S_0 be the collection of intervals σ that are "good" for W_0: these are those intervals on which we ensures that $\Gamma(\sigma)$ meets W_0, or those at which we had a correct belief that $[\Gamma(\sigma)] \cap \mathcal{W}_0 = \emptyset$. This set will in fact be d.c.e., so is certainly Σ^0_2; and it satisfies the W_0 requirement. Since we break it up into finer and finer subintervals (each time our hopes for an easy win are dashed), $\mathcal{S}_0$ ends being dense.

The strategy for weaker W_e is similar, except that of course we need to take into consideration injury by stronger requirements.

Getting property 2 To ensure that the range of Γ is nowhere dense, we could, whenever we define some axiom $\Gamma(\sigma) = \tau$, pick some extension τ' of τ and declare that no value of Γ may ever extend τ'. This straightforward approach, however, interferes with the priority mechanism that ensures property (1), in the following way. Suppose that we mark some interval $[\sigma_0]$ for W_1, and later define $\Gamma(\sigma_1) = \tau$ for some $\sigma_1 \supset \sigma_0$, marking σ_1 for W_2. We then declare that the range of Γ must be disjoint from $[\rho]$, where $\rho \supset \tau$. A later W_0 action elsewhere invalidates σ_0's marking, so we mark $[\sigma_1]$ for W_0. Then, some string extending ρ enters W_0; W_0 is prohibited from winning by directing Γ through that string on a subinterval of σ_1. We will indeed direct Γ to go through some extension τ' of τ which is incomparable with ρ, and this presumably will give us another chance of attacking W_0; but this process may repeat itself, since following the straightforward approach compels us to first declare some extension ρ' of τ' disjoint from the range of Γ. After infinitely many failed attempts at meeting W_0 we have Y in the range of Γ belonging to the closure of $\mathcal{W}_0$ but not to $\mathcal{W}_0$ itself.

This in fact *must* happen, because we made the collection of prohibited intervals a dense c.e. class, ensuring that no element of the range of Γ is even weakly 1-generic (indeed, the recursion theorem and the "slowdown lemma" imply that there is some e such that W_e is the set of prohibited intervals, and that every σ is enumerated into W_e only after it was declared prohibited.) The solution is to use the priority mechanism that was introduced for getting property (1). When we define $\Gamma(\sigma) = \tau$ for meeting W_e, we define one extension to be prohibited with priority e. This prohibition can be ignored by strings $\sigma' \subset \sigma$ that are working for stronger $W_{e'}$. The whole mechanism does the work for us, so we in fact don't need to use the word "prohibited" during the construction, just to make $\Gamma(\sigma)$ long enough.

Construction

In the beginning, the entire space $2^\omega = [\langle\rangle]$ is marked *active* for W_0.

At *stage s*:

1. A string τ is enumerated into some W_e. Suppose that there is some σ, marked active for W_e, such that $\Gamma(\sigma) \subset \tau$ (there will be at most one such σ). Do the following:

- Unmark σ from being active for W_e. Choose a very large number m.
- Enumerate $(\sigma 0^s, \tau 1^m)$ into Γ. Mark $[\sigma 0^s]$ as active for W_{e+1}.
- For every $k < s$, remove all markings of all strings $\sigma' \supseteq \sigma 0^k 1$. For every $\sigma' \in 2^m$ extending $\sigma 0^k 1$ for some $k < s$, mark σ' as active for W_e; find some $\tau_{\sigma'} \supset \Gamma(\sigma')$ of length m that is incompatible with $\tau 1^m$, and enumerate $(\sigma', \tau_{\sigma'}\sigma')$ into Γ.

2. Inductively for $e < s$, for every $[\sigma]$ that is marked as active for W_e, for $k \leqslant s$, mark $[\sigma 0^k 1]$ as active for W_{e+1}, and enumerate $(\sigma 0^k 1, \Gamma(\sigma) 0^k 11)$ into Γ.

Verification

Let $e < \omega$. Suppose that at some stage s, an interval $[\sigma]$ is marked active for W_e. Then for all $k < s$, we let $\sigma 0^k 1 \in S_e[s]$ (and call $\sigma 0^k 1$ *good* for W_e.) If at some stage s, action is taken for σ (that is, we define Γ on $\sigma 0^s$), then we extract $\sigma 0^k 1$ (for $k < s$) from S_e and enumerate $\sigma 0^s$ into S_e. This interval will later be extracted from S_e only if action is taken for some interval containing $[\sigma]$ (and for some requirement stronger than e). The sets S_e are (uniformly) d.c.e., and so $\mathcal{A} = \cap_e \mathcal{S}_e$ is a Π^0_2 class.

For simplicity of notation, we let (at every stage) $S_{-1} = \{\langle\rangle\}$.

Lemma 11.15.14. *Each $\mathcal{S}_e$ is dense.*

(And so $\mathcal{A}$ is dense too.)

Proof. We take some $\sigma \in S_{e-1}$ and show that $\mathcal{S}_e$ is dense in $[\sigma]$. Suppose that σ is put into S_{e-1} at stage s_0.

We first note that if some $[\sigma'] \subset [\sigma]$ is ever marked active for W_e at a stage $s \geqslant s_0$, then there is a subinterval of $[\sigma']$ that is (permanently) in S_e. This is because either no action is taken for σ', in which every $\sigma' 0^k 1$ is in S_e, or action is taken for σ' at some stage s, in which case $\sigma' 0^s$ is in S_e.

The point is that these markings cannot be eliminated by action below σ', because such action would remove σ from S_{e-1}.

At every stage $s > s_0$ we have certain density: for every $\rho \in 2^{<s}$ there is some σ' in $S_e[s]$ that is compatible with ρ. Let $\rho \supset \sigma$ and suppose for contradiction that $[\rho] \cap \mathcal{S}_e = \emptyset$. Let $s_1 > s_0, |\rho|$, and find some $\sigma' \in S_e[s_1]$ compatible with ρ (so $\sigma' \supset \sigma$). At a later stage s_2, σ' is extracted from S_e; at that stage, all successors of σ' at some level $m > |\rho|$ are marked active for W_e. At least one of these successors σ'' is compatible with ρ, and so must actually extend ρ. σ'' has some extension that is in S_e, contradiction. □

Lemma 11.15.15. *Suppose that $X \in \mathcal{S}_e$. Then some initial segment of $\Gamma(X)$ determines W_e.*

(Note that we do not assume that $\Gamma(X)$ is total.)

Proof. Straightforward. Suppose that $\sigma \in S_e$ and $\sigma \subset X$. If $\sigma = \sigma' 0^s$ for some σ' that is marked active for W_e, then $\Gamma(\sigma)$ extends some $\tau \in W_e$. Otherwise, $\sigma = \sigma' 0^k 1$ for some σ' that is marked active for W_e (and this mark is never removed). Then no $\tau \supset \Gamma(\sigma')$ is ever enumerated into W_e. □

Corollary 11.15.16. *$\mathcal{A} \subset \mathrm{dom}\Gamma$ and every $Y \in \Gamma[\mathcal{A}]$ is 1-generic.*

Proof. For the first part, use $W_e = 2^{\geqslant n}$. □

Lemma 11.15.17. *$\Gamma[2^\omega]$ is contained in a nowhere dense $\Pi^0_1(\emptyset')$ class.*

Proof. Let T be he downward closure of the range of Γ, viewed as a relation on *strings*, i.e.

$$T = \{\tau : \exists\sigma\ [(\sigma, \tau) \in \Gamma]\}.$$

T is c.e., and so $[T]$, the class of paths through T, is a $\Pi^0_1(\emptyset')$ class that certainly contains the image of Γ on 2^ω.

$[T]$ is closed, so to show that it is nowhere dense, it suffices to show that it does not contain any interval.

For any σ, (σ, τ) is enumerated into Γ (for some τ) at most once, when σ is marked active for some W_e. Suppose for contradiction that $[\rho]$ is an interval contained in $[T]$, which means that every extension of ρ is in T. By definition of T, there are some σ and τ such that $(\sigma, \tau) \in T$ and $\tau \supset \rho$.

For almost all e, σ has extensions in S_e. One would suffice, because of the following two facts:

Lemma 11.15.18. *Suppose that $\sigma \in \cup_e S_e$. Then:*

1. *For all σ', $\Gamma(\sigma') \supset \Gamma(\sigma)$ implies that $\sigma' \supset \sigma$.*
2. *$\Gamma(\sigma) = \tau 1$ for some τ, and $\tau 0$ is not on T.*

Proof. For (1), note that at every stage s, if $[\sigma]$ and $[\sigma']$ are two disjoint intervals that are marked active (not necessarily for the same W_e), then $\Gamma(\sigma) \perp \Gamma(\sigma')$. □

$\square$

$\square$

11.16 Weakly n-generic and n-generic

The following theorem will be used to classify weakly $n+1$-generic degrees. Recall that a degree $\mathbf{a}$ is called *hyperimmune* with respect to $\mathbf{c}$ iff there is a function f computable from $\mathbf{a}$ that is not majorized by any function computable from $\mathbf{c}$.

Theorem 11.16.1 (Kurtz [165]). $\mathbf{b}$ *is the n-th jump of a weakly $n+1$-generic degree iff* $\mathbf{b} > \mathbf{0}^n$ *and* $\mathbf{b}$ *is hyperimmune with respect to* $\mathbf{0}^n$.

Proof. Assume that $\mathbf{b} = \mathbf{a}^n$ with $\mathbf{a}$ weakly $n+1$-generic. We prove that $\mathbf{b}$ is hyperimmune with respect to $\mathbf{0}^n$. Miller and Martin [218] proved that the hyperimmune degrees (with respect to $\mathbf{c}$) are closed upwards. Thus it suffices to show that $\mathbf{a}$ is hyperimmune with respect to $\mathbf{0}^n$. Let p_A denote the principal function of A, the function that enumerates A in increasing order, and p_σ the principal function of $\{n : \sigma(n) = 1\}$. Then let $S_f = \{\sigma : (n < \max\{k : \sigma(k) = 1\} \wedge p_\sigma > f(n)\}$. Then S_f is a dense Δ^0_{n+1} set. If A is weakly $n+1$-generic, and of degree $\mathbf{a}$, A must meet S_f. Therefore f cannot majorize p_A.

The other direction is more intricate. Again we follow Kurtz' proof. It is a variant of the Friedberg completeness criterion. Let $\mathbf{b}$ be above $\mathbf{0}^n$ and hyperimmune with respect to $\mathbf{0}^n$. Choose a sequence $\{f_i; i \in \omega\}$ or functions uniformly partial computable in $\mathbf{0}^n$ enumerating the collection $\{S_i : i \in \omega\}$ an acceptable enumeration of the Σ^0_{n+1} sets of strings. We ask that f_i is total iff S_i is nonempty. Note that f_i will either be empty or total. Let

$$S_i^s = \{\sigma : \exists k \leqslant p_B(s) : (f_i(k) = \sigma\}.$$

Since $\mathbf{b}$ is hyperimmune with respect to $\mathbf{0}^n$, there is a set $B \in \mathbf{b}$ such that p_B is not majorized by and function computable from $\mathbf{0}^n$. Using the finite extension method, we construct a sequence of strings σ_s with $A = \lim_s \sigma_s$ having $A^{(n)} \equiv_{\mathrm{T}} B$ and A weakly $n+1$-generic.

We meet the requirements

$$R_e : S_e \text{ dense implies } A \text{ meets } S_e.$$

We say that R_e *requires attention* if σ_s does not yet meet S_e^s, but there is a string $\tau \supseteq \sigma_s$ such that τ is of the form

$$\tau = \sigma_s \widehat{\ } 0 \widehat{\ } 1^{e+1} 0 \widehat{\ } \gamma,$$

and meets S_e^s.

The peculiar form that τ must have comes from the coding of B in $A^{(n)}$. The e is a marker as to which requirement is being actioned, and this will allow us to "unwind" the construction to recover the coding. We say that a stage s is a *coding stage* if we act to code information about B into A. Let $c(s)$ denote the number coding stages before s. We let $c(0) = 0$, and declare that 0 is a coding stage. Finally we set $\sigma_0 = B(0)$.
Construction, stage $s + 1$.

Case 1 No requirement R_e requires attention for $e \leqslant s + 1$.
Action: Let $\sigma_{s+1} = \sigma_s$. This is not a coding stage.

Case 2 Let R_e be the highest priority requirement. Let m_0 me least with

$$f_e(m) = \sigma_s \frown 0 \frown 1^{e+1} \frown 0 \frown \gamma,$$

for some γ.

Action: If $|f_e(m_0)| > s + 1$, then set $\sigma_{s+1} = \sigma_s$ and declare this as a non-coding stage. Otherwise, we set $\sigma_{s+1} = f_e(m_0) \frown B(c(s+1))$. Declare that $s + 1$ is a coding stage.

Lemma 11.16.2. *R_e receives attention only finitely often.*

Proof. Let s_0 be such that all the R_i for $i < e$ have ceased activity by stage s_0. Assume that $R - e$ receives attention at $t > s_0$. Let $f_e(m_0) \leqslant t$ then we define σ_t so that it meets S_e^t, and henceforth R_e will not again receive attention. The other case is that we have $|f_e(m_0)| > t$. Since all the R_i of higher priority have ceased activity, σ_u will remain fixed as σ_t until stage $|f_e(m_0)|$, unless we meet R_e through some other string. □

Lemma 11.16.3. *A is weakly $n + 1$-generic.*

Proof. Assume that S_e is dense. Define g as follows: $g(s)$ is the least k such that for all strings σ of length $\leqslant s + 1$, there is a string τ of the form

$$\sigma \frown 1^{e+1} \frown 0 \frown \gamma = f_e(j),$$

for some $j \leqslant k$. Then $g \leqslant_T \emptyset^{(n)}$. (Since S_e is dense and f_e is total.) Let s be a stage whereby R_e has priority. Since p_B is hyperimmune with respect to $\emptyset^{(n)}$, there is some stage $t > s$ where $p_B(t) > g(t)$. During such a stage R_e will require attention unless it is already met, and will keep requiring attention until it is met. □

Lemma 11.16.4. $A^{(n)} \leqslant_T B$.

Proof. The construction of A is computable from $\emptyset^{(n)}$, and B. Hence the construction is computable from B as $\emptyset^{(n)} \leqslant_T B$. By the previous lemma, A is weakly $n + 1$ generic and hence $A^{(n)} \leqslant A \oplus \emptyset^{(n)}$, and hence $B \geqslant_T A \oplus \emptyset^{(n)} \equiv_T A^{(n)}$. □

Lemma 11.16.5. $B \leqslant A^{(n)}$.

Proof. Note that A is infinite, as it is weakly $n+1$-generic. Elements are added to A only during coding stages, nd there are thus infinitely many such stages. During coding stage s_k we define σ_{s_k} with $\sigma_{s_k}(|\sigma_{s_k}|-1) = B(k)$. Thus it is enough to show that the function $g(k) = |\sigma_{s_k}$ is computable from $A^{(n)}$.

We know that $g(0) = 1$.. For $m \leqslant k$, assume that we have computed $g(m)$ using the $A^{(n)}$ oracle. Then we know what σ_{s_k} was. That is it is an initial segment of A of length $g(k)$. We know that A will next extend some string of the form $\sigma_k\widehat{\ }0\widehat{\ }1^{e+1}\widehat{\ }0\widehat{\ }\gamma$ for some e and γ. There fore we will satisfy some requirement R_e at stage $s_{k+1}+1$, and f_e is total. Since $\emptyset^{(n)} \leqslant A^{(n)}$, we can compute from $A^{(n)}$, the least m such that $f_e(m)$ is of the form $\sigma_k\widehat{\ }0\widehat{\ }1^{e+1}\widehat{\ }0\widehat{\ }\gamma$. Since $\sigma_{s_{k+1}} = \sigma_k\widehat{\ }0\widehat{\ }1^{e+1}\widehat{\ }0\widehat{\ }\gamma\widehat{\ }B(c(s_{k+1}))$, we see that $g(k+1) = |f_e(m)| + 1$. □

□

Corollary 11.16.6. *If A is weakly $n+1$-generic, then A is hyperimmune with respect to $\emptyset^{(n)}$.*

Corollary 11.16.7. *If A is weakly 2-generic then A is hyperimmune. If A is 1-generic then A is hyperimmune.*

Corollary 11.16.8. *There is an n-generic degree that is not weakly $n+1$-generic.*

Proof. There is an n-generic degree that is computable from $\emptyset^{(n)}$. No degree computable from $\mathbf{0}^n$ can be hyperimmmune with respect to $\mathbf{0}^n$. □

Corollary 11.16.9. *There is a weakly $n+1$-generic degree that is not $n+1$-generic.*

Proof. Since $\mathbf{0}^{(n+1)}$ is hyperimmune with respect to $\mathbf{0}^{(n)}$ it contains a weakly $n+1$ generic degree. But $\mathbf{0}^{(n+1)}$ cannot be $n+1$-generic since A $n+1$-generic implies that $A^{(n+1)} \equiv_{\mathrm{T}} A \oplus \emptyset^{(n)}$. □

11.16.1 n-generic vs n-random

So what about the interplay of n-genericity and n-randomness?

Theorem 11.16.10 (Kurtz [165]). *Every weakly 1-generic set is weakly 1-random.*

This gives the result mentioned earlier linking 1-randomness and hyperimmunity.

Corollary 11.16.11 (Kurtz [165]). *Every hyperimmune degree is Kurtz 1-random.*

Proof. (of Theorem 11.16.10) Suppose that G is weakly 1-generic. Let S be a measure 1 computably enumerable open set. There is a computably enumerable set of strings V such that $S = \{A : A \text{ meets } V\}$. Since S has measure 1, it is dense, and hence G meets V, and hence must be a member of S. $\square$

Corollary 11.16.11 only works in one direction. We have seen in Proposition 11.1.3 that there are 1-random reals of hyperimmune-free degree.

Actually more can be gleaned from Theorem 11.16.10.

Lemma 11.16.12. *No weakly 1-generic set is Schnorr random.*

Proof. Consider the set of strings $S = \cup_i S_i$, where $S_1 = \{00, 010, 0110, \dots\}$, and $S_{i+1} = \{\sigma\tau : \sigma \in S_i \wedge \tau \in S\}$. Then S is c.e. sand sense so that is A is weakly 1-generic then it meets S infinitely often. But $V_i = \cup\{[\sigma] : \sigma \in S_i\}$ is a Schnorr test, and hence A is not Schnorr random, since $A \notin \cap_i V_i$. $\square$

Corollary 11.16.13. *Every hyperimmune degree contains a Kurtz random real that is not Schnorr random.*

However, the following result shows that outside of the hyperimmune degrees all the randomness concepts coincide.

Theorem 11.16.14 (Nies, Stephan, Terwijn [232])**.** *Suppose that A is of hyperimmune-free degree. Then A is Kurtz random iff A is Martin-Löf random.*

Proof. Suppose that A has hyperimmune free degree, and A is Kurtz random. Suppose that A is not Martin-Löf random. Then since Then there is a Martin-Löf test $\{V_n : n \in \mathbb{N}\}$, such that $A \in \cap_n V_n$. Using A we can compute A-computably compute a stage $g(n)$ such that $A \in V_{g(n)}$, and without loss of generality we can suppose that $V_{g(n+1)} \supseteq V_{g(n)}$. But as A has hyperimmune free degree, we can choose a computable function f so that $f(n) > g(n)$ for all n. Then if we define $W_n = V_{f(n)}$, being a Kurtz test such that $A \in \cap_n W_n$, a contradiction. $\square$

Actually, Yu Liang observed that the *same* proof shows the following.

Proposition 11.16.15 (Yu Liang)**.** *If A has hyperimmune-free degree then A is Kurtz random iff A is weakly 2-random.*

This improves an earlier direct proof (by Joe Miller [211]) that there is a weakly 2-random hyperimmune-free degree. It has a nice corollary.

Corollary 11.16.16. *There is a cone of Turing degrees all of which are weakly 2-random.*

Theorem 11.16.10 is only useful for $n = 1$. After that genericity and randomness diverge. The following result of Kurtz shows this.

Theorem 11.16.17 (Kurtz [165])**.** *The upward closure of the weakly* 2-*generic degree has measure 0.*

Lemma 11.16.18 (Kurtz [165])**.** *There is a fixed function $f \leqslant_{\mathrm{T}} \emptyset'$ such that for almost every $\mathbf{a}$ and almost every $g \leqslant_{\mathrm{T}} \mathbf{a}$, f dominates g.*

Proof. Kurtz' proof is very attractive and fairly typical (no pun intended) of measure-theoretical arguments in computability theory. We define a collection of functions $\{f_{n,k} : n, k \in \mathbb{N}\}$, computable from $\emptyset'$ such that either $f_{n,k}$ majorizes Φ_k^A or Φ_k^A is nontotal in a set of measure $1 - 2^{-(n+k+1)}$. We define $f_{n,k}(m)$, using $\emptyset'$ as an oracle, compute j such that $j \cdot 2^{-(n+k+m+2)} < \mu(\{A : \Phi_k^A(m) \downarrow\}) < (j+1) \cdot 2^{-(n+k+m+2)}$. To do this, for each j ask whether there is a finite set of strings $\{\sigma_i : i \in I\}$ with $\Phi_k^{\sigma_i}(m) \downarrow$ and $\sum_{i \in I} 2^{-|\sigma_i|} > j \cdot 2^{-(n+k+m+2)}$. By compactness, such a finite set must exist if $\mu(\{A : \Phi_k^A(m) \downarrow\}) > j \cdot 2^{-(n+k+m+2)}$.

Once we have j, we can effectively list a sequence $\{\sigma_i' : i \in I'\}$ with $\sum_{i \in I} 2^{-|\sigma_i'|} > j \cdot 2^{-(n+k+m+2)}$ and $\Phi_k^{\sigma_i'}(m) \downarrow$ for all σ_i'. We define

$$f_{n,k}(m) = \max\{\Phi_k^{\sigma_i'}(m) : i \in I'\} + 1.$$

Note that either $f_{n,k}$ is greater than $\Phi_k^A(m)$ or $\Phi_k^A(m)$ is undefined except on a set of measure $2^{-(n+k+2)}$.

Therefore, $f_{n,k}$ either dominates Φ^A or Φ_k^A is undefined save upon a set of measure at most $2^{-(n+k+1)}$, since $\mu(\{A : \Phi^A$ total and not dominated by $f_{n,k}\})$ $\leqslant \sum_{m \in \mathbb{N}} 2^{-(n+m+k+2)} = 2^{-(n+k+1)}$.

We define $f_n(m) = \max_{k \leqslant n}\{f_{n,k}\} + 1$. The claim is that f_n dominates all functions computable from A except on a set of measure at most 2^{-n}. This is because $\mu(\{A : \Phi^A$ total and not dominated by $f_n\})$ $\leqslant \sum_{k \in \mathbb{N}} 2^{-(n+k+1)} \leqslant 2^{-n}$. Finally, we finish the proof by setting, similarly, $f(m) = \max_{n \leqslant m} f_n(m)$. □

Proof. (of Theorem 11.16.17) Let f be as in Lemma 11.16.18. Let S_n be the set of strings such that for at least n x, we have $p_\sigma(x) > f(x)$. Then S_n is a Δ_2^0 set of strings, and evidently dense. Thus if G is weakly 2-generic, then it must meet S_n. Thus the principal function of G fails to be dominated by f. Let V be the set of all reals A such that no function computable from A majorizes f. Then is $A \in V$, no weakly 2-generic set can be computable from A. But V is a set of measure 1. □

Since the property of being weakly 2-generic is arithmetical, and in fact Δ_3^0 it follows that no 3-random real can be (even) weakly 2-generic. This follows by the effective $0 - 1$ Law, Theorem 11.9.2. Another limitation is seen as follows (example due to Kautz) Consider the dense Σ_1^0 classes $cup\{[\sigma] : |\sigma| \geqslant k \wedge \frac{\#\text{ of 0's in }\sigma}{|\sigma|} \geqslant \frac{3}{4}\}$ and $\cup\{[\sigma] : |\sigma| \geqslant k \wedge \frac{\#\text{ of 1's in }\sigma}{|\sigma|} \geqslant \frac{3}{4}\}$. If A is weakly 1-generic then it must meet these classes. Hence

$$\lim_{n \to \infty} \frac{\#\text{ of 0's in } A \upharpoonright n}{|\sigma|}$$

fails to exist for any weakly 1-generic real. By the law of large numbers this limit is always $\frac{1}{2}$ for any 1-random set. Thus *no weakly* 1*-generic set can even be* 1*-random.*

11.16.2 Below almost all degrees

We finish this section with the proof of Martin's Theorem 11.16.19 and Kurtz' Theorem that almost every degree bounds a 1-generic degree. The former can be derived from the latter via the fact that every 1− generic degree is hyperimmune. However, the main idea, "risking measure," used in Kurtz' Theorem is an elaboration of that used in Martin's, is much easier to understand in Martin's argument. The same kind of ideas had earlier been used by Paris [236].

Theorem 11.16.19 (Martin, unpubl.)**.** *Almost every degree is a hyperimmune degree.*

Actually, the *proof* of Theorem 11.17.3, in Section 11.17, will actually show the following.

Theorem 11.16.20. *If* **a** *is 2-random, then* **a** *is hyperimmune.*

Proof. By the zero-one law it is enough to prove that $\mu(\{A : A$ bounds a hyperimmune degree $\}) > 0$. To achieve this we construct a partial computable functional Ξ so that

$$\mu(\{A : \Xi^A \text{ is total and not dominated by any computable function}\} \geqslant \frac{1}{2}.$$

We will consider Ξ as a partial computable function from strings to strings. The reader should recall from Section 5.7, the main rule we need observe is that if $\Xi(\nu)\downarrow$ and $\widehat{\nu}$ extends ν, and $\Xi(\widehat{\nu})\downarrow$, then $\Xi(\nu) \subseteq \Xi(\widehat{\nu})$. Note that it is not necessary that if $\Xi(\nu)\downarrow$ then $\Xi(\gamma)\downarrow$ for all $\gamma \subseteq \nu$.

We will meet the requirements (on a set of positive measure):

$$R_e : \varphi_e \text{ total } \to \Xi^A \text{ is not majorized by } \varphi_e.$$

The most naive attempt to meet R_e is to define a witness n_e^A for R_e and then define

$$\Xi^A(n_e) = \varphi_e(n_e^A) + 1.$$

Of course the problem with this is that if, for instance, φ_e is empty, we would not be able to define $\Xi^A(n_e^A)$.

Martin's *fundamental idea* is to to try to implement this but fail for a set of A's whose total measure is less than $\frac{1}{2}$. That is we can appoint harmful witnesses n_e^A such that $\Xi^A(n_e^A)\uparrow$ for some e has total measure bounded by $\frac{1}{2}$.

This injury measure is divided amongst the R_e so that for a single R_e the injury will amount to at most $2^{-(e+2)}$. $\Xi(\gamma)\downarrow$ for all $\gamma \subseteq \nu$.

For a single e, R_e's strategy works as follows. Let $\nu_i^0, \ldots, \nu_i^{2i-1}$, list lexicographically all the strings of length i. Then R_e begins by working only on ν_{e+2}^0, appointing a witness n_e^0 for all strings extending ν_{e+2}^0. (Should we *satisfy* R_e on this cone, then a new witness will be chosen for the next cone we attempt to satisfy R_e on, as we see below.)

R_e will ask that $\Xi^A(n_e^0) \uparrow$ for all A extending ν_{e+2}^0 unless we see a stage s where $\varphi_e^s(n_e^0) \downarrow$.

Should such a stage s occur, we will be able for then define $\Xi^{\nu_{e+2}^0}(n_e^0) = \varphi_e(n_e^0) + 1$, and note that R_e will be met in the cone of reals extending ν_{e+2}^0.

Also should such a stage s occur, R_e's next action would be to move into the cone above ν_{e+2}^1. It would assert control of this cone and appoint a large number n_e^1 for these strings.

In the actual construction, we let the strategies work by priorities. Whilst R_e, say is working in some cone $[\nu_{e+2}^j]$ it will do so with priority e. Lower priority requirements must work outside this cone. If R_f had been working on some cone $[\nu_{f+2}^k] \subset [\nu_{e+2}^j]$, it would (perhaps temporarily) abandon this cone and move to the first available cone $[\nu_{f+2}^{\widehat{k}}] \subseteq \overline{[\nu_{e+2}^j]}$, (and similarly respect all $R_{f'}$ for $f' < f$.) However, should R_e achieve its goals on $[\nu_{e+2}^j]$, R_f would be free to move back there, at least as far as R_e is concerned.

Since each R_e can get stuck on at most one cone, and that cone has measure at most $2^{-(e+2)}$, we see that Ξ^A fails to be defined on a set of measure at most

$$\sum_e 2^{-(e+2)} = \frac{1}{2}.$$

The result follows by a standard application of the finite injury method.

Finally, following Kautz [140], we remark that 2-randomness is enough for the proof, since we simply look at the Π_2^0 class $\{A : \Xi^A \text{ is total, and not dominated by any computable function}\}$ which will be a Π_2^0 class of positive measure zero, and hence contain representatives of all 2-random degrees. $\square$

We remark that that this result can also be obtained by the following elegant argument of Nies, Stephan, and Terwijn.

Proof. (Nies, Stephan, and Terwijn [232]) Recall that in Lemma 9.7.10, it is shown that *A is 2-random iff for all computable time bounds g with $g(n) \geqslant n^2$,*

$$\exists d \exists^\infty n (C^g(A \upharpoonright n) \geqslant n - d).$$

Define

$$f(k) = \mu n[\exists p_1, \ldots, p_k \leqslant n](C^g(A \upharpoonright p_i) \geqslant p_i - d).$$

Then f is A computable. We claim that f is not dominated by any computable function. Suppose that $f(k) \leqslant h(k)$ for all k. Define the Π_1^0 class

$$\mathcal{P} = \{Z : \forall k[\exists p_1, \ldots, p_k \leqslant h(k)](C^g(Z \upharpoonright p_i) \geqslant p_i - d)\}.$$

Then $\mathcal{P} \neq \emptyset$, and every member of $\mathcal{P}$ is 2-random. But then there is a 2-random real that is, for instance, of low degree. $\square$

In Kuatz's Thesis, [140], it is claimed that weak 2-randomness is enough. We have already seen that this is not true in Corollary 11.16.16, since every 1-random hyperimmune-free degree is, in fact, weakly 2-random. as the following

Notice that because of this result we see a failure of the degree invariance on classes of positive measure for weak 2-randomness. (That is because we actually do succeed on a set of positive measure, and if a real is weakly 2-random it will meet the requirements on that set.)

Kurtz [165] remarks that there is no total computable operator Ψ such that Ψ^A is hyperimmune for almost every A. Now we turn to the more difficult theorem of Kurtz.

Theorem 11.16.21 (Kurtz [165])**.** *Almost every degree bounds a* $1-$*generic degree.*

We will prove Theorem 11.16.21 in spite of the fact that it can be derived as a corollary to Theorem 11.17.3 of the next section. We prove it as its proof will help to understand the more difficult proof of Theorem 11.17.3. Additionally, we will also elaborate on the method of the next theorem to prove Kurtz's Theorem, Theorem 11.18.2, that almost every degree has the property that the 1-generic degrees are downward dense below it.

Proof. We will give two proofs of this result. The first is much more like the proof of Martin's Theorem in the way that it deals with distributing measure. The second uses the method introduced by Kurtz [165], and is harder to understand as to exactly why it works. However, Kurtz methods allow us to introduce a number of technical devices which will be critical in our proof that all 2 randoms are computably in and above, Theorem 11.17.3, and also will be modified for the proof that the 1-generic degrees are downward dense below almost all degrees, Theorem 11.18.2.
(**Proof 1**) In view of the effective $0-1-$law it suffices to prove that

$$\mu(\{A : A \text{ bounds a 1-generic degree }\}) \geqslant \frac{1}{2}.$$

The requirements we must meet are the following (for a set of positive measure).

$$R_e : \Xi^A \text{has an initial segment in } V_e \text{ or } \exists \sigma \prec \Xi^A (\forall \tau \in V_e)(\sigma \nsubseteq \tau).$$

Here we are building the reduction Ξ and V_e denotes the e-th set of strings. The argument is a straightforward Π^0_2 priority argument.

In the construction we build a procedure Ξ in stages. Ξ must be defined on a set of positive measure.

As with Martin's Theorem, we will consider Ξ as a partial computable function from strings to strings. We will be willing to have Ξ^A not defined on a set of measure at most $2^{-(e+2)}$.

For R_0 we will divide the universe 2^ω into two portions. These are, for definiteness, $[00] \cup \overline{[00]}$. The reader should think of $[00]$ as taking the role of ν_0^0 in our proof of Martin's Theorem.

What we would *like* to do is to find some $\sigma \in V_0$ and the define Ξ so that σ would be in its range, thereby meeting R_e. *However,* we cannot know that such a desirable situation will occur. Thus, as with the proof of Martin's Theorem, we have set aside some measure (namely $2^{-|[00]|} = \frac{1}{4}$) where we choose *not* to define Ξ whilst we wait for some σ to occur in V_0.

While we are waiting for some string σ to occur in V_0^s for some s, we will pursue a strategy based on the belief that $V_0 = \emptyset$. This strategy will build the reduction Ξ, but *only* in the clopen set $\overline{[00]}$.

The benefit gained by us is that in our smaller universe $\overline{[00]}$, this strategy will believe that assume that $V_0 = \emptyset$, and hence only need to meet R_i for $i \neq 0$.

Thus, we will never define $\Xi_s(\nu)$ for ($\nu = 0$ and) any $\nu \in [00]$ unless some string γ occurs in V_0^s for some s. Should some γ occur in V_0^s, then we are free to define $\Xi_s(00) = \gamma$. Note that in that case we can regard R_0 as met in the cone $[00]$ because if $\widehat{\nu}$ extends 00, and $\Xi(\widehat{\nu}) \downarrow$, then, by the consistency conditions on Ξ, $\Xi(\widehat{\nu})$ extends $\Xi(00) = \gamma$.

Should such a stage s occur, we will thereafter attempt to meet $\{R_i : i \neq 0\}$ in the cone $[00]$. (More on this later)

If we see $\gamma \in V_0^s$, following the role model of Martin's Theorem, we'd like to switch R_0 to worrying about $\nu_0^1 = [01]$. In Martin's Theorem we pick a new large follower n_0^1 for all the strings of length s in this cone and will wait to define the *function* Ξ^A on this number for all A extending ν_0^1.

The principal problem not found in the proof of Martin's Theorem is that in the proof of our theorem is that *now* Ξ^A is a *set* and we have more to do than simply define it on a new value.

The point is that whilst we have been waiting for some γ to occur in V_0^s other requirements might well have already defined Ξ^ν for several ν with $01 \preccurlyeq \nu$. For example, we might have ν_1 and ν_2 with $\Xi^{\nu_1} = \delta_1$ and $\Xi^{\nu_2} = \delta_2$, and such that δ_1 and δ_2 are incomparable strings.

Now R_0 would be seeking some $\gamma_i \in V_0^t$ with $\delta_i \preccurlyeq \gamma_i$, for $i = 1, 2$. There is no particularly good reason that both must occur.

The solution to this problem is the following. First, for ease of notation, for all strings ν in $[01]$ of length s, say $\{\nu_1, \dots, \nu_{p(s)}\}$, we will define $\Xi_s(\nu)$ consistently with $\Xi_s(\widehat{\nu})$ which are already defined, so that $\Xi_s(\nu)$ will be a string of length $\leqslant s$ at that stage.

Each such string ν_i will now become associated with R_0 and will have with it a quanta of 2^{-s} that it is possible for it to reassign to the construction. R_0's goal in each such cylinder $[\nu_i]$ is to find some string $\gamma_i \in V_0$ with $\Xi^{\nu_i} \preccurlyeq \gamma_i$. Should such a string occur in V_0^t at some stage t, then R_0 is free to abandon the cone $[\nu_i]$, and we can define $\Xi^{\nu_i j} = \gamma_i$ for $j \in \{0, 1\}$.. We would say that R_0 becomes *verified* in the cone $[\nu_i$. It will then be free to redistribute its quote of 2^{-s} to the lexicographically least available strings of length t, in the same was as it did at stage s.

The point of all of this is that at any stage s of the construction, R_0 has under its control exactly 2^{-2} of measure where we are waiting for strings to occur in V_0.

R_1, and other lower priority requirements, work in the obvious way. Initially, R_1 will be assigned to [010] and its quota is $2^{-(1+2)} = \frac{1}{8}$. This version is, of course, guessing that R_0 is met because there is *no* $\gamma \in V_0$. Should R_0 be verified in [00], R_0 may well assert control of the cone above [01] and, perhaps therefore, the cone [010]. There is not problem here, as with Martin's Theorem.

R_1 will begin to work in [00] and will simply avoid any place that R_0 wishes to work. Naturally, R_1 can go back to a place R_0 once controlled should R_0 be confirmed in that place.

In the circumstances outlined above, R_0 has asserted control of [01]. Perhaps R_1 was not met there. R_0 did so via $\nu_1, \ldots, \nu_p$. Suppose that R_1 had not left [010]. If R_0 sees some $\gamma_i \in V_0^t$ extending Ξ^{ν_i} so that it can define $\Xi^{\nu_i j} = \gamma_i$, the strings $\nu_i j$ would become available for R_1 to control. if they were in a previously abandoned cone, then R_1 will re-assert control with priority 1. Notice that again the measure controlled at every stage by R_1 is again 2^{-3}, and more generally R_e controls $2^{-(e+2)}$. The remaining details fit together in a familiar way,

(**Proof 2**) Now we will look at the method introduced by Kurtz. We will sketch the proof of the result since full details will be provided in Theorem 11.17.3.

As above, for the sake of R_e, we will divide the universe 2^ω into two portions. These are, for definiteness, $[00] \cup \overline{[00]}$.

In this version of the construction, we will have four basic colours: *red, blue, green and yellow.* (And each of these will also have a subscript denoting what requirement the colour refers to, so that red$_4$ would be red for R_4. Naturally enough these are really e-states and the argument to follow is a kind of full approximation one.)

We will declare that the string 00 will have colour *red*, with, more generally in the full construction, it would have colour red$_0$. As above the goal of this string is to try to find some $\sigma \in V_0$ and define $\Xi^{00} = \sigma$, thereby meeting R_0 in the cone [00]. If we actually get to meet R_0 as above in this cone then the colour of 00 will change to green. At the beginning the empty string λ is given the colour blue$_0$, indicating (as we soon see) that this is where R_0's action is taking place.

While we are a waiting for this to happen, we will assign the colour yellow$_0$ to all the strings of length, say, 2, not coloured red. The idea is that 00 becomes green then we will try to deal with the yellow strings (via extensions). Now, whilst we are doing this, as above we will initially, R_1 will be assigned to [0100] and its quota is $2^{-(1+2)} = \frac{1}{16}$. (The string 010 would have colour red$_1$.)

Now the construction runs along more or less as above, until some stage where 00 becomes green because we see some $\sigma \in V_0^s$ and get to define $\Xi^{00} = \sigma$.

It is now that the action changes in Kurtz's proof. At this very stage, we will deal with all the yellow$_0$ strings (which are 01,11,10). They may well have $\Xi^\nu \downarrow [s]$ for various extensions of them, for instance, $010 \preccurlyeq \nu$, and R_1 might have acted.

As with the previous construction, we will consider all the length s extensions of such yellow$_0$ strings which again we can denote as $\{\nu_1, \dots, \nu_{p(s)}\}$

Here is the difference. We will give each of these ν_i colour blue$_0$ and begin a new construction at each of them which is a clone of the basic construction. That is we would

- Remove the colours yellow$_0$ from the length 2 strings
- make ν_i a blue$_0$ string,
- make ν_i00 a red$_0$ string, and
- make the length 2 extensions of ν_i yellow$_0$ strings.

Now the *goal* of R_0 via each red ν_i00 string is to find a strings $\sigma_i \in V_0^t$ with $\Xi^{\nu_i} \preccurlyeq \sigma_i$, and then make ν_i00 green$_0$ by defining $\Xi^{\nu_i00} = \sigma_i$.

Of course, within the cone [00], once R_0 is met then we can have R_1 take the role of R_0 and meet it precisely as we did for R_0.

More generally, R_e will have a number of blue$_e$ nodes ν, with a tree of length $e+2$ extensions coloured yellow$_e$, save one $\widehat{\nu}$ which is coloured red$_e$. Once the red$_e$ is realized, we will have defined $\Xi^{\widehat{\nu}} = \gamma$ where $\gamma \in V_e$ and γ extends Ξ^ν.

The key point is that each time we attempt to meet R_0 above some string we do so with probability at least $\frac{1}{2}$ Hence, in this construction,

$$\mu(\{A : \text{ infinitely many attempts to satisfy } R_0 \text{ for } A\}) = 0.$$

That is because this probability is $\Pi_{i\in\mathbb{N}}(1 - \frac{1}{2}) = 0$.

We will give more details of this analysis and, indeed, the construction when we give the details for Theorem 11.17.3. $\square$

With a more precise analysis of the classes that arise in the proof above, we can actually show that $\Xi(A)$ is total for all 2-random sets A, because the classes we need to avoid are Π_2^0 classes of zero measure. Moreover, since

$\{A : \Xi(A) \text{ total }\}$ is a Π^0_2 class of positive measure, and it contains a 2-random real witha 1-generic predecssor, the result for 2-randoms will follow from the effective 0-1 Law:

Corollary 11.16.22. *If A is 2-random then it has a 1-generic predecessor.*

Again Kautz [140] claimed that the result holds for weakly 2-randoms. It is certainly true that holds for a class of 2-randoms of positive measure.
Kurtz [165] points out the following corollary to the proof above.

Corollary 11.16.23 (Kurtz [165])**.** *The initial segment of degrees below almost all degrees fails to be linearly ordered. Indeed this holds below a 2-random degree.*

Proof. By Jockusch [133], every 1-generic degree is computably enumerable in some strictly lower one. Every nonzero computably enumerable degree bound a minimal degree by Theorem 5.15.5. In relativized form this gives the result. □

It is tempting to conjecture that almost every degree is computable in a 1-generic one. However, this result fails to hold.

Theorem 11.16.24 (Kurtz [165])**.** *Almost no degree is computable in a 1-generic degree.*

Proof. If we suppose otherwise, then there must be a procedure Φ and a number n_0 such that

$$\mu(\{A : A = \Phi^G \text{ for some 1-generic } G\}) > 2^{-n_0}.$$

The proof is to sonstruct a Σ^0_1 class S of strings such that $\mu(\{A : A = \Phi^B$ for some B meeting $S\}) \leqslant 2^{-n_0}$ and such that if G is 1-generic, then either G meets S or Φ^G is not total.
Construction Let σ_i denote the i-th string. Let τ_i denote the least τ extending σ_i such that Φ^τ is defined on at least $i + n_0 + 1$ values. τ_i is undefined should no such τ exist. Then the function $f : \sigma_i \mapsto \tau_i$ is partial computable, and we define S to be the range of f.
End of Construstion

Lemma 11.16.25. $\mu(\Phi^S) \leqslant 2^{-n_0}$.

Proof. This follows since $\mu(\Phi^{\tau_i}) \leqslant 2^{-(n_0+i+1)}$ for each i and by countable addtivity. □

Lemma 11.16.26. *If G is 1-generic, then either Φ^G is non-total, or G meets S.*

Proof. Suppose otherwise for G. Then by 1-genericity, there is a string $\sigma_i \prec G$ such that, for all $\tau \in S$, $\sigma_i \not\preceq \tau$. Therefore, Φ^G can be defined on at most $i + n_0$ many arguments, and is not total. □

It follows that

$$\mathcal{G} = \{A : A = \Phi^G \text{ for some 1-generic } G\}$$

is a subset of $\{A : A = \Phi^B \text{ for some } B \in S\}$. By monotonicity of measure it follows that

$$\mu(\mathcal{G}) \leqslant \mu\{A : A = \Phi^B \text{ for some } B \in S\} \leqslant 2^{-n_0},$$

being a contradiction. □

Actually, there is a complete *local* characterization of sets computable in 1-generic ones. We say that a set of strings S is a (Σ_1)-*cover* set for a set A iff density holds, that is, for all $\sigma \prec A$, there is a $\tau \in S$, such that $\sigma \preccurlyeq \tau$. We say the cover is *proper* if no member of S is an initial segment of A.

Definition 11.16.27 (Chong and Downey [51, 50])**.** We say that a proper cover S of A is a (Σ_1)-*tight cover*[5] if for all covers $\widehat{S}$ of A, there is a string $\sigma \in S$, and a string $\tau \in \widehat{S}$ such that $\sigma \preccurlyeq \tau$.

The reader might think of a tight cover as a "simple set" for covers. The following definative result classifies sets and degrees computable in 1-generic ones.

Theorem 11.16.28 (Chong and Downey [50, 51])**.** *The following are equivalent.*

(i) A has no tight cover.

(ii) There is 1-generic G with $A \leqslant_{\mathrm{T}} G$.

Furthermore, the proof in [51] shows that there is a single procedure Φ such that if $A \leqslant_{\mathrm{T}} G$ and G is 1-generic, then there is a 1-generic $\widehat{G} \leqslant A''$ such that $A = \Phi^{\widehat{G}}$. Notice that it is a consequence of Kurtz Theorem that almost every set has a tight cover, since almost no degree is computable in a 1-generic degree.

11.17 Every 2-random is CEA

In this section we prove the remarkable result of Kurtz [165] that almost every real A is computably enumerable relative to some set $X <_{\mathrm{T}} A$, that is almost every real is *computably enumerable in and above* (CEA). The proof resembles the proof (Proof 2) that almost every degree bounds a 1-generic.

The history here is that Jockusch [133] had proven that if we let $\Xi(A) = \{2^i 3^j : i \in A \wedge 2^i 3^j \notin A\}$, then the following holds.

[5] In the Chong-Downey papers, this was called a Σ_1^0 dense set for A, which would perhaps be confusing in our context.

Theorem 11.17.1 (Jockusch [133])**.** *If X is 1-generic then X is computably enumerable in $\Xi(X)$ and $\Xi(X) <_{\mathrm{T}} X$. Hence every 1-generic is CEA*[6]*.*

Proof. Clearly, $A \leqslant_{\mathrm{T}} \Xi(X)$. Since A is immune, for each i there is a j such that $2^i 3^j \notin A$. Thus, $i \in A$ iff $(\exists j)[2^i 3^j \in \Xi(A)]$. Hence, A is computably enumerable in $\Xi(A)$.

1-genericity is used in the proof that $A \not\leqslant_{\mathrm{T}} Xi(A)$. Let $\Xi(\sigma)$ be the unique string ν of length $|\sigma|$ such that

$$\nu^{-1}(1) = \{2^i 3^j : \sigma(i) = 1 \wedge \sigma(2^i 3^j) = 0\}.$$

Lemma 11.17.2. *Let $\nu \preccurlyeq \sigma$ be arbitary and $n > |\nu|$ not of the form $2^i 3^j$. Then there is a string τ with $\nu \preccurlyeq \tau$, $\tau(n) = 1$, and $\Xi(\sigma) \preccurlyeq \Xi(\tau)$.*

Proof. If $\sigma(n)$ is 1 or undefined, then τ can be any extension of σ with $\tau(n) = 1$. Suppose that $\sigma(n) = 0$. Let S be the (under inclusion) smallest set such that

(i) $n \in S$, and

(ii) $2^i 3^j \in S$ whenever $i \in S$ and $\sigma(i) = 0$ and $2^i 3^j < |\sigma|$.

Let τ be the string of length $|\sigma|$ with $\tau^{-1}(1) = \sigma^{-1}(1) \cup S$. Then it follows that $\nu \preccurlyeq \tau$. To finish we need to prove that $\Xi(\sigma) \preccurlyeq \Xi(\tau)$. Suppose that $k < |\Xi(\sigma)| = |\sigma|$ is given. If k is not of the form $2^i 3^j$, the clearly $\Xi(\sigma)(k) = \Xi(\tau)(k) = 0$. Thus we may assume that $k = 2^i 3^j$.

If $\Xi(\sigma)(k) = 0$, then either $\sigma(i) = 0$ or $\sigma(2^i 3^j) = 1$. Clearly, if $\sigma(2^i 3^j) = 1$ then $\tau(2^i 3^j) = 1$, and hence $\Xi(\tau)(2^i 3^j) = 0$. Now suppose that $\sigma(i) = 0$. If $i \in S$, then $2^i 3^j \in S$ by the closure properties of S, $\tau(2^i 3^j) = 1$, and hence $\Xi(\tau)(2^i 3^j) = 0$. Finally, if $i \notin S$, then $\tau(i) = 0$ and hence $\Xi(\tau)(2^i 3^j) = 0$.

If $\Xi(\sigma)(k) = 1$, then $\sigma(i) = 1$ and $\sigma(2^i 3^j) = 0$. Since $\sigma(i) = 1$, $\tau(i) = 1$. Also it follows by induction on the construction of S that it contains no numbers of the form $2^d p^q$ with $\sigma(d) = 1$. Since $\sigma(2^i 3^j) = 0$, and $2^i 3^j \notin S$, we see that $\tau(2^i 3^j) = 0$, and hence $\Xi(\tau)(k) = 1$. □

To finish the proof of the Jockusch's Theorem, Suppose that A is 1-generic and Ψ is a Turing functional with $\Psi(\Xi(A)) = A$. Let V be the set of strings σ such that σ is compatible with $\Psi(\Xi(\sigma))$. Clearly A extends no string in V, and T is a computable set of strings. Thus by 1-genericity, there is a $\nu \prec A$ such that no extension of ν is in T. Choosing $n \geqslant |\nu|$ with n not of the form $2^i 3^j$, we can choose σ with $\nu \preccurlyeq \sigma$ and with $\Psi(\Xi(\sigma))(n) = 0$. By Lemma 11.17.2, we can choose τ with $\nu \preccurlyeq \tau$ such that $\tau(n) = 1$ and $\Xi(\sigma) \preccurlyeq \Xi(\tau)$. Then $\Psi(\Xi(\tau))(n) = 0$ and $\tau(n) = 1$ so that $\tau \in V$, contradicting the fact that $\nu \preccurlyeq \tau$. □

[6] Jockusch [133] attributes the idea behind this proof to Martin.

However, Kurtz [165] proved that for almost every real B, $B \leqslant_{\mathrm{T}} \Xi(B)$, and hence a different approach will be needed for dealing with random reals.

Theorem 11.17.3 (Kurtz [165])**.** *Suppose that A is 2-random. Then A is CEA.*

Proof. As in Theorem 11.16.21, and Martin's Theorem, we construct an operator Ξ so that

$$\mu(\{A : \Xi(A) \text{ total and } \Xi(A) \nleqslant_{\mathrm{T}} A\}) \geqslant \frac{1}{4},$$

and A is c.e. in $\Xi(A)$ whenever $\Xi(A)$ is total. The calculation that 2-randomness is enough comes from again analyzing the method of satisfaction of the requirements. The main ideas of the proof below are due to Kurtz [165], but we will give the details as carefully presented by Kautz [140]. Neither of these accounts have ever been published other than in Kutrz's and Kautz's respective PhD Theses.

To make A c.e. in $\Xi(A)$ we will ask that $n \in A$ iff $\langle n, m\rangle \in \Xi(A)$ for some m. Thus, in fact A is enumeration reducible to $\Xi(A)$. Now whilst doing this we must meet requirements of the form

$$R_e : A \neq \Phi_e^{\Xi(A)},$$

there Γ_e denotes the e-Turing procedure.

Definition 11.17.4. We say that a string ξ is *acceptable* for a string σ iff

$$\xi(\langle m, n\rangle) = 1 \to \sigma(n) = 1.$$

Remember we are trying to make A c.e. in Ξ^A. Thus we will always require that $\Xi^\sigma[s]$ is acceptable. We will also try, whenever possible, to use ξ to represent string in the range of $\Xi[s]$.

Now we recall the main players from Proof 2 of Theorem 11.16.21. There were the blue strings representing the base of one of the cones above which we are trying to meet R_e and a tree of strings with yellow leaves $\ni_i$, and one red leaf, which is "testing" and we are trying to turn green.

In this construction, initially the method looks similar, but, as Kurtz [165], page 99 says, "the roles of these familiar devices are subtly changed."

The red strings will be changed most. First as we see they will always be of the form $\beta 1^{e+2}$ instead of $\beta\frown 0^{e+2}$ as they were in the proof of Theorem 11.16.21, Proof 2. The reason for this change will become apparent later, but this is certainly not the only change. In the proof of Theorem 11.16.21, the red(=red$_e$) strings had the role of testing the hypothesis "does the blue$_e$ string β they extend have the property that Ξ^β has an extension in V_e?" If this *failed* to happen then in the cone above ν we have actually won R_e, by the pressure exerted on V_e caused by the red string and passively above the yellow ones.

In this construction, as we now see, there are new players, the *purple*$_e$ strings. These will represent cones R_e will forbid us to build within the construction. The role of the red$_e$ strings, is as a (in the words of Kurtz) "safety valves" for the pressure exerted by the purple$_e$ strings. In some sense this happens because we can see where it would be bad to build because of R_e, but we are not allowed to forbid too much measure.

In more detail, we attempt to meet R_e in some cone above a blue$_e$ string β as follows. We form our tree of strings above β as before, and give the colour red$_e$ to $\beta 1^{e+2}$, and the others the colour yellow$_e$.

The new players, the purple$_e$ strings will represent areas where we are failing to satisfy R_e in the cone $[\beta]$ as we see below. We must make sure that the measure they represent is not too big. If it becomes too big the "safety valve" pops and we will be able to argue that we can meet things because of a back up strategy.

The following definition is the key.

Definition 11.17.5. We say that a string θ is *threatening* a requirement R_e is there is a yellow$_e$ strings ν extending its blue$_e$ predecessor β with $\nu \preccurlyeq \theta$, and a strings ξ acceptable to θ such that

(i) $|\theta| \leqslant s$

(ii) $|\xi| = s$

(iii) $\Xi^{\nu}[s] \preccurlyeq \xi$

(iv) $\Phi_e^{\xi}(k) = \nu(k) = 0[s]$ for some k such that $|\beta| < k \leqslant |\nu|$, and

(v) no initial segment of θ has colour purple$_e$.

That is, to say that θ is threatening means that it is projecting via Ξ to something ξ that *potentially* could be an initial segment of a real β which $\Phi_e^{\beta} = \alpha$ and perhaps $\Xi^{\alpha} = \beta$. We hope that we avoid such reals. We will monitor the situation and pursue a different strategy should there appear too many such potential reals.

The reader should note that item (iv) above means that if ρ denotes the unique red$_e$ extension of β at stage s, for some k with $\rho(k) = 1$ we must have $\Phi_e^{\xi}(k) = \nu(k) = 0 \neq \rho(k) = 1$, by fact that $\rho = \beta 1^{e+2}$. That means if we chose to define Ξ^{α} on extensions α of ρ to emulate Ξ^{ν}, then it *cannot be* that $\Phi_e^{\Xi^{\alpha}} = \alpha$ as it must be wrong on ρ.

In the construction we will give any string θ which is threatening R_e colour purple$_e$.

Suppose that A is a set that extends a yellow$_e$ string ν (as its final colour) and that ν has no purple$_e$ extension. The we will claim that A satisfies R_e. Suppose not. Then $A = \Phi_e^A$. Then, by choice of ρ, we must have $\Phi^A = \nu(k) = 0$ for some k with $|\beta| < k \leqslant |\nu|$. But that is a contradiction, since then we would have coloured some initial segment of A purple$_e$, as $A \upharpoonright \varphi_e(k)$ is an an acceptable string threatening R_e extending β.

Now all this is fine, provided that we don't make purple$_e$ too many strings extending ν and hence kill too much measure. Kurtz's key idea is the following.

For each purple$_e$ string θ, let θ' denote the unique string of length $|\theta|$ extending the red$_e$ string ρ with $\theta(m) = \theta'(m)$ for all $n \geqslant |\rho|$. The idea is that we will be able to give these θ' the colour green$_e$, and bound their density away from 0, if the density of purple$_e$ strings grows too much.

Specifically, when the density of purple$_e$ strings above the blue$_e$ string β exceeds $2^{-(e+3)}$, there must be some yellow$_e$ string ν such that the density of purple$_e$ strings above ν must also exceed $2^{-(e+3)}$, by Lebesgue density. Let $\{\theta_1, \ldots, \theta_n\}$ list the purple$_e$ strings above ν, where for definiteness ν is chosen lexicographically least. For each θ_i, let ξ_i be the least string which witnesses the threat to R_e according to Definition 11.17.5. Thus,

(i) ξ_i is acceptable for Θ_i,

(ii) $\Xi^\nu \preccurlyeq \xi_i$, and

(iii) $\Phi_e^{\xi_i}(k) = 0 = \theta_i(k)$ for some k with $\rho(k) = 1$.

The the construction will have been arranged so that $\Xi^\nu[s] = \Xi^\rho[s]$. The action is that

(a) Any string extending β loses its colour.

(b) β loses colour blue$_e$.

(c) Each string θ'_i is given colour green$_e$.

(d) We define $\Xi^{\theta'_i} = \xi_i$, which will then force $\Phi_e^{xi_i}(k) \neq \theta'(k)$ for some k with $|\beta| < k \leqslant |\nu|$.

Notice that (d) above justifies the use of the colour green$_e$ for the strings θ'_i. The reader should note that each time we are forced to pop the safety valve we are guaranteed to succeed on set of measure at least $2^{-(e+5)}$ above β, and hence we will succeed on a set of positive measure. Notice also that since we remove all colours from the purple strings then we will be able to replicate this construction on strings extending those that have lost their colour. Finally, we remark that the fact that we only consider acceptable strings will mean that some subset of A is computably enumerable in Ξ^A. This will be fixed via a "catch up" when we assign blue$_e$ colours. The formal details will now follow. We follow the account of Kautz [140], A.2.

Construction

Stage 0 Assign λ colour blue$_0$, and the strings 00,01,10 the colour yellow$_0$, and the string 11 the colour red$_0$.

Stage $s+1$ This consists of four substages. At each stage we will have defines a set of strings D_s where we will have defined $\Xi[s]$. The leaves of D_s will be called *active* strings.

Substage 1. For each $e \leqslant s$ any string in D_s threatening R_e will be coloured purple$_e$.
Substage 2 For each $e \leqslant s$ do the following for each blue$_e$ string β.

If the density of purple$_e$ strings extending β is at least $2^{-(e+3)}$, we say that R_e *acts*. Let ρ be the red$_e$ string extending β, and ν a yellow$_e$ string extending β with high density of purple$_e$ strings extending it. Let $\{\theta_1, \ldots, \theta_n\}$ list the purple$_e$ strings extending ν, $\{\xi_1, \ldots, \xi_n\}$ the corresponding witnesses for the $theta_i$'s threat to R_e, and $\{\theta_1, \ldots, \theta_n\}$ as above.

Then every string above β loses its colour, except those with colour purple$_j$ for some $j < e$ (priority). There are then two cases.
Case 1 ρ has a string with colour purple$_j$ for some $j < e$.
Action Do nothing.
Case 2 Otherwise. Then for each i with $1 \leqslant i \leqslant n$, define

$$Xi^{\theta_i'} = \xi_i[s+1].$$

Give each θ_i' colour green$_e$.
Substage 3 For each active string σ such that σ is not red and σ has no purple predecessor, choose the least e such that σ has no predecessor coloured green$_e$ or yellow$_e$. Then define $\Xi^{\sigma 0} = \Xi^{\sigma 1} = \xi[s+1]$ where ξ is the lexicographically least string which is acceptable for σ and such that

$$\sigma(n) = 1 \text{ iff } \xi(\langle n, m\rangle) = 1, \text{ for some } m,$$

and give $\sigma 0$ and $\sigma 1$ colour blue$_e$.
Substage 4 For each e and each active blue$_e$ string β, we define $\Xi^{\beta\tau} = \Xi^{\beta}$ for all τ of length $e+2$. Give $\beta 1^{e+2}$ the colour red$_e$. Give the other length $e+2$ extensions colour yellow$_e$.
End of Construction
Verification We will need the following colour classes.

$$B_e = \{A : \exists \sigma \prec A(\sigma \text{ has final colour yellow}_e \text{ or green}_e)\}.$$

$$S_e = \{A : \exists \sigma \prec A(\sigma \text{ has final colour red}_e)\}.$$

$$P_e = \{A : \exists \sigma \prec A(\sigma \text{ has final colour purple}_e)\}.$$

Define $S = \cup_e S_e$, and $P = \cup_e P_e$.

Lemma 11.17.6. *If $A \in \cap_e B_e$, then Ξ^A is total, and $A \not\leqslant_{\mathrm{T}} Xi^A$.*

Proof. By Substages 3 and 4 of the construction we define Ξ^A for arbitrarily long initial segments of A should A be in every B_e class, and by Substage 3, A is computably enumerable in Ξ^A. Now we argue as in the intuitive discussion. Suppose that for some least e, $A = \Phi_e^{\Xi^A}$. Let $\sigma \prec A$ have final colour yellow$_e$ or green$_e$.

Case 1 σ has final colour green$_e$. Then, by construction, at some stage s_0, we defined $\Xi^\sigma = \xi$ for some ξ with $\Phi_e^\xi(k) = 0$ yet with $\sigma(k) = 1$ for some k. Contradiction.
Case 2 σ has colour yellow$_e$. The let ρ be the associated red$_e$ string. It is the case that for some k, we must have $\rho(k) = 1 \neq 0 = \sigma(k)$. By our assumption, $\sigma(k) = \Phi_e^{\Xi^A}(k)$. Let ξ be the shortest initial segment of Ξ^A such that $\Phi_e^\xi(k) = 0[s_0]$, at some least s_0. Then, ξ is acceptable for σ, and thus by definition, σ threatens R_e via ξ at all stages $s \geqslant s_0$. Thus σ would have been coloured purple$_e$ by the construction, unless some initial segment τ of σ was already coloured purple$_j$ for some $j < e$. Note that if τ were to lose the colour purple$_j$ at any stage after s_0, then every string extending the blue$_j$ string $\beta(\tau) \prec \tau \prec \sigma$. Thus, inductively, σ would have lost the colour yellow$_e$. But then since we know that $A \in \cap_{j \leqslant e} B_j$, it must be that σ will receive the colour purple$_e$. But that is a contradiction, as then the construction stops above σ, a contradiction.

Therefore, we can conclude $A \not\leqslant_T \Xi^A$. □

Lemma 11.17.7. $\mu(B_{e-1} - (B_e \cup S_e \cup P_e)) = 0$.

Proof. We assume that $e > 0$, and for $e = 0$ we use the proof below but with $B_{-1} = 2^\omega$ and $\sigma = \lambda$ to begin.

The class B_{e-1} is a disjoint union of clopen sets $[\sigma]$ where σ is either coloured yellow$_{e-1}$ or green$_{e-1}$. Suppose that $B_{e-1} - (B_e \cup S_e \cup P_e)$ has nonzero measure. Then we would note that $\overline{(B_e \cup S_e \cup P_e)}$ would have density bigger than 0 in some interval $[\sigma]$ where σ has final colour green$_{e-1}$ or yellow$_{e-1}$. The by the Lebesgue Density Theorem, Theorem 4.2.3, there must be some string $\widehat{\sigma}$ with $\sigma \prec \widehat{\sigma}$, such that $\overline{(B_e \cup S_e \cup P_e)}$ has density greater than $1 - 2^{-(2e+5)}$ in $[\widehat{\sigma}]$.

Thus, to establish Lemma 11.17.7, it will suffice to show that for any string $\widehat{\sigma}$ extending σ, the density of $(B_e \cup S_e \cup P_e)$ in $[\widehat{\sigma}]$ is at least $2^{-(2e+5)}$.

If σ has final colour green$_{e-1}$ or yellow$_{e-1}$ then let s_0 be the stage where σ received its final colour.

The action of any requirement R_i will remove the colours from all extensions of a blue$_i$ string β except a purple$_m$ string for $m < i$. The first thing that this means is that for $j < e$, if any string τ comparable to σ has final colour purple$_j$ that is then its final colour. That is because a colour purple$_j$ can only be removed because of the action of a higher priority R_i, and a blue$_i$ predecessor of τ would assigned be a blue$_i$ predecessor of σ. R_i's action would remove the colour from σ. The second thing that this means is that if any string τ extending σ has colour c_e after s_0, then it will only lose that colour through the action of R_e. Suppose it is because of some R_i with $i \neq e$. If $i < e$, then the action of R_i removing c_e from τ would also remove σ's colour. If $i > e$, then either c_e =purple$_e$, in which case the action of R_i does not remove the colour of τ, or else no blue$_i$ string is a predecessor of τ and hence τ is unaffected by the action of R_i, after all.

Now suppose that some initial segment $\tau \prec \sigma$ has colour purple$_j$ for some $j < e$, by necessity at some stage $t \geqslant s_0$, Then τ has final colour purple$_j$. Therefore P has density 1 in $[\widehat{\sigma}]$, giving the result. Thus we can suppose that no initial segment of σ is ever purple after stage s_0. Then we know that some initial segment of $\widehat{\sigma}$ receives colour blue$_e$ after stage s_0. though this is perhaps temporary. Choose i with $\sigma i \preccurlyeq \widehat{\sigma}$. Let $\beta = \sigma i$. Then β receives colour blue$_e$ in Substage 3 of the construction by the end of stage $s_0 + 1$.

Now define β_0 be the longest initial segment of $\widehat{\sigma}$ to ever receive the colour blue$_e$ in the construction.

If β_0 has final colour blue$_e$, then every set A extending β_0 passes through a node with final colour yellow$_e$ or red$_e$. Hence $B_e \cup S_e$ has density 1 in $[hat\sigma]$, giving the result.

Now suppose that β_0 loses the colour blue$_e$ at some stage. As we have seen, this can only happen if the density of purple$_e$ strings above β_0 exceeds $2^{-(e+3)}$ because R_e acts. During the stage when β_0 loses its colour, we form a disjoint cover $T = \{\tau_0, \ldots, \tau_m\}$ of $[\beta_0]$ by strings τ such that either

(i) τ is green$_e$,

(ii) τ has a purple$_j$ initial segment for some $j < e$, or

(iii) τ is blue$_e$.

Note that if τ has colour green$_e$ that is its final colour, and hence if $\tau \prec \widehat{\sigma}$ B_e has density 1 in $[\widehat{\sigma}]$. If τ has a colour purple$_j$ predecessor, then P has density 1 in $[\widehat{\sigma}]$.

Now $\widehat{\sigma}$ cannot extend any τ with colour blue$_e$ by choice of β_0. Thus, the final option is that $\widehat{\sigma}$ does not extend any string in the cover T. The conclusion is that $[\widehat{\sigma}]$ is it self covered by strings in $\tau \in T$. We will prove that for each such τ, the density of $(B_e \cup S_e \cup P)$ in $[\tau]$ is at least $2^{-(2e+5)}$. This is evident should τ be coloured either green$_e$ or has an initial segment which is purple$_j$ for some $j < e$.

The final possibility is that τ is coloured blue$_e$. If τ has final colour blue$_e$, then every set extending τ passes through a node with final colour red$_e$ or yellow$_e$, and hence $B_e cup S_e$ has density 1 in $[\tau]$. On the other hand, if τ loses its colour blue$_e$ at some stage it can only be that the density of purple$_e$ strings above $[\tau]$ exceeds $2^{-(e+3)}$. There are two possibilities.

Case 1 There is some purple$_j$ string $\theta \prec \rho$, where $\rho = \tau 1^{e+2}$ is the red$_e$ string extending τ, and $j < e$. Then θ has final colour purple$_j$ and the density of P in $[\tau]$ is at least $2^{-(e+2)}$.

Case 2 Each θ'_i in the collection of strings $\{\theta'_1, \ldots, \theta'_n\}$ extending ρ gets colour green$_e$, which will be its final colour. By the way these are selected, the density of green$_e$ strings in $[\rho]$ is at least $2^{-(e+3)}$ and hence in $[\tau]$ is at least $2^{-(2e+5)}$. $\square$

Lemma 11.17.8. $B_e \subseteq B_{e-1}$.

Proof. Let $A \in B_e$, and $\sigma \prec A$ have final colour green$_e$ or yellow$_e$, with $\beta \prec \sigma$ the associated blue$_e$ string. If $e = 0$ there is nothing to prove. Assume that $e > 0$. By construction, when β received its blue$_e$ colour, there must have been some string $\tau \prec \beta$ which had colour either green$_{e-1}$ or yellow$_{e-1}$. Because of the way that the construction works, any action of some requirement removing this colour from τ would necessarily remove the colour from σ as well, and hence τ's colour is final. □

Lemma 11.17.9. $\mu(S \cup P) + \mu(\cap_e B_e) = 1$.

Proof. By Lemma 11.17.7, since $\overline{S} \subset \overline{S_e}$,

$$\mu((B_{e-1} - B_e) \cap (\overline{S \cup P})) = 0.$$

By Lemma 11.17.8, we also see that

$$2^\omega = \cup_e (B_{e-1} - B_e) \cup (\cap_e B_e).$$

However, $(S \cup P) \cap (\cap_e B_e) = \emptyset$. Therefore,

$$\begin{aligned} S \cup P &= (S \cup P) \cap (\cup_e (B_{e-1} - B_e) \cup (\cap_e B_e)) \\ &= (S \cup P) \cap \cup_e (B_{e-1} - B_e). \end{aligned}$$

Thus,

$$\begin{aligned} \mu(\cup_e (B_{e-1} - B_e) &= \mu(\cup_e (B_{e-1} - B_e) \cap (S \cup P)) + \mu(\cup_e (B_{e-1} - B_e) \cap \overline{S \cup P}) \\ &= \mu(\cup_e (B_{e-1} - B_e) \cap (S \cup P)) + 0 = \mu(S \cup P), \end{aligned}$$

since $\mu((B_{e-1} - B_e) \cap (\overline{S \cup P})) = 0$. Therefore,

$$1 = \mu(\cup_e (B_{e-1} - B_e)) + \mu(\cap_e B_e) = \mu(S \cup P) + \mu(\cap_e B_e),$$

as required. □

Lemma 11.17.10. *(i)* $\mu(S) \leqslant \frac{1}{2}$.

(ii) $\mu(P) \leqslant \frac{1}{4}$.

(iii) $\mu(\cap_e B_e) \geqslant \frac{1}{4}$.

Proof. (i) Since $S = \cup_e S_e$, it is enough to show that for each e, $\mu(S_e) \leqslant 2^{-(e+2)}$. By construction, the strings β with final colour blue$_e$ are all disjoint, and the density of red$_e$ strings in any cone $[\beta]$ is bounded by $2^{-(e+2)}$. Therefore the total measure of strings with final colour red$_e$ is bounded by the estimate

$$\sum_{\{\beta : \beta \text{ final colour blue}_e\}} 2^{-(e+2)} 2^{-|\beta|} \leqslant 2^{-(e+2)}.$$

(ii) Use the same method as we did for (i) and the fact that the density of purple$_e$ strings in any blue$_e$ cone $[\beta]$ is bounded by $2^{-(e+3)}$.

(iii) This follows by (i) and (ii), together with an application of Lemma 11.17.9. $\square$

Now, if $A \in S \cup P$, then Ξ^A is not total, and if $A \in \cap_e B_e$, then $\Xi^A <_{\mathrm{T}} A$. Again we note that $\{A : \Xi^A \text{ is total }\}$ is a Π^0_2 class containing members of each 2-random degree. Whenever Ξ^A is total, A is CEA Ξ^A and $\Xi^A <_{\mathrm{T}} A$. This completes the proof. $\square$

Earlier we mentioned that Theorem 11.16.21 can be deduced from Theorem 11.17.3. The way that we deduce Theorem 11.16.21 uses Theorem 11.17.11 below. The following result was known to many authors in the 1970's. The proof below is due to Richard Shore.

Theorem 11.17.11. *Suppose that A is CEA(B) with $B <_{\mathrm{T}} A$. Then A bounds a 1-generic degree.*

Proof. Let $A = \cup_s A_s$ be a B-computable enumeration of A. Let $g(n)$ denote the computation function of A:

$$c(n) = \mu s(A_s \upharpoonright n = A \upharpoonright n).$$

We construct the 1-generic set $G = \lim_s G_s$ via a finite extension argument. Let V_e denote the e-th c.e. set of strings. At stage s we let $G_{s+1} = G_s$ unless (for some least e) we see some extension $\gamma \in V_{e,c(s+1)}$ with $G_s \prec \gamma$. In this latter case, let $G_{s+1} = \gamma$.

Suppose that G is not 1-generic. We claim that $A \leqslant_{\mathrm{T}} B$. Let e such that G does not meet V_e and every initial segment of G is extended by one in V_e. Let s_0 be a stage after which e has priority. It suffices to compute $c(n)$ from B, since B enumerates A. Using simultaneous induction on s and G_s, for $s > s_0$.

Now assume that G_s is known. Compute a minimal stage t such that some extension γ_s of G_s occurs in $V_{e,t}$. The it must be that $t > c(s+1)$, lest we would act for e. This allows us to compute $c(s+1)$, and hence G_{s+1}. $\square$

Note that therefore it follows by Miller's Theorem [211] that there are weakly 2-random reals that are *not* CEA.

11.18 Where 1-generic degrees are downward dense

Definition 11.18.1. We say that a class $\mathcal{C}$ of degrees is *downward dense* below a degree $\mathbf{a}$ iff for all nonzero $\mathbf{b} < \mathbf{a}$ there is a degree $\mathbf{c} \leqslant \mathbf{b}$ with $\mathbf{c} \in \mathcal{C}$.

The following result appears in Kurtz's Thesis and is otherwise unpublished. It seems very surprising that it is true. Kurtz stated the result as *below almost every degree the 1-generic degrees are downward dense.* The calculation that 2-randomness is enough is again simply a calculation on the nullsets used in Kurtz's proof, and is more or less implicit in his work.

Theorem 11.18.2 (Kurtz [165])**.** *The 1-generic degrees are downward dense below any 2-random degree.*

Proof. The method of proof is a subtle modification of the proof of Theorem 11.16.21. Assume that the result is false. Then by the Lebesgue Density Theorem, Theorem 4.2.3, There is a a partial computable operator Ψ_e such that

$$\mu(\{A : \Psi_e^A \text{ total, noncomputable and does not bound a 1-generic set}\}) > \frac{7}{8}.$$

Following Kurtz [165], let $F(\sigma)$ denote the largest initial segment of Ψ_e^σ obtainable in $|\sigma|$ many steps, $\Psi_{e,|\sigma|}^\sigma$, and define $F(A) = \lim_s F(A \restriction s)$.

Our contradiction will be obtained by constructing a partial computable functional Φ such that

$$\mu(\{A : \Phi^{F(A)} \text{ total and 1-generic}\}) > \frac{1}{8}.$$

Let

$$p^*(\sigma) = \mu(\{A : \Phi^{F(A)} \text{ total, noncomputable, and extends } \sigma\}).$$

Kurtz's idea is to use the method of Theorem 11.16.21, but with the probability $p^*(\sigma)$ replacing $2^{-|\sigma|}$ as the measure of the sets extending σ. Notice that we can't actually compute p^* but for the first approximation to the construction, we will *pretend* that we can.

Again we will have the familiar players, the red$_e$, blue$_e$ and yellow$_e$ strings, but we will need much more subtlety in the way that they are assigned. (There will be an additional colour grey$_e$ whose role will be discussed later.)

In this construction we will be working, as usual, above a blue$_e$ string β, and search for a finite maximal pairwise incompatible extensions E_β such that for some element ρ of this set we will have

$$p^*(\beta)2^{-(e+3)} \leqslant p^*(\rho) \leqslant p^*(\beta)2^{-(e+2)}.$$

The we will give ρ the colour red$_e$ and the other members of E_β colour yellow$_e$. Since $p^*(\sigma) = p^*(\sigma 0) + p^*(\sigma 1)$, and, by Corollary 11.7.2,

$$\lim_{n\to\infty} p^*(A \restriction n) = 0,$$

for all $A \in 2^\omega$, there must exist such a set E_β. Then our construction, which is actually only a first approximation to the real one, would define Φ so that

$$\mu(\{A : F(A) \text{ total and } \Phi^{F(A)} \text{ not 1-generic}\})$$

$$\leqslant \frac{1}{2}(\mu(\{A : F(A) \text{ total and } \}) \leqslant \frac{1}{2}.$$

Then we get a contradiction since

$$\mu(\{A : \Phi^{F(A)} \text{ total and 1-generic}\}$$

$$\geqslant \mu(\{A : F(A) \text{ total and } \}) -$$

$$\mu(\{F(A) \text{ total and } \Phi^{F(A)} \text{ not 1-generic}\})$$

$$\geqslant \frac{7}{8} - \frac{1}{2} > \frac{1}{8}.$$

However, the problem with the above is that the actual value of p^* cannot be computed effectively. Thus, in the real construction, we are forced to use an approximation p to p^* defined as follows: Let

(i) $p_s(\sigma) = \mu(\cup\{\delta \in 2^s : \sigma \preccurlyeq F(\delta)\})$, and

(ii) $p(\sigma) = \mu(\{A : \sigma \prec F(A)\})$.

Then evidently, $\lim_s p_s(\sigma) = p(\sigma)$. The the key idea is that at each stage s we will use p_s in place of p^* in the real construction. The remaining details of the construction are how to overcome the difficulties that this use of approximations causes.

First we note that p_s converges to p and not to p^*. This causes two problems. Let

$$Y = \{\sigma : \mu(\{A : \sigma \prec F(A) \text{ and } F(A) \text{ is non-total or computable}\}) > \frac{1}{2}p(\sigma).$$

Then we will later prove that the set of A meeting Y has measure less than $\frac{1}{4}$. The two difficulties that arize cause the apparent density of red$_e$ strings extending certain blue$_e$ strings being too large. Our solution in both cases is to argue that in such cases the relevant red$_e$ strings actually belowen to Y, and hence will not result in an unacceptably large fraction of the domain.

The first difficulty arizes because $F(A)$ might not be total for all A. As a consequence, it might be that $p(\sigma) \neq p(\sigma 0) + p(\sigma 1)$, for some σ. Let β be an active blue$_e$ string. Suppose that we use the following strategy for making red$_e$ and yellow$_e$ strings. Begin by giving β colour red$_e$. during later stages, if ρ is the current red$_e$ string extending β, compute $p(\rho 0)$ and $p(\rho 1)$. Let i be least with

$$p(\rho i) > p(\beta)2^{-(e+3)}.$$

Should no such i exist, we need to do nothing. Should such i exist, then make ρi the new red$_e$ extension of β, giving $\rho(i-1)$ colour yellow$_e$.

The problem is that if F is not total for all A, we might well have $p(\rho) > p(\beta)2^{-(e+2)}$, but also have both $p(\rho 0)$ and $p(\rho 1)$ below $(\beta)2^{-(e+3)}$.

Thus the problem we face is the following. *Either* we allow ρ to retain the colour red$_e$, *or* we assign it to one of its children.

In the first case, we might well lose an unacceptably large fraction of the potential domain in this attempt to meet R_e. In the second case, there is the risk of having insufficient density of green$_e$ strings if Φ^β can be extended to meet V_e for the sake of R_e.

Kurtz's solution to this dilemma is to give one of the ρi the colour red$_e$ if

$$p(\rho i) > p(\beta)2^{-(e+4)}.$$

Indeed, it may be that $p(\rho) > p(\beta)2^{-(e+2)}$ yet for $i \in \{0, 1\}$, $p(\rho i) \leqslant p(\beta)2^{-(e+4)}$. Thus it would appear that the same problem presents itself. However, in this case we see that $\rho \in Y$, and hence the apparent loss of measure caused by leaving ρ with the colour red$_e$ can be blamed on Y. We will make sure that the effet on the construction of the set Y can be controlled, and will ensure that the excessive loss of measure occurs only acceptably often.

The *second* problem that presents itself is that $F(A)$ might be computable for some A. As a consequence, it may be that $\mu(F^{-1}(C)) > 0$ for some computable set C. Suppose that β is an active blue$_e$ string. It might well be the case that there is a computable set C extending β such that

$$\mu(F^{-1}(C)) > p(\beta)2^{-(e+4)}.$$

However, we may always choose our red$_e$ strings ρ extending β so that $\rho \prec C$. The first problem is that this causes the finite maximal set of pairwise incompatible extensions of β to be infinite in the limit, since since the cardinality of this set increases by 1 each time ρ is deleted from it whilst both $\rho 0$ and $\rho 1$ are added to it. Secondly, along C we still lose an unacceptably large amount of measure whilst we are attempting, vainly, to settle upon a final red$_e$ string.

The first problem turns out to be irrelevant: whether the final set of extensions of red$_e$ or yellow$_e$ strings extending β is finite or infinite is immaterial; the only question is whether the *measures* assigned to the colours are correct. The second question is more important. However, if $\mu(F^{-1}(R)) > p(\beta)2^{-(e+2)}$, then by countable additivity, $p(C \upharpoonright n) > \frac{\mu(F^{-1}(C))}{2}$, for all sufficiently large n. (This is the gist of Lemma 11.18.5.) Then such $C \upharpoonright n$ must be in Y. Thus we can again use Y to save the day, since the measure loss due to Y is controlled.

The last problem we must oversome is that we will be using the estimates p_s of p within the construction. Thus it might well be that we have settled on a red$_e$ string ρ extending a blue$_e$ string β such that $p_s(\beta)2^{-(e+2)} > p_s(\rho) \geqslant p_s(\beta)2^{-(e+4)}$. However, at a later stage t, $p_t(\beta)$ could well grow due to finding more strings δ with $F(\delta)$ extending β. Of course, all of this additional measure may not contribute to $p_t(\rho)$. As a consequence the ratio of $p_t(\rho)$ over $p_t(\beta)$ may become unacceptably small. Since we need to only assure that the density of red$_e$ and green$_e$ strings above each blue$-e$ strings be bounded away from zero, we can *declare* that "unacceptably small" mean that the apparent density be below $2^{-(e+6)}$.

This where the new colour grey$_e$ comes in. These new strings will act as "traffic lights" with the red$_e$ and yellow$_e$ strings occurring as extensions of grey$_e$ strings, whilst the blue$_e$ strings control the placement of the grey$_e$ strings. In this construction, it will now be possible for one blue$_e$ string to extend another. With this in mind, we say that a blue$_e$ string β_0 *belongs* to a blue$_e$ string β if β is the longest blue$_e$ string extended by β_0. A green$_e$ or grey$_e$ string γ is said to belong to β is β is the longest blue$_e$ string (not necessarily properly) extended by γ. Finally, a red$_e$ string ρ belongs to β if ρ extends a grey$_e$ string belonging to β.

The idea is that if β is a blue$_e$ string, then either no string extending β possesses any colour, or else the set of green$_e$, blue$_e$ and grey$_e$ strings which belong to β form a finite maximal set of incomparible extensions of β. Moreover, when we compute the density of green$_e$ and red$_e$ strings which extend a blue$_e$ string β then we will only use those that actually belong to β. This will work as follows:

Suppsoe that β is an active blue$_e$ string at some stage s, when $p_s(\beta)$ is positive. (No action will be taken until $p_s(\beta)$ is positive.) We begin by giving β itself the colour grey$_e$. Suppose that at some stage $t > s$ we appoint a red$_e$ string ρ extending β, and $p_t(\rho) \geqslant p_t(\beta)2^{-(e+4)}$. At an even later stage $u > t$, the value of $p_u(\beta)$ might have increased, so that now

$$p_u(\rho) < p_u(\beta)2^{-(e+4)}.$$

At this stage u it would appear that the density of red$_e$ strings above β is now unacceptably small. (We remark that ρ might even have changed its colour to green$_e$ before stage u. In this case the construction will have ensured that β would have lost the colour grey$_e$, and each other string extending β would have lots its colour. We would then have given all active strings except ρ the colour blue$_e$. However, the same problem is still present. The density of green$_e$ strings belonginf to β is still unacceptably small.)

Our action is to remove all colours from all the strings extending β, and remove colour grey$_e$ from β. Let $\{\alpha_0, \ldots, \alpha_n\}$ be a complete list of active strings which extend β at stage u. Then we will do nothing above β unless a stage $v > u$ occurs where

$$\frac{p_v(\beta)}{2} < \sum_{j=0}^{n} p_v(\alpha_j).$$

The key observation is that some such stage v must exist unless $\beta \in Y$, and this can be handled, as we remarked earlier. Now, should such a stage v occur, we will give each α_j the colour grey$_e$, and for each j seek red$_e$ extensions of α_j.

Finally, if at some still later stage $w > v$, $p_w(\beta)$ increases again so that the density of red$_e$ and green$_e$ strings belonginf to β is unacceptably small we will again *wipe out* above β and try to place grey$_e$ strings above β as before. We remark that wipe-outs can occur at most finitely often, since the apparent measure at β must increase four fold for the first wipe-out to occur and then double for each successive wipe-out. After the final wipe-out, we can meet R_e above β without any interference.

We now turn to the details. As we began, if the Theorem were to fail, there would need to be a a procedure Ψ_e such that Ψ_e such that

$\mu(\{A : \Psi_e^A$ total, noncomputable and does not bound a 1-generic set$\}) > 0$.

By the Lebesgue Density Theorem, there is a σ such that

$\mu(\{A : \sigma \prec A \wedge \Psi_e^A$ total, noncomputable and does not bound a 1-generic set$\}) > \frac{7}{8}\mu(\sigma)$.

Define the operator

$$\Phi_e^A = \Psi_e^{\sigma \frown A}.$$

Then

$\mu(\{A : \Phi_e^A$ total, noncomputable and does not bound a 1-generic set$\}) > \frac{7}{8}$.

Now we define F, and p_s as in the intuitive discussion.

Construction

Stage 0. The only active string is λ. Give λ the colour blue$_0$, and define $\Phi^\lambda = \lambda$.

Stage $s+1$.

Substage 1. (Red action) For $e = 0, \ldots, s+1$, do the following. for each red$_e$ string ρ adopt the case below that pertains.

Case 1. There is a $\nu \in V_{e,s}$ with $\Phi_s^\rho \preccurlyeq \nu$.

Action. Define $\Phi^r ho_{s+1} = \nu$. Give ρ colour green$_e$. Let γ be the unique grey$_e$ string extended by ρ. Remove the colour grey$_e$ from γ. If γ additionally has colour red$_e$ (that is, $\gamma = \rho$) remove this colour as well.

Case 2. Otherwise.

Action. Do nothing.

Substage 2. (Grey placement, or wipe-out recovery) For each blue$_e$ string β there are adopt the case below that pertains.

Case 1. There is a finite maximal pairwise incompatible extensions of β each of which has colour grey$_e$, blue$_e$ or green$_e$.

Action. Do nothing.

Case 2. Otherwise.

Action. We assume that no string properly extending β can have any colour, and β does not have the colour grey$_e$. Let $\{\alpha_0, \ldots, \alpha_n\}$ be a complete list of active strings which extend β. If

$$\frac{p_s(\beta)}{2} < \sum_{j=0}^{n} p_s(\alpha_j),$$

give $\alpha_0, \ldots, \alpha_n$ each colours grey$_e$ and red$_e$.

Substage 3. (Red push out) For each red$_e$ string ρ do the following. Let γ be the unique grey$_e$ string that ρ extends. if both

(i) $\frac{p_{s+1}(\rho 0)}{p_{s+1}(\gamma)} < 2^{-(e+4)}$ and

(ii) $\frac{p_{s+1}(\rho 1)}{p_{s+1}(\gamma)} < 2^{-(e+4)}$,

then do nothing. Otherwise, remove the red$_e$ colour from ρ and give red$_e$ to ρi for the least such i, and yellow$_e$ to $\rho(1-i)$. Define

$$\Phi_{s+1}^{\rho 0} = \Phi_{s+1}^{\rho 1} = \Phi_s^r ho.$$

Substage 4. (Wipe-outs) For each $e \leqslant s+1$ do the following. For each blue$_e$ string β let $\delta_0, \ldots, \delta_n$ be a complete list of those red$_e$ or green$_e$ strings that belong to β. If

$$\frac{\sum_{j=0}^{n} p_{s+1}(\delta_j)}{p_{s+1}(\beta)} > 2^{-(e+6)},$$

do nothing. Otherwise, every string extending β loses its colours, and β loses the colour grey$_e$, were it to have it.

Substage 5. (Blue placement) For each non-active string α, choose the case below that pertains.

Case 1. There is a blue$_e$ string β extended by α such that no string extended by β has any colour, and β is not grey$_e$.

Action. In this case we are still attempting wipe-out recovery, and will do nothing.

Case 2. Otherwise.

Action. Let e be least such that α does not extend a green$_e$ nor a yellow$_e$ string. Give α colour blue$_e$.

—bf End of Construction

We need the following classes:

$$B_e = \{A : A \text{ meets a string with final colour green}_e \text{ or yellow}_e\}.$$

$$S_e = \{A : A \text{ meets a string with final colour rede}\}.$$

Let Y be as in the intuitive discussion, and let

$$\widehat{Y} = \{A : A \text{ meets } Y\}.$$

Lemma 11.18.3. $\mu(\{A : F(A) \in \widehat{Y}\}) < \frac{1}{4}$.

Proof. Let $T = \{\sigma : \sigma \in Y \wedge (\forall \tau \not\preccurlyeq \sigma)(\tau \notin Y)\}$. Then T is the maximal pairwise incompatible subsets of Y. We note that

$$\frac{1}{8} \geqslant \mu(\{A : F(A) \text{ is non-total or computable}\})$$

$$\geqslant \mu(\{A : \exists \sigma \in T(\sigma \prec F(A) \text{ and } F(A) \text{ is computable or non-total})\})$$

$$= \sum_{\sigma \in T} \mu(\{A : \sigma \prec F(A) \text{ and } F(A) \text{ is computable or non-total})\})$$

$$> \frac{\sum \sigma \in Tp(\sigma)}{2}.$$

Hence $sum\sigma \in Tp(\sigma) < \frac{1}{4}$. However, since $p(\sigma) = \mu(\{A : \sigma \prec F(A)\})$, t follows that $\mu(\{A : F(A) \in \widehat{Y}\}) < \frac{1}{4}$. □

Lemma 11.18.4. $B_e \subseteq B_{e-1}$.

Proof. This works the same was as Lemma 11.17.8. □

Lemma 11.18.5. $\mu(\{A : F(A)$ *is computable and not in* $\widehat{Y}\}) = 0$.

Proof. Assume not. Then there is a single computabel set C such that

$$\mu(\{A : F(A) = C\}) > 0,$$

and $C \notin \widehat{Y}$. Evidently, $\{A : F(A) = C\} = \cap_n \{A : C \restriction n \prec F(A)\}$. By countable additivity, there must be some n such that $\mu(\{A : C \restriction n \prec F(A)\} > \frac{1}{2}\mu(\{A : F(A) = C\}$. The point is that $C \restriction n \in Y$, by definition of Y. Therefore, $C \in \widehat{Y}$, a contradiction. □

Lemma 11.18.6. $\mu(\{A : F(A)$ *total and in* $B_{e-1} - (B_e \cup S_e \cup \widehat{Y})\}) = 0$.

Proof. We remark that the intuitive reason that this lemma is true is that if $F(A)$ is total and in $B_{e-1} - (B_e \cup S_e \cup \widehat{Y})$, then we will have made infinitely many distinct attempts to satisfy R_e for A. As we prove below, probability that this happens is $\Pi_i(1 - 2^{-(e+9)}) = 0$. Kurtz's proof below is very technical, relying on some nontrivial measure theory. The remainder of the proof is easy once we have proved Lemma 11.18.6.

Let $F^+ :^\omega 2 \mapsto^\omega 3$, be a total function defined by $F^+(A; n) = 2$ iff $F(A; n) \uparrow$. Then F^+ is Borel, since the inverse image of a Borel subset of ${}^\omega 3$ is Borel subset of ${}^\omega 2$. We define $p^+(\sigma) = \mu(\{A : \sigma \prec F^+(A)\})$ in ${}^\omega 3$. Finally, we let $\overline{p^+}(U) = \mu(\{AF(A) \in U\}$, for Borel subsets U of ${}^\omega 3$. Then we have $p^+(\sigma) = p(\sigma)$ for $\sigma \in 2^{<\omega}$. Then $p^=$ generates a measure on the algebra on ${}^\omega 3$ generated by the basic open sets. Then by a theorem of Carathéodory (Royden [254], page 257), there is a unique extension of

p^+ to a measure on the Borel sets of ${}^\omega 3$. This must be $\overline{p^+}$. By the defining property of Carathéodory's measure, we have

$$\overline{p^+}(U) = \inf\{\sum_{i \in \mathbb{N}} p^+(\sigma_i)\},$$

where σ_i ranges over sequences of basic open intervals with $U \subset \cup_{i \in \mathbb{N}} \sigma_i$. The Lebesgue Density Theorem can be proved for $\overline{p^+}$, obtaining the following basic result.

For every Borel $U \subseteq^\omega 3$, and each $\delta > 0$, there is a basic open set $\sigma \in^\omega 3$ such that the $\overline{p^+}$-density of U in σ is at least δ

Now, if Lemma 11.18.6 is false, then

$$p^+(\{A : A \in B_{e-1} - (B_e \cup S_e \cup \widehat{Y}) \wedge (foralln)[A(n) \neq 2]\}) > 0.$$

This follows from the definitions of F^+ and $\overline{p^+}$. Thus there is a basic open interval $\sigma \in^\omega 3$ such that the $\overline{p^+}$ density of $\{A : A \in B_{e-1} - (B_e \cup S_e \cup \widehat{Y}) \wedge (\forall n)[A(n) \neq 2]\}$ exceeds $1 - 2^{-(e+9)}$. Since B_{e-1} is the union of basic open intervals determined by strings whose final colour is yellow$_{e-1}$ or green$_{e-1}$, we can assume that σ extends such a string τ.

Notice that σ must be in $2^{<\omega}$ or else $A(n) = 2$ for some n for every A with $\sigma \prec A$. We claim that σ cannot extend a string γ with final colour grey$_e$. If not, then either $A \in B_e$, $B \in S_e$, $(\exists n)[A(n) = 2]$, or else A is computable, and is the limit of a sequence of red$_e$ strings. In the last case, $A \in \widehat{Y}$, by Lemma 11.18.5.

It follows that we appoint a maximal pairwise incompatible set $\{\beta_0, \dots, \beta_n\}$, of extensions iof σ each of which receive the colour blue$_e$. Now either $\sigma \in \widehat{Y}$ (and we are done), or else there is a β_j such that the p^+-density of $\{A : A \in B_{e-1} - (B_e \cup S_e \cup \widehat{Y}) \wedge (\forall n)[A(n) \neq 2]\}$ exceeds $1 - 2^{-(e+8)}$ above β_j. After β_j has wiped-out for the last time, we appoint a maximal pairwise incompatible set $\{gamma_0, \dots, \gamma_m\}$ of grey$_e$ strings exceeding β_j. Let δ_i^s denote the red$_e$ or green$_e$ string extending γ_i during stage s, and let $\delta_i^s = \gamma_i$ if no string has been appointed at stage s. Let

$$X_s = \{A : \exists i(\delta_i^s \prec A\}.$$

Let $X = \cap_s X_s$. Then if $A \in X$, we claim that $A \in (B_e \cup S_e \cup \widehat{Y})$. There are two cases. The first case is that $\lim_s \delta_i^s = \delta_i$ is a string. Then it follows that this string δ_i must have final colour green$_e$ or red$_e$. The other case is that $\lim_s \delta_i^s = C$, a computable set. Then clearly $p^+(\{C\}) > 0$, and hence $A = C \in \widehat{Y}$.

Therefore, it is enough to show that the p^+-density of X_s in β_j exceeds $2^{-(e+8)}$. Assume not. By countable additivity, $\overline{p^+}(X_s) < 2\overline{p^+}(X)$ for sufficiently large s. Thus, $\frac{\overline{p^+}(X_s)}{\overline{p^+}(\beta_j)} < 2^{-(e+7)}$, for sufficiently large s. Fix such an s. By countable additivity, for all sufficiently large $t > s$,

$$\sum_{i<n} p^t(\delta_i^s) < 2p^+(X_s).$$

Therefore, as $\delta_i^s \preccurlyeq \delta_i^t$, we have $\sum_{i<n} p^t(\delta_i^t) \leqslant \sum_{i<n} p^t(\delta_i^s)$. Thus, $\frac{\sum_{i\in\mathbb{N}} p^t(\delta_i^t)}{p_t(\beta_j)} < 2^{-(e+6)}$. Therefore, β_j must wipe-out again, contradicting the assumption that β_j had wiped-out for the final time. □

Lemma 11.18.7. $\mu(\{A : F^+(A) \in (S_e - \widehat{Y})\}) \leqslant 2^{-(e+2)}$.

Proof. Let $\{\gamma_i : I < k\}$ be a complete list of strings with colour grey$_e$. (Here $k \leqslant \omega$.) Then $\{\gamma_i : i < k\}$ is a set of pairwise incompatible strings. For each i let ρ_i denote the red$_e$ string extending γ_i at stage s, with $\rho_i = \gamma_i$ if none yet appointed.

Suppose that $\frac{p^+(\rho_i^s)}{p^+(\gamma_i)} > 2^{-(e+2)}$, for all s. Then we claim that ρ_i^s is a member of Y for all sufficiently large s. Again there are two cases. First, $\lim_s \rho_i^s = \rho_i$ is a string. Then $p(\rho_i 0) + p(\rho 1) < \frac{p(\rho)}{2}$, lest rho_i^t would become ρi for some i at some large t. Thus $\rho_i \in Y$. The other case is that $\lim_s \rho_i^s = C$ a computable set. By countable additivity, $\mu(\{C\}) > 0$. Again we see that ρ_i^s must be contained in $\widehat{Y}$ at all sufficiently large s, by the same analysis we used in the proof of Lemma 11.18.5. Now the result follows easily.

$$\mu(\{A : F^+(A) \in (S_e - \widehat{Y})\}) \leqslant p^+(S_e - \widehat{Y})$$

$$\leqslant \sum \{p^+(\rho) : \rho \text{ has final colour red}_e \text{ and } \rho \notin Y\}$$

$$\leqslant \sum 2^{-(e+2)} \{p^+(\gamma) : \gamma \text{ has final coulour grey}_e\} \leqslant 2^{-(e+2)}.$$

□

Lemma 11.18.8. $\mu(\{A : F(A)$ *total and an element of* $\cap_e B_e\}) > \frac{1}{8}$.

Proof.

$$^\omega 2 = \{A : F(A) \text{ total and an element of } \cap_e B_e\} \cup$$

$$\cup_e \{A : F(A) \text{ total and an element of } B_{e-1} - (B_e \cup S_e \cup \widehat{Y})\} \cup$$

$$\{A : F(A) \text{ total and an element of } S_e\} \cup$$

$$\{A : F(A) \text{ total and an element of } \widehat{Y}\} \cup$$

$$\{A : F(A) \text{ not total}\}.$$

Therefore

$$1 < \mu(\{A : F(A) \text{ total and an element of } \cap_e B_e\}) + \sum_e 0 + \sum_e 2^{-(e+2)} + \frac{1}{4} + \frac{1}{8}.$$

Hence, $\mu(\{A : F(A)$ total and an element of $\cap_e B_e\}) > \frac{1}{8}$, as required. □

Lemma 11.18.9. *If $A \in \cap_e B_e$, then A is 1-generic.*

Proof. (sketch) The argument if now pretty straightforward. Suppose that $A \in \cap_e B_e$. Then let $T_m = \{\sigma : |\sigma| = m\}$. If m' is a c.e. index for T_m, then by definition of $R_{m'}$, $\Phi^A(n)$ etends to a string with final colour green$_{m'}$ or yellow$_{m'}$ for all $n < m$ and hence Φ^A is total.

To see that Φ^A is 1-generic, let V_e be a c.e. set of strings. Since $A \in \cap_e B_e$ there is a string σ with final colour green$_e$ or yellow$_e$ extended by A. In the case of green$_e$, then Φ^A meets V_e by the construction. In the yellow$_e$ case, by construction, there cannot be an extension of Φ^A_s in V_e else R_e would receive attention. □

□

One corollary from this was one of the first difficult theorems about measure and randomness.

Corollary 11.18.10 (Paris [236])**.** *The upward closure of the minimal degrees has measure 0.*

Indeed, Kurtz points out that since the Jockusch [133] proved that the initial segment below a 1-generic degree is never a lattice, we get the following.

Corollary 11.18.11 (Kurtz [165])**.** *The upward closure of of degrees whose initial segments form lattices has measure 0.*

11.19 Solovay genericity and randomness

Solovay [283] used forcing with closed sets of positive measure to construct a model of set theory (without the Axiom of Choice) in which every set of reals is Lebesgue measurable. In the same way that Cohen forcing is miniaturized to give the notion of n-genericity, Kautz [140] gave a miniaturization of Solovay's notion that yields a characterization of weak n-randomness in terms of a forcing relation. Thus, while there is no correlation between Cohen forcing and randomness, we do have a notion of forcing related to algorithmic randomness.

The construction of a hyperimmune-free degree and many other constructions in set theory and computability theory use infinite conditions such as perfect trees. Kautz proved that the forcing relation where the conditions are Π^0_n classes coincides with weak n-randomness. Let $\mathbb{P}_n$ denote the partial ordering of Π^0_n classes of positive measure, ordered by inclusion.

Definition 11.19.1 (Kautz [140])**.** (i) If $T \in \mathbb{P}_n$, we say that $T \Vdash \varphi$ if $A \models \varphi$ for all $A \in T$.

(ii) We say that a real A forces φ iff there exists a $T \in \mathbb{P}_n$ such that $A \in T$ and $T \Vdash \varphi$.

(iii) A is *Solovay n-generic* if for every Σ_n^0 sentence, either $A \Vdash \varphi$, or $A \Vdash \overline{\varphi}$.

Lemma 11.19.2 (Kautz [140]). *(i) (Monotonicity) $T \Vdash \varphi$ implies $(\forall \widehat{T} \subset T)(\widehat{T} \in \mathbb{P}_n \to \widehat{T} \Vdash \varphi)$.*

(ii) (Consistency) It is not the case that $T \Vdash \varphi$ and $T \Vdash \overline{\varphi}$.

(iii) (Quasi-completeness) For every Σ_n^0 sentence φ, and each $T \in \mathbb{P}_n$, there is an $S \subseteq T$, such that either $S \Vdash \varphi$, or $S \Vdash \overline{\varphi}$.

(iv) (Forcing=Truth) A is Solovay n-generic iff for each Σ_n^0 or Π_n^0 sentence, φ

$$A \Vdash \varphi \textit{ iff } A \vDash \varphi.$$

Proof. We prove (iii) and (iv). Let $\{P_i : i \in \mathbb{N}\}$ be the complement of the universal n-Martin-Löf test. Since $\mu(P_i) \to 1$, there is a i such that $\mu(P_i) \geqslant 1 - 2^{-1}\mu(T)$. Thus $S \cap T \in \mathbb{P}_n$, and all of its members are n-random. If $A \vDash \varphi$ for all $A \in S$ we are done, since then $S \Vdash \varphi$. Otherwise the Π_1^0 class $U = s \cap \{A : A \nvDash \varphi\}$ is nonempty. By the 0-1 Law, Theorem 11.9.2, since this class has positive measure and the class contains only n-random reals, $U \Vdash \overline{\varphi}$.

For (iv), first suppose A is Solovay n-generic. Suppose that φ is either Σ_n^0 or Π_n^0. If $A \Vdash \varphi$, then $A \vDash \varphi$. If A fails to force φ then by Solovay genericity, $A \Vdash \overline{\varphi}$, and hence $A \nvDash \varphi$. Conversely, if the conclusion holds, thus either $A \Vdash \varphi$ or $A \Vdash \overline{\varphi}$. □

Theorem 11.19.3 (Kautz [140]). *A is Solovay n-generic iff A is Kurtz n-random.*

Proof. Suppose that A is not Kurtz n-random. Then A is in some Π_n^0 nullset, $S = \{B : B \vDash \sigma\}$ with σ Π_n^0. As $A \vDash \varphi$, A does not force $\overline{\varphi}$. On the other hand there is no Π_n^0 class of positive measure forcing φ since S has measure 0.

Conversely, suppose that A is Kurtz n-random. Now let φ be a Σ_n^0 sentence, and let $S = \{B : B \vDash \varphi\}$. S is a union of Π_{n-1}^0 classes. If $A \in S$, then A is in some class C_i having positive measure since A is Kurtz n-random. Hence $C_i \Vdash \varphi$. If $A \notin S$, then similarly A is in the Π_n^0 class $\overline{S} = \{B : B \vDash \overline{\varphi}\}$, again having positive measure, meaning that $\overline{S} \Vdash \overline{\varphi}$. □

12

von Mises strikes back-selection revisited

12.1 Monotone selection

This last chapter is a particularly appropriate way to finish in that we will return to the roots of the subject, re-examining von Mises ideas of selection; and we will also finish at the very forefront of the development of the subject with a fundamental open question.

In particular, in this chapter we will look at some new interpretations of the notion of selection and look at a possible refutation of Schnorr's critique of Martin-Löf randomness.

The fundamental intuition in the present chapter is that it is not the notion of computable *selection* that causes the problem with the von-Mises Church Wald notion of randomness, *rather* it is *monotonic* nature of the rules we have used. However, this anticipates things somewhat.

Let's first return to von Mises intuition of selection. The basic idea is that a selection rule is one that given some bits of a real predicts another bit.

Definition 12.1.1. A *selection rule* is a partial function $r : 2^{<\omega} \mapsto \{yes, no\}$.

The subsequence of a sequence A *selected* by a selection rule r is that with $r(A \upharpoonright n-1) = yes$. The sequence of selected places are those n_i such that $r(A \upharpoonright n_i - 1) = yes$. Then for a given selection rule r and a given real A, we generate a sequence $n_0, n_1, \ldots$ of selected places, and we say that a real is *stochastic with respect to admissible selection rules* iff for any such selection rule, either the sequence of selected places is finite, or

$$\lim_{d\to\infty} \frac{\{i \leqslant d : A(n_i) = 1\}}{d} = \frac{1}{2}.$$

For the time being, the selection rules will be *monotonic* in that the generated sequence is increasing.

Definition 12.1.2. (i) We say that a real is *von-Mises-Church-Wald stochastic* iff it is stochastic for all partial computable (monotonic) selection rules.

(ii) We say that a real is *Church stochastic* iff it is stochastic for all computable selection rules.

There is a canonical way of turning a selection rule into a martingale. Namely, initially have $F_0(\lambda) = 1$. Then for n_0, the first selected place, double the bet on the 1's. That is, for example, if $n_0 = 2$ so that 2 is the first selected place, we would have $F_1(00) = F(10) = 1$, $F_1(\sigma) = F_1(\tau) = 2$, for all σ with $01 \preccurlyeq \sigma$, and τ with $11 \preccurlyeq \tau$, and we'd raise $F_1(0) = F_1(1) = 1+\frac{1}{2}$, and $F_1(\lambda) = 1 + \frac{1}{4}$, and we would continue this process in the obvious way. Clearly this process will give a martingale with $\lim_s F_s(\lambda) \leqslant 2$. The martingale is computable if the selection rule is. The following results are therefore immediate.

Theorem 12.1.3. *(i) If α is Martin-Löf random, then α is von-Mises-Church-Wald stochastic.*

(ii) If α is computably random, then α is Church stochastic.

The converses of the above fail to hold.

Theorem 12.1.4 (van Lambalgen [314])**.** *(i) There are von-Mises-Church-Wald stochastic reals that are not Martin-Löf random.*

(i) There are Church stochastic reals that are not computably random.

Proof. For (i) choose any real which is not 1-random but has Σ^0_1 dimension 1. The other case is similar. □

12.2 Partial computable martingales and Merkle's gap phenomenom

Actually, there are von-Mises-Church-Wald stochastic reals whose initial segment complexities are *far* from those of random reals. The following theorem follows a line of work including Daley [59] and Li-Vitanyi [185]. For the strongest statement of the results we will need a new notion. A partial computable (super-) martingale is, as the reader would expect, a partial computable $F : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$, such that if $F(\sigma i)$ is defined the so is $F(\sigma)$, and F, where defined, satisfies the (super-)martingale property. We say that real is partial computably random iff no partial computable martingale succeeds on it.

This strengthening of the notion of computable randomness where we additionally consider partial computable martingales, was first considered by Ambos-Spies and Yongge Wang. (See Ambos-Spies [5], just before the bibliography.) They are also discussed in Terwijn's Thesis [301] were they are attributed to Fortnow, Freivalds, Gasarch, Kummer, Kurtz, Smith, and Stephan [108]. Clearly, if something is partial computably random is is, *a fortiori*, von-Mises-Wald-Church stochastic.

Theorem 12.2.1 (Merkle [202])**.** *Let $\mathcal{F}$ be the collection of all computable functions $f : \mathbb{N} \mapsto \mathbb{N}$ that are nondecreasing and unbounded. Then*

(i) There is a computably random real α such that for all $f \in \mathcal{F}$, and almost all n,

$$C(\alpha \upharpoonright n|n) \leqslant f(n).$$

(ii) There is a partially computably random real β such that for all $f \in \mathcal{F}$, and almost all n,

$$C(\beta \upharpoonright n|n) \leqslant f(n) \log n.$$

Before we prove Merkle's Theorem above, we mention a nice corollary. It follows that there is a computably random set with $C(\alpha \upharpoonright n) < 2 \log n + c$, and hence we can obtain the following result of Ambos-Spies.

Corollary 12.2.2 (Ambos-Spies [5])**.** *There is a computably random real that is not von-Mises-Church-Wald stochastic.*

Proof. (i) Let $F_s = \{i \leqslant s : \varphi_i \in \mathcal{F}\}$. Let $\{d_i : i \in \mathbb{N}\}$ be an effective enumeration of all partial computable martingales with initial capital 1. We then define $D_s = \{i \leqslant s : d_i \text{ total }\}$. Merkle defines $n_0 = -0$ than then the folllowing sequence of parameters:

$$n_s = \min\{n : n > n_{s-1} \wedge \varphi_i(n) > 3s \text{ for all } i \in F_s\}.$$

$$I_s = [n_{s-1}, n_s).$$

We construct a set R in stages. Let

$$\widehat{d_s} = \sum_{i \in D_s} 2^{-(i+u_i)} d_i.$$

Note that $\widehat{d_s}$ is a martingale. In particular for any string w either $\widehat{d}(w0) = \widehat{d}(w1)$ or exactly one is smaller than $\widehat{d}(w)$

We then will define $R(n)$ for $n \in I_s$ assumeing that we have already constructed $R \upharpoonright (n-1)$. Let $w = R \upharpoonright (n-1)$. We set $R(n) = 0$ if $\widehat{d}(w0) \leqslant \widehat{d_s}(w)$, and $R(n) = 1$ otherwise.

Claim

$$\text{For all } s \text{ and } n \in I_s, \widehat{d}_s(R \upharpoonright n) < 2 - 2^{-s}.$$

We prove this claim by induction. There is nothing to prove for $n = 0$. For the induction step, since the construction of $\widehat{d}_s$ in I_s is nondecreasing, unless n is the minimum in the interval I_s. In the case that n is the minimum, the let $w = R \upharpoonright (n-1)$ and we note that

$$\widehat{d}_s(wR(n)) \leqslant \widehat{d}_s(w) = \widehat{d}_{s-1}(w) + 2^{-(s+u_s)} d_s(w)$$
$$\leqslant 2 - 2^{-(s-1)} + 2^{-s} < 2 - 2^{-s}.$$

Note that R is computably random. Otherise, suppose that d_i succeeds on R. Then by definition of $\widehat{d}_s$, by the claim, we have that for all n and almost all s,

$$2^{-(i+u_i)} d(R \upharpoonright n) \leqslant \widehat{d}_s(R \upharpoonright n) < 2.$$

To conclude the proof, we now need to show that the plain complexity is low. The key thing to note is that the the sets F_s and D_s can be coded by a string x_s or length $2s$ and that, given these two sets, the construction up to and including stage s can be simulated. Indeed this procedure can be adjusted to output, on input (i, x_s) the length i prefix of R whenever i is in one of the intervals $I_0 \ldots I_s$. Thus $C(R \upharpoonright n|n) \leqslant 2s + c$, and (i) follows by choice of n_s, since for any $f \in F_s$, we have that for almost all s and all $n \in I_s$,

$$2s + c \leqslant 3s \leqslant f(n_s) \leqslant f(n).$$

The proof of (ii) is a modification of (i). The martingale that we diagonalize against is now a convex sum of all d_i for $i \leqslant s$. except that we omit all d_i which are undefined for any prefix of w. In order to simulate the construction up to and including stage s, we only need to know about places at which one of more of the betting strategies d_i are now defined. Thus to compute $R(n)$ for $n \in I_s$, we need to supply s numbers less than or equal to n plus the set F_s. This information can be coded by $\leqslant 3s \log n$ bits, and hence needs ar most $f(n) \log n$ bits for all $f \in \mathcal{F}$ for almost all n. □

Merkle [202] also showed that being stochastic has *some* consequences for the initial segment complexity of a real. The following generalizes the known fact that no computably enumerable set can be von-Mises-Church-Wald stochastic. Together with Theorem 12.2.1 (ii), these results establish a kind of gap phenomoenom in the behaviour of von-Mises-Church-Wald stochastic sequences.

Theorem 12.2.3 (Merkle [202]). *Suppose that there is a c such that for almost all n,*

$$C(X \upharpoonright n) \leqslant c \log n.$$

Then X is not von-Mises-Church-Wald stochastic.

Proof. We prove for k. Let $c = 2k + 2$. Choose a computable sequence $m_0, m_1, \ldots$ with each m_i is a multiple of $k+1$, $m_0 \geqslant 0$ and for all i,

$$10(m_0 + \cdots + m_{i-1}) < \frac{m_i}{k+1}.$$

Now partition $\mathbb{N}$ into blocks of consecutive integers, $I_0, I_1., \ldots$ with $|I_i| = m_i$, and then divide each I_i into $k+1$ consecutive disjoint intervals, $J_i^1, \ldots, J_i^{k+1}$, each of identical length $\ell_i = \frac{m_i}{k+1}$.

We let w_i^j represent $X \upharpoonright_{\min J_i^j}^{\max J_i^j}$. We will henceforth write such restrictions as $X \upharpoonright [J_i^j]$.

Merkle's idea is the following. Assume that for some t we have a procedure which given s and $X \upharpoonright (\min J_s^t - 1)$, enumerates a set T_s of words such that

(i) w_s^t is in T_s^t for almost all s, and

(ii) $|T_s^t| \leqslant .2\ell_s$, for infinitely many s.

Then it is claimed that one of the following selection rules r_0 or r_1 will demonstrate that X is not von-Mises-Church-Wald stochastic. The idea is that r_i tries to select places in J_s^t where the corresponding bit is of X is i. For all s, let $v_s^1, v_s^2, \ldots$ be the assumes enumeration of T_s^t, and there are no repetitions. Pick s_0 so that w_s is in T_s^t for all $s \geqslant s_0$. Bothe selection rules select numbers in intervals of the form J_s^t for $s \geqslant s_0$.

On entering such an interval, r_i puts $e = 1$. It then starts scanning numbers in the interval. Assuming that $X \upharpoonright [I_s] = v_s^e$, the selection rule selects n iff the corresponding bit of v_s^e is i. This is done until *either* the end of the interval is reached or one of the scanned bits differs from the corresponding one of v_s^e. In the second case, we increment the counter e and the procedure iterates by scanning the remaining bits. Then V_s^e is always defined due to choice of s_0 and (i) above; and because iteration e is only reached in case the true word w_s is not amongst $v_s^1, \ldots, v_s^{e-1}$.

Now, for all $s \geqslant s_0$, every number in the interval J_s^t is selected by either r_0 or r_1. Ww say that a number is *selected correctly* is it is selected by r_i and the corresponding bit is i. Then in J_s^t there are at most $|T_s^t| - 1$ numbers n that are selected incorrectly. Hence, but the assumptions, for infinitely many s there are at least $.8\ell_s$ numbers in J_s^t that are selected correctly. Hence for some i and infinitely many s, the selection rule r_i selectes al least $.4\ell_s$ of the numbers in J_s^t correctly, and at most $.2\ell_s$ incorrectly. Bu the parameters concerning the sizes, there are at most $.1\ell_s$ number that r_i that could have been selected before r_i entered the interval. Hence up to and including each such interval J_s^t, the selection rule r_i selects at elast $.4\ell_s$ numbers correctly and at most $.3\ell_s$ incorrectly. That is r_i witnesses that the sequence is not von-Mises-Wald-Church stochastic.

The remainder of Merkle's proof is to show that there is a t where there is a procedure as above enumerating sets T_s^t satisfying (i) and (ii). Let $w_s = X \upharpoonright [I_s]$. Let $A_s = \{w : |w| = m_s \wedge C(w) < k \log m_s\}$. Then $w_s \in A_s$ for almost all s. To see this, by the parameter of the size of I_s, since $m_0 + \cdots + m_{s-1} < m_s$, we have

$$C(X \upharpoonright \sum_{i=0}^{s-1} m_i) \leqslant c(\log(\sum_{i=0}^{s-1} m_i)) \leqslant 2c(\log m_s).$$

Thus, for almost all s,

$$C(w_s) = C(X \upharpoonright [I_s]) \leqslant (2c+1)\log m_s = (k-1)\log m_s.$$

This $w_s \in A_s$.

We now get the sets T_s^t. For all $s \geqslant 0$ and all $j = 1, \ldots, k+1$,

$$T_s^{k+1} = \{v : w_s^1 \ldots w_s^k v \in A_s\}, \text{ and,}$$

$$T_s^j = \{v : |v| = |J_s^j| \text{ and there are at least } (.2\ell_s)^{k+1-j}$$

$$\text{strings } u \text{ such that } w_s^1 \ldots w_s^{j-1} vu \in A_s\}.$$

There is a Turing machine which, upon input s, enumerates A_s, hence there is a machine which, given j, s and $w_s^1, \ldots, w_s^{j-1}$, enumerates T_s^j. Thus, to conclude the proof, it suffices to show that (i) and (ii) are satisfied for some t.

Observe that if (ii) is not satisfied sfor some $t > 0$, then (i) is satisfied for t replaced by $t-1$. For if (ii) is not satisfied then for almost all s, there are at least $.2\ell_s$ strings in T_s^t where each of these strings can vbe extended by at least $(.2\ell_s)^{k+1-t}$ strings u to a string $w_s^1 \ldots w_s^{t-1} vu$ in A_s. Thus, for each such s, there are at least $(.2\ell_s)^{k+1-(t-1)}$ strings vu that extend $w_s^1 \ldots w_s^{t-1}$ to a strings in A_s. That is, for almost all s, the string $w_s^{t-1} \in T_s^t$.

Condition (i) is satisfied for $t = k+1$, so if (ii) is satisfied as well, we will be done by letting $t = k+1$. Otherwise, by the previous argument, we know that (i) is satisfied for $t = k$ and we can iterate the argument. Proceedings this way it suffices to argue that (ii) cannot fail for $t = 1$. Assuming that it does, for almost all s, there are at least $.2\ell_s$ many assignments on J_s^1 which can be extended in $(.2\ell_s)^k$ ways to strings in $A - s$. Thus for sufficiently large s and some $\varepsilon > 0$,

$$|A_s| \geqslant (.2\ell_s)^{k+1} \geqslant (.2\frac{m_s}{k+1})^{k+1} = \varepsilon m_s^{k+1} > m_s^k.$$

This is a contradiction. A_s has, by definition at most M_s^k members because for any n there are fewer thatn 2^n strings w with $C(w) < n$. $\square$

12.3 Stochasticity and martingales

Ambos-Spies, Mayordomo, Wang and Zheng [10] have shown that stochasticity can be viewed as a kind of randomness for *restricted* kind of martingales. These authors stated the following for time bounded stochasticuty, but their proof works for Church stochasticity also.

Definition 12.3.1 (Ambos-Spies, Mayordomo, Wang and Zheng [10]). (i) A martingale d is called *simple* if there is a number $q \in \mathbb{Q} \cap (0,1)$ such that for all σ, and $i \in \{0,1\}$,

$$d(\sigma i) \in \{d(\sigma), (1+q)d(\sigma), (1-q)d(\sigma)\}.$$

(ii) We say that a martingale d is *almost simple* iff there exists a finite set $\{q_1, \ldots, q_m\}$ of rationals such that for all σ and i, there is a $1 \leqslant k \leqslant m$ such that

$$d(\sigma i) \in \{d(\sigma), (1+q_k)d(\sigma), (1-q_k)d(\sigma)\}.$$

We will say that a real is *(almost) simply random*[1] iff no (almost) simple computable martingale succeeds on it. Actually the distinction between almost simple randomness and simple randomness is illusory:

Lemma 12.3.2 (Ambos-Spies, Mayordomo, Wang and Zheng [10]). *Suppose that d is an almost simple martingale. Then there are at most finitely many simple martingales $d_1, \ldots, d_m$ such that*

$$S^{\infty}[d] \subseteq \cup_{k=1}^{m} S^{\infty}[d_k].$$

Proof. Suppose that d is almost simple with rationals $\{q_1, \ldots, q_k\}$. For each i, define a simple martingale d_k which copies d (with d_k in place of d) for all σ with $d(\sigma i) \in \{d(\sigma), (1+q_k)d(\sigma), (1-q_k)d(\sigma)\}$. and defines $d_k(\sigma i) = d_k(\sigma)$, otherwise. The it is clear that

$$S^{\infty}[d] \subseteq \cup_{k=1}^{m} S^{\infty}[d_k].$$

□

Corollary 12.3.3. *A real is almost simply random iff it is simply random.*

We can now state the characterization of Church stochasticity.

Theorem 12.3.4 (Ambos-Spies, Mayordomo, Wang and Zheng [10]). *(i) A real is Church stochastic iff it is random for all computable (almost) simple martingales.*

(ii) A real is von Mises Church Wald stochastic iff it is random for all partial computable (almost) simple martingales.

[1] Ambos-Spies et. al. use the terminimogy "weakly random" which is already used with relation to Kurtz randomness.

Proof. We do (i) as (ii) is essentially the same.

Suppose that A is Church stochastic and yet there is a simple computable martingale d with rational q that succeeds upon A. We define a computable selection function f by letting $f(X \upharpoonright x) = yes$ if $d((X \upharpoonright x-1)i) \neq d(X \upharpoonright x)$, and letting $f(X \upharpoonright x) = no$ if $d((X \upharpoonright x-1)0) = d(X \upharpoonright x)$. Then f is (partial) computable if d is. Moreover, since d succeeds upon A,

$$\limsup_n \frac{|\{y < n : d((A \upharpoonright y-1)A(y)) = (1+q)d(A \upharpoonright y-1)\}|}{|\{y < n : d((A \upharpoonright y-1)A(y)) = (1-q)d(A \upharpoonright y-1)\}|} > 1.$$

Thus the limsup of the ratio of the places of A selected by f which are 1 over the total number, is bigger than $\frac{1}{2}$ and hence A is not Church stochastic.

Now for the other direction. We assume that A is random for all computable martingales, but there is a computable selection function f that demonstrates that A is not Church stochastic. By symmetry we may assume that there is a rational number ε such that

$$\liminf_n \frac{|\{y < n : f(A \upharpoonright y-1) = yes \wedge A(y) = 1\}|}{|\{y < n : f(A \upharpoonright y-1) = yes \wedge A(y) = 0\}|} =_{\text{def}} \frac{p(n)}{q(n)} < 1 - \varepsilon.$$

Now choose a rational number $\alpha \in (0,1)$ such that

$$-\frac{\log(1+\alpha)}{\log(1-\alpha)} > 1 - \frac{\varepsilon}{2}.$$

Then

$$(1+\alpha) \geqslant (\frac{1}{(1-\alpha)})^{1-\frac{\varepsilon}{2}}.$$

Then we may define a simple martingale d by $d(\lambda) = 1$, and if $f(\sigma) = yes$, letting $d(\sigma 1) = (1-\alpha)d(\sigma)$, $d(\sigma 0) = (1+\alpha)d(\sigma)$ and defining $d(\sigma i) = d(\sigma)$ if $f(\sigma) = no$. Then d is a computable simple martingale. Also

$$d(A \upharpoonright n) = (1-\alpha)^{p(n)}(1+\alpha)^{q(n)}$$

$$\geqslant (1-\alpha)^{p(n)}(\frac{1}{(1-\alpha)})^{(1-\frac{\varepsilon}{2})q(n)} = (\frac{1}{(1-\alpha)})^{(1-\frac{\varepsilon}{2})q(n)-p(n)}$$

However, we know that $p(n) < (1-\varepsilon)q(n)$ infinitely often, and hence

$$d(A \upharpoonright n) \geqslant (\frac{1}{(1-\alpha)})^{\frac{\varepsilon}{2}q(n)}.$$

Hence d succeeds upon A. □

12.4 Nomonotonic randomness

12.4.1 Nonmonotonic betting strategies

We begin by re-considering the material in the light of *nonmonotonic* selection. This concept was introduced by Muchnik, Semenov, Uspensky [222].

Let's be more precise about our betting strategies. We use the notation of Merkle, Miller, Nies, Reimann, Stephan [207].

Definition 12.4.1. An ordered finite assignment (f.a.) is a sequence

$$x = (r_0, a_0) \dots (x_n, r_n) \in (\mathbb{N} \times \{0,1\})^*,$$

consisting of natural numbers and bits.

The r_i will be the places selected in the strategy. We call this the (selection) *domain* and denote it by $\mathrm{dom}(x)$. The function which determines the next place to be selected is called the *scan rule*. Thus a scan rule is a partial function f from the set of finite assignments to $\mathbb{N}$ such that for all finite assignments w, $s(w) \notin \mathrm{dom}(x)$.

Definition 12.4.2 (Merkle et. al. [207])**.** A stake function is a partial function from the collection of finite assignments to $[0, 2]$. Then we can formally define a *nonmonotonic betting strategy* as a pair which consists $b = (s, q)$ consisting of a scan rule, and a steke function q.

Thus, we can use this idea to define the analog of a supermartingale for nonmonotonic betting strategies. This is called in [207] a capital function, d. Clearly, what will happen is that, given a real X, we begin with $d(\lambda)$, and given a finite assignment x, the strategy picks $s(x)$ as the next place to bet upon. If $q(x) < 0$, it bets $X(s(x)) = 1$ and if $q(x) > 1$, it bets $X(s(x)) = 0$, and it places no bet if $q(x) = 1$. Then as in the case of a martingale, if $X(s(x)) = 0$, the current capital is multiplied by $q(x)$ and otherwise it is multiplied by $2 - q(x)$. (That is if the strategy makes the correct guess its stake is doubled, else lost.)

We can consider nonmonotonic betting strategies as a game played on reals. Let $b = (s, q)$ be a non-monotonic betting strategy. We define the partial function as the partial play via (dropping the subscript b in the below) $p^X(0) = \lambda$, and

$$p^X(n+1) = p^X(n)\widehat{(}s(p^X(n)), q(P^X(n)).$$

We regard $p^X(n+1)\uparrow$ should $s(p^X(n))$ be undefined. Using this we can formally define the payoff as $c^X(n+1) = q(p^X(n))$ if $X(p^X(n)) = 0$ and $c^X(n+1) = 2 - q(p^X(n))$, otherwise. Thus we can finally define the *payoff function*, a nonmonotonic martingale as

$$d^X(n) = d^X(\lambda)\Pi_{i=1}^n c^X(i).$$

Definition 12.4.3 (Muchnik, Semenov, Uspensky [222])**.** A nonmonotonic betting strategy b succeeds on X iff

$$\limsup_{n\to\infty} d_b^X(n) = \infty.$$

We say that X is *Kolmogorov-Loveland random* if no partial computable non-monotonic betting strategy succeeds on it.

Actually, the use of partial and total computable nonmonotonic strategies gives the same result.

Theorem 12.4.4 (Merkle [203]). *X is Kolmogorov-Loveland random iff no total computable non-monotonic betting strategy succeeds on it.*

Proof. we need only show that no total non-monotonic strategy succeeding implies that no partial one does either. Thus take real A and a partial computable strategy (s, b) that fails on A. This strategy selects a sequence of places $s_0, s_1, \ldots$ so that infinite capital is gained on the sequence $A(s_0), A(s_1) \ldots$. Now we can decompose this sequence into evens and odds and note that the winning must be true of either an infinite odd sequence, or evens. Without loss of generality, sucoose it is the evens.

The idea is then to build a computable betting strategy that emulates (s, b) on the evens, but, whilst it is waiting for convergence on some even, it plays fresh odds but makes no biased bet. Then if ever we get convergence on the current even being scanned, it goes back to emulating (s, b). $\square$

is waiting for convergenge at any stage, i

The fundamental properties of Kolmogorov-Loveland randomness were first established in Muchnik, Semenov, Uspensky [222]. Many of these were improved by the later paper of Merkle, Miller, Nies, Reimann, Stephan [207], which by and large, used techniques which were elaborations of those of [222].

Naturally, we will define $C \subseteq 2^\omega$ as a *Kolmogorov-Loveland nullset* if there is a partial computable nonmonotonic betting strategy succeeding on all $X \in C$.

Theorem 12.4.5 (Muchnik, Semenov, Uspensky [222]). *Suppose that A is 1-random. Then A is Kolmogorov-Loveland random.*

Before we prove this result we remark that, at the time of writing of this book, it remains a fundamental open question whether this theorem can be reversed.

Question 12.4.6 (Muchnik, Semenov, Uspensky [222]). *Is every Kolmogorov-Loveland random real Martin-Löf random?*

We remark that most workers feel that the answer is no.

Proof. Suppose that A is not Kolmororov-Loveland random and that b is a partial computable betting strategy that succeeds upon it. We define a Martin-Löf test $\{V_n : n \in \mathbb{N}\}$ as follows. We put $[\sigma]$ into V_n if $d_b^\sigma(j)$ achieves 2^n or greater, say by a series of plays at places $s(0) \ldots s(m)$, and hence $|\sigma| \geqslant m$. By Kolmogorov's inequlaity, $\{V_n : n \in \mathbb{N}\}$ is a Martin-Löf test and hence, since $A \in \cap_n V_n$, A is not 1-random. $\square$

The same proof shows that if we looked at computably enumerable nonomotonic betteing strategies, then we aould again arrive at the concept of 1-randomness. Thus we know that 1-randomness implies Kolmogorov-Loveland randomness which in turn implies computable randomness.

Actually one of the first results proven about Kolmogorov-Loveland randomness was that it is pretty close to 1-randomness in some sense.

Theorem 12.4.7 (An. A. Muchnik [222][2])**.** *(i) Suppose that h is a computable function with*

$$K(A \upharpoonright h(n)) \leqslant h(n) - n,$$

for all n. The A is not Kolmogorov-Loveland random.

(ii) Indeed there are two partial computable non-monotonic such that any sequence satisfying the hypotheses of (i) is covered by one of them.

(iii) Moreover these non-monotonic partial computable betting strategies can be converted into total ones.

Proof. Using iterations of h we can find a computable function g such that

$$K(A \upharpoonright_{g(n)}^{g(n+1)} < g(n+1) - g(n) - 1.$$

A computable g exists sinc ethe hypothese say that $K(A \upharpoonright h(m)) \leqslant h(m) - m$, and if we choose $m >> g(n)$, so as to code $g(n)$, we are done. Let $I_{2e} = [g(2e), g(2e+1))$ and similarly I_{2e+1}. Then Muchnik's argument is as follows. Begin a betting strategy based upon the belief that $K(A \upharpoonright I_{2e})$ settles after $K(A \upharpoonright I_{2e+1}$, for infinitely many e. □

Notice that by Schnorr's Theorem, we can replace Kolmogorov-Loveland randomness by 1-randomness if we delete the word "computable" before h in the statement of Muchnik's Theorem, Theorem **??**.

One key problem which comes when dealing with nonmonotonic randomness is the lack of universal tests. The level to which the lack of universality hurts us can be witnessed by the following simple result.

Theorem 12.4.8 (Merkle et. al. [207])**.** *No partial computable nonmonotonic betting strategy succeeds on all computably enumerable* sets*.*

Proof. Let $b = (s, q)$ be partial computable. We bukld a c.e. set W. We compute $x_n = (r_0, a_0) \dots (r_{n-1}, a_{n-1})$, starting with $x_0 = \lambda$, and setting $r_{n+1} = s(x_n)$, with a_{n+1} being 1 if $q(x_n) \geqslant 1$ and $a_{n+1} = 0$, otherwise. We will enumerate r_{n+1} into W if $a_{n+1} = 1$. Note that b cannot succeed on W. □

Strangely enough, it turns out that *two* nonmonotonic betting stragies are enough to succeed on all c.e. sets.

Theorem 12.4.9 (An. A. Muchnik [222], Merkle et. al. [207])**.** *There exist computable nonmonotonic betting strategies b_0 and b_1 such that for each c.e. set W, at least one of b_0 or b_1 succeed on W.*

Proof. — □

Corollary 12.4.10 (Merkle et. al. [207]). *The Kolmogorov-Loveland nullsets are not closed under finite union.*

12.4.2 van Lambalgen's Theorem revisited

We have seen that looking at splittings yields significant insight into randomness, as witnessed by van Lambalgen's Theorem. Let Z be an infinte co-infinite set of numbers, then we can define the Z-join of A_0 and A_1 via $A_0 \oplus_Z A_1(n) = A_0(|\overline{Z} \cap \{0, \ldots, n-1\}|)$ if $Z(n) = 0$ and $A_0 \oplus_Z A_1(n) = A_1(|Z \cap \{0, \ldots, n-1|)$, for $Z(n) = 1$. That is, we use Z as a guide as to how to merge A_0 and A_1. Clearly van Lambalgen's theorem says that if Z is computable then $A_0 \oplus_Z A_1$ is 1-random iff A_i is 1-A_{1-i}-random. Here is an analog of van Lambalgen's Theorem for Kolmogorov-Loveland randomness.

Theorem 12.4.11 (Merkle et. al. [207]). *Suppose that $A = A_0 \oplus_Z A_1$ for a computable Z. Then A is Kolmogorov-Loveland random iff A_i is Kolmogorov-Loveland random relative to A_{1-i}.*

An important corollary of this result is the following.

Corollary 12.4.12 (Merkle et. al. [207]). *For computable Z with $A_0 \oplus_Z A_1$ Kolmogorov-Loveland random, at least one of A_0 or A_1 is 1-random.*

Part III

Relative Randomness

13
Measures of relative randomness

13.1 Introduction

In this Chapter will will introduce several measures of relative randomness. These measures attempt to quantify when one real is "more random" than another. We will introduce them mainly through the agency of the left-c.e. reals. In Chapter 14 to follow, we will look at other measures of relative randomness which seem more applicable to *random* reals.

13.2 Solovay reducibility

We wish to look at reals, especially left-c.e. reals under notions of relative randomness. Ultimately, we would seek to understand $\leqslant_K$ and $\leqslant_C$ "reducibilities," for instance, where for $E = K$ or C, we have

$$\alpha \leqslant_E \beta \text{ iff } \forall n[E(\alpha \upharpoonright n) \leqslant E(\beta \upharpoonright n) + \mathcal{O}(1)].$$

We view $\leqslant_E$ as an initial segment measure of relative randomness. There are a number of natural reducibilities which imply $\leqslant_E$. One was introduced by Solovay, and some are more recent. In this section we will look at some basic results on Solovay reducibility, and in later sections look at other initial segment measures of relative randomness.

Definition 13.2.1 (Solovay [284])**.** We say that a real α is *Solovay reducible* to β (or β *dominates* α), $\alpha \leqslant_{\mathrm{S}} \beta$ iff there is a constant c and a partial computable function f, so that for all $q \in \mathbb{Q}$, with $q < \beta$,

$$c(\beta - q) > \alpha - f(q).$$

The intuition is that a sequence of rationals converging to β can be used to generate one converging to α at the same rate. The point is that if we have a c.e. sequence $\{q_n : n \in \mathbb{N}\}$ of rationals converging to β then we know that $f(q_n)\downarrow$. Notice that if $r_n \to \alpha$ then for all m there is some k, which can be effectively computed from k, such that $\alpha > r_k > f(q_m)$. (The reals are not rational.) Noticing this yields the following characterization of Solovay reducibility.

Lemma 13.2.2 (Calude, Coles, Hertling, Khoussainov [35])**.** *For left-c.e. reals, $\alpha \leqslant_{\mathrm{S}} \beta$ iff for all c.e. $q_i \to \beta$ there exists a total computable g, and a constant c, such that, for all m,*

$$c(\beta - q_m) > \alpha - r_{g(m)}.$$

The following corollary is immediate:

Corollary 13.2.3. *For any reals α and β, if $\alpha \leqslant_{\mathrm{S}} \beta$, then $\alpha \leqslant_T \beta$.*

Proof. To prove this use $\beta \restriction n \to \beta$, to generate an approximation to α. □

Turing to Kolmogorov complexity, we first show that $\leqslant_{\mathrm{S}}$ is indeed an initial segment measure of relative randomness both for C and K. To do this we will use the following Lemma of Solovay.

Lemma 13.2.4 (Solovay [284])**.** *For all k there is a constant c_k depending on k alone, such that for all n, $|\sigma| = |\tau| = n$ and $|\sigma - \tau| < 2^{k-n}$, then for $E = K$ or C,*

$$E(\sigma) \leqslant E(\tau) + c_k.$$

Proof. Here is the argument for C. We can write a program depending on k which, when given σ, reads the length of σ then computes the ν such that ν has the same length as σ and $|\sigma - \nu| < 2^{k-n}$. Then, given a program for σ, all we need to generate τ is to use the program for the ν's and compute which ν is τ on the list. This is nonuniform, but only needs about $\log k$ many bits since the size of the list depends on k alone.

The argument for K is similar. Suppose that we have a prefix-free M. When we see some ν with $M(\nu) = \sigma$, then we can enumerate a requirement $|\nu| + 2^{k+1}, \tau$ for each of the 2^k τ with $|\sigma - \tau| < 2^{k-n}$. Now apply Kraft-Chaitin. □

Now we use Lemma 13.2.4 to relate Solovay reducibility to complexity.

Theorem 13.2.5 (Solovay)**.** *Suppose that $\alpha \leqslant_{\mathrm{S}} \beta$. Then for $E = K$ or C, $\alpha \leqslant_E \beta$.*

Proof. Suppose that $\alpha \leqslant_{\mathrm{S}} \beta$ via $c < 2^k$, f. Notice that

$$\alpha - f(\beta \restriction (n+1)) < 2^k(\beta - \beta \restriction (n+1)).$$

In particular,

$$\alpha \upharpoonright n - f(\beta \upharpoonright (n+1)) \upharpoonright n < 2^{k-n},$$

and we can apply Lemma 13.2.4. □

Another characterization of $\leqslant_S$ is the following:

Theorem 13.2.6 (Downey, Hirschfeldt, Nies [79]). *For left-c.e. reals, $\alpha \leqslant_S \beta$ iff for all c.e. sequences $\{q_i : i \in \omega\}$ such that $\beta = \sum_i q_i$, there is a computable function $\varepsilon : \omega \mapsto [0,1]$ and a constant c, such that,*

$$\alpha = c(\sum_i \varepsilon(i) q_i).$$

Hence $\alpha \leqslant_S \beta$, iff there exists a c and a left-c.e. real γ such that

$$c\beta = \alpha + \gamma.$$

Proof. (if) One direction is easy. Suppose that c and ε exist. Notice that

$$c(\beta - \sum_{i=1}^{n} q_i) > \alpha - \sum_{i=1}^{n} \varepsilon(i) q_i.$$

Hence $\alpha \leqslant_S \beta$. □

For the other direction, we need the following Lemmas. The first is implicit in Solovay's manuscript, but is first proven in [77].

Lemma 13.2.7. *Let α and β be left-c.e. reals, and let $\alpha_0, \alpha_1, \ldots$ and $\beta_0, \beta_1, \ldots$ be computable increasing sequences of rationals converging to α and β, respectively. Then $\beta \leqslant_S \alpha$ if and only if there are a constant d and a total computable function f such that for all $n \in \omega$,*

$$\beta - \beta_{f(n)} < d(\alpha - \alpha_n).$$

The proof is straightforward and is left as an exercise.

Lemma 13.2.8 (Downey, Hirschfeldt, Nies [79]). *Let $\beta \leqslant_S \alpha$ be left-c.e. reals and let $\alpha_0, \alpha_1, \ldots$ be a computable increasing sequence of rationals converging to α. There is a computable increasing sequence $\widehat{\beta}_0, \widehat{\beta}_1, \ldots$ of rationals converging to β such that for some constant c and all $s \in \omega$,*

$$\widehat{\beta}_s - \widehat{\beta}_{s-1} < c(\alpha_s - \alpha_{s-1}).$$

Proof. Fix a computable increasing sequence $\beta_0, \beta_1, \ldots$ of rationals converging to β, let d and f be as in Lemma 13.2.7, and let $c > d$ be such that $\beta_{f(0)} < c\alpha_0$. We may assume without loss of generality that f is increasing. Define $\widehat{\beta}_0 = \beta_{f(0)}$.

There must be an $s_0 > 0$ for which $\beta_{f(s_0)} - \beta_{f(0)} < d(\alpha_{s_0} - \alpha_0)$, since otherwise we would have

$$\beta - \beta_{f(0)} = \lim_s \beta_{f(s)} - \beta_{f(0)} \geqslant \lim_s d(\alpha_s - \alpha_0) = d(\alpha - \alpha_0),$$

contradicting our choice of d and f. It is now easy to define $\widehat{\beta}_1, \dots, \widehat{\beta}_{s_0}$ so that $\widehat{\beta}_0 < \cdots < \widehat{\beta}_{s_0} = \beta_{f(s_0)}$ and $\widehat{\beta}_s - \widehat{\beta}_{s-1} \leqslant d(\alpha_s - \alpha_{s-1}) < c(\alpha_s - \alpha_{s-1})$ for all $s \leqslant s_0$. For example, if we let μ the minimum value of $d(\alpha_s - \alpha_{s-1})$ for $s \leqslant s_0$ and let t be least such that $\widehat{\beta}_0 + d(\alpha_t - \alpha_0) < \beta_{f(s_0)} - 2^{-t}\mu$ then we can define

$$\widehat{\beta}_{s+1} = \begin{cases} \widehat{\beta}_s + d(\alpha_{s+1} - \alpha_s) & \text{if } s+1 < t \\ \beta_{f(s_0)} - 2^{-(s+1)}\mu & \text{if } t \leqslant s+1 < s_0 \\ \beta_{f(s_0)} & \text{if } s+1 = s_0. \end{cases}$$

We can repeat the procedure in the previous paragraph with s_0 in place of 0 to obtain an $s_1 > s_0$ and $\widehat{\beta}_{s_0+1}, \dots, \widehat{\beta}_{s_1}$ such that $\widehat{\beta}_{s_0} < \cdots < \widehat{\beta}_{s_1} = \beta_{f(s_1)}$ and $\widehat{\beta}_s - \widehat{\beta}_{s-1} < c(\alpha_s - \alpha_{s-1})$ for all $s_0 < s \leqslant s_1$.

Proceeding by recursion in this way, we define a computable increasing sequence $\widehat{\beta}_0, \widehat{\beta}_1, \dots$ of rationals with the desired properties. $\square$

We are now in a position to prove Lemma 13.2.6 for the other direction.

Proof. Suppose that $\beta \leqslant_S \alpha$. Given a computable sequence of rationals $a_0, a_1, \dots$ such that $\alpha = \sum_{n\in\omega} a_n$, let $\alpha_n = \sum_{i\leqslant n} a_i$ and apply Lemma 13.2.8 to obtain c and $\widehat{\beta}_0, \widehat{\beta}_1, \dots$ as in that lemma. Define $\varepsilon_n = (\widehat{\beta}_n - \widehat{\beta}_{n-1})a_n^{-1}$. Now $\sum_{n\in\omega} \varepsilon_n a_n = \sum_{n\in\omega} \widehat{\beta}_n - \widehat{\beta}_{n-1} = \beta$, and for all $n \in \omega$,

$$\varepsilon_n = (\widehat{\beta}_n - \widehat{\beta}_{n-1})a_n^{-1} = (\widehat{\beta}_n - \widehat{\beta}_{n-1})(\alpha_n - \alpha_{n-1})^{-1} < c.$$

$\square$

13.3 The Kučera-Slaman Theorem

Solovay introduced Solovay reducibility to define a class of left-c.e. reals he called "Ω-like", which he observed had many of the properties of Ω. First Solovay noted that Ω is Solovay complete.

Lemma 13.3.1 (Solovay [284])**.** *Suppose that α is a left-c.e. real. Then $\alpha \leqslant_S \Omega$.*

Proof. If α is a left-c.e. real, then $\alpha = \mu(\mathrm{dom}(M))$ for some prefix-free machine M. Then M is coded in the universal prefix-free machine U by some coding constant e. That is, $M(\sigma) = U(1^e 0\sigma)$. Hence if we go to a stage where we know Ω to within $2^{-(e+n+1)}$, then we will know α to within 2^{-n}. $\square$

The reader should note that "Solovay completeness" means that this means for all left-c.e. *reals* (not just c.e. *sets*) α, $\alpha \leqslant_S \Omega$. Furthermore if $\Omega \leqslant_S \alpha$, for any (not necessarily left-c.e.) real α then α must be random. This follows by the Chaitin definition of randomness and by Theorem 13.2.5.

Definition 13.3.2 (Solovay [284])**.** We say that a left-c.e. real α is Ω-like if $\Omega \leqslant_S \alpha$.

Solovay proved that Ω-like reals possessed many of the properties that Ω possessed. He remarks:

"It seems strange that we will be able to prove so much about the behavior of $K(\Omega \upharpoonright n)$ when, a priori, the definition of Ω is thoroughly model dependent. What our discussion has shown is that our results hold for a class of reals (that include the value of the universal measures of ...) and that the function $K(\Omega \upharpoonright n)$ is model independent to within $\mathcal{O}(1)$."

The following two results establish that using Ω-like numbers in place of Ω makes no difference in the same way that using any set of the same m-degrees as the halting problem gives a version of the halting problem. Thus, it turns out that Solovay's observations are not so strange after all!

The first result is a straightforward application of Kraft-Chaitin to show that any Ω-like number is a halting probability.

Theorem 13.3.3 (Calude, Hertling, Khoussainov, Wang [36])**.** *Suppose that α is a left-c.e. real and that $\Omega \leqslant_S \alpha$. Then α is a halting probability. That is, there is a universal machine $\widehat{U}$ such that $\mu(dom(\widehat{U})) = \alpha$.*

Proof. Thus suppose that for all $q < \alpha$, $f(q) \downarrow < \Omega$ and $2^c(\alpha - q) > \Omega - f(q)$. We will build our machine M in stages. Take an enumeration $\Omega = \lim_s \Omega_s = \sum_{U(\sigma)\downarrow[s]} 2^{-|\sigma|}$. By Lemma 13.2.8, there is an enumeration of $\alpha = \lim_s \alpha_s$, such that for all t and s, there is a t' such that $2^c(\alpha_{t'} - \alpha_s) > \Omega_t - \Omega_s$. Thus we can speed up such an enumeration to ask that for all s, $2^c(\alpha_{s+1} - \alpha_s) > \Omega_{s+1} - \Omega_s$.

Now the proof is easy. At stage s, suppose that $U(\sigma) \downarrow$, so that σ enters the domain of U, and hence $\Omega_{s+1} - \Omega_s \geqslant 2^{-|\sigma|}$. We can assume that exactly on string enters the domain of U. Then $\alpha_{s+1} - \alpha_s \geqslant 2^{-(|\sigma|+c)}$. Hence we can issue a Kraft-Chaitin request: $\langle |\sigma| + c, U(\sigma)\rangle$. Then by Kraft-Chaitin, there is a machine M honoring all such requests, and such a machine M is universal. $\square$

The complete characterization of the random left-c.e. reals is given by the following beautiful result of Kučera and Slaman. Kučera and Slaman showed that domination provides a precise characterization of randomness.

Theorem 13.3.4 (Kučera and Slaman [161])**.** *Suppose that α is random and left-c.e.. Then for all left-c.e. reals β, $\beta \leqslant_S \alpha$.*

Proof. Suppose that α is random and β is a left-c.e. real. We need to show that $\beta \leqslant_S \alpha$. We enumerate a Martin-Löf test $F_n : n \in \omega$ in stages. Let $\alpha_s \to \alpha$ and $\beta_s \to \beta$ computably and monotonically. We assume that $\beta_s < \beta_{s+1}$. At stage s if $\alpha_s \in F_n^s$, do nothing, else put $(\alpha_s, \alpha_s + 2^{-n}(\beta_{s+1} - \beta_{t_s}))$ into F_n^{s+1}, where t_s denotes the last stage we put something into F_n. One verifies that $\mu(F_n) < 2^{-n}$. Thus the F_n define a Martin-Löf test. As α is random, there is a n such that for all $m \geqslant n$, $\alpha \notin F_m$. This shows that $\beta \leqslant_{\mathrm{S}} \alpha$ with constant 2^n. □

Corollary 13.3.5. *For left-c.e. reals α the following are equivalent:*

(i) α is 1-random.

(ii) For all left-c.e. reals, β, for all n, $K(\beta \upharpoonright n) \leqslant K(\alpha \upharpoonright n) + \mathcal{O}(1)$.

(iii) For all left-c.e. reals, β, for all n, $C(\beta \upharpoonright n) \leqslant C(\alpha \upharpoonright n) + \mathcal{O}(1)$.

(iv) For any version of Ω, for all n, $C(\Omega \upharpoonright n) \leqslant C(\alpha \upharpoonright n) + \mathcal{O}(1)$ and $K(\Omega \upharpoonright n) = K(\alpha \upharpoonright n) + \mathcal{O}(1)$.

(v) For all left-c.e. reals β, $\beta \leqslant_{\mathrm{S}} \alpha$.

(vi) α is the halting probability of some universal machine.

So we see that Solovay reducibility is good with respect to randomness. Notice that (iv) and (v) above say something very strong. Consider a random left-c.e. real x. Then for e.g. K, we know that

$$K(x \upharpoonright n) \geqslant n + \mathcal{O}(1).$$

However we also know that

$$K(\sigma) \leqslant |\sigma| + 2\log(|\sigma|) + \mathcal{O}(1).$$

It would seem that there could be random y and random x where for infinitely many n, $x \upharpoonright n$ had K-complexity $n+\log n$, yet y had K-complexity n. Why not? After all the the complexity only needs to be above n to "qualify" as random, and it certainly can be as large as $n + \log n$.

However, Kučera and Slaman's theorem says that this is not *so. All random left-c.e. reals have "high" complexity (like $n+\log n$) and low complexity (like n) at the same n's!*

The above also says that there should be a plain complexity characterization of randomness for left-c.e. reals. It was a longstanding question whether there was a plain complexity characterization of 1-randomness. As we have seen, Miller and Yu have found such a characterization. We saw this in Theorem 9.8.2.

13.4 The structure of the Solovay degrees

Despite the many attractive features of the Solovay degrees of left-c.e. reals, their structure is largely unknown. Some results come for free by the fact that $\leqslant_{\mathrm{S}}$ implies $\leqslant_T$. (Theorem 13.2.3) Thus there are minimal pairs of $\leqslant_{\mathrm{S}}$ degrees, etc.

Actually, the question that originally inspired the research of the present book was whether the Solovay degrees of left-c.e. reals form a dense partial ordering. We will soon see that the answer is yes. However, before we turn to that, we classify the structure as a distributive upper-semilattice.

Theorem 13.4.1 (Downey, Hirschfeldt and Nies [79]). *The Solovay degrees of left-c.e. reals forms a distributive upper semilattice, where the operation of join is induced by +, arithmetic addition (or multiplication) (namely $[x] \vee [y] \equiv_{\mathrm{S}} [x+y]$.)*

Theorem 13.4.2 (Downey, Hirschfeldt and Nies [79]). *If $[\Omega] = \mathbf{a} \vee \mathbf{b}$ then either $[\Omega] = \mathbf{a}$ or $[\Omega] = \mathbf{b}$.*

Proof. We will be applying Theorem 13.2.6, but for convenience will write ε_i instead of $\varepsilon(i)$.

To see that the join in the Solovay degrees is given by addition, we again apply Lemma 13.2.6. Certainly, for any left-c.e. reals β_0 and β_1 we have $\beta_i \leqslant_{\mathrm{S}} \beta_0 + \beta_1$ for $i = 0, 1$, and hence $[\beta_0 + \beta_1] \geqslant_{\mathrm{S}} [\beta_0], [\beta_1]$. Conversely, suppose that $\beta_0, \beta_1 \leqslant \alpha$. Let $a_0, a_1, \dots$ be a computable sequence of rationals such that $\alpha = \sum_{n \in \mathbb{N}} a_n$. For each $i = 0, 1$ there is a constant c_i and a computable sequence of rationals $\varepsilon_0^i, \varepsilon_1^i, \dots < c_i$ such that $\beta_i = \sum_{n \in \mathbb{N}} \varepsilon_n^i a_n$. Thus $\beta_0 + \beta_1 = \sum_{n \in \mathbb{N}} (\varepsilon_n^0 + \varepsilon_n^1) a_n$. Since each $\varepsilon_n^0 + \varepsilon_n^1$ is less than $c_0 + c_1$, a final application of Lemma 13.2.6 shows that $\beta_0 + \beta_1 \leqslant_{\mathrm{S}} \alpha$. Multiplication is similar.

To show that the structure is distributive, suppose that $\beta \leqslant_{\mathrm{S}} \alpha_0 + \alpha_1$. Let $a_0^0, a_1^0, \dots$ and $a_0^1, a_1^1, \dots$ be computable sequences of rationals such that $\alpha_i = \sum_{n \in \mathbb{N}} a_n^i$ for $i = 0, 1$. By Lemma 13.2.6, there are a constant c and a computable sequence of rationals $\varepsilon_0, \varepsilon_1, \dots < c$ such that $\beta = \sum_{n \in \mathbb{N}} \varepsilon_n (a_n^0 + a_n^1)$. Let $\beta_i = \sum_{n \in \mathbb{N}} \varepsilon_n a_n^i$. Then $\beta = \beta_0 + \beta_1$ and, again by Lemma 13.2.6, $\beta_i \leqslant_{\mathrm{S}} \alpha_i$ for $i = 0, 1$. This establishes distributivity.

To see that the join in the Solovay degrees is given by addition, we again apply Lemma 13.2.6. Certainly, for any left-c.e. reals β_0 and β_1 we have $\beta_i \leqslant_{\mathrm{S}} \beta_0 + \beta_1$ for $i = 0, 1$, and hence $[\beta_0 + \beta_1] \geqslant_{\mathrm{S}} [\beta_0], [\beta_1]$. Conversely, suppose that $\beta_0, \beta_1 \leqslant \alpha$. Let $a_0, a_1, \dots$ be a computable sequence of rationals such that $\alpha = \sum_{n \in \mathbb{N}} a_n$. For each $i = 0, 1$ there is a constant c_i and a computable sequence of rationals $\varepsilon_0^i, \varepsilon_1^i, \dots < c_i$ such that $\beta_i = \sum_{n \in \mathbb{N}} \varepsilon_n^i a_n$. Thus $\beta_0 + \beta_1 = \sum_{n \in \mathbb{N}} (\varepsilon_n^0 + \varepsilon_n^1) a_n$. Since each $\varepsilon_n^0 + \varepsilon_n^1$ is less than $c_0 + c_1$, a final application of Lemma 13.2.6 shows that $\beta_0 + \beta_1 \leqslant_{\mathrm{S}} \alpha$. Multiplication is similar. $\square$

We finish this section with the proof that the structure is *not* a lattice.

Theorem 13.4.3. *There exist c.e.* sets *A and B such that the Solovay degrees of A and B have no infimum in the Solovay degrees.*

Proof. The proof is a straightforward adaptation of the corresponding proof that the c.e. weak truth table degrees are not a lattice; and we use the method invented by Jockusch [131]. We will build computably enumerable *sets* A and B and auxiliary c.e. sets X_e, to meet the requirements

$$R_e : V_e \leqslant_{\mathrm{S}} A \text{ via } 2^e, \varphi \wedge V_e \leqslant_{\mathrm{S}} B \text{ via } 2^e, \psi_e$$

$$\text{implies } X_e \leqslant_{\mathrm{S}}, A, B \wedge \forall i(R_{e,i}), \text{ where,}$$

$$R_{e,i} : X_e \not\leqslant_{\mathrm{S}} V_e \text{ via } 2^i, \xi_i.$$

Here, of course, V_e denotes the e-th left-c.e. real, and φ, ψ, ξ are partial computable function meant to be the relevant Solovay reductions. The argument is a standard finite injury one, and it will suffice to describe the action for a single $R_{e,i}$.

For the sake of $R_{e,i}$ we will pick some location n, which will be a large fresh number. This number is targeted for X_e. We then await a stage s where $2^i V_e$ computes X_e to within 2^{-n}, via $\xi_i[s]$; and both A_s and B_s compute $V_e[s]$ to within $2^{-(n+e+i+1)}$ via, respectively $\varphi_e[s]$ and $\psi_e[s]$.

Then the action is to initialize lower priority requirements, and implement the following two steps.

First we put add 2^{-n} to A_s, but *restrain* $B_t = B_s \upharpoonright n+e+i+1$ until a stage t occurs, where the φ_e computations claim to compute from A_t, $V_e \upharpoonright n+i+1[t] = V_e \upharpoonright n+i+1$.

Second, we then would add 2^{-n} to B_t, and add 2^{-n} into $X_e[t+1]$.

Note that $X_e \leqslant_{\mathrm{S}} A, B$ with constant 1, *if* R_e's hypotheses are satisfied. Second we see that $X_e \not\leqslant_{\mathrm{S}} V_e$ by the above. □

Note We remark that we could have used the hypothesis that $V_e \leqslant_T A, B$ in the above, and made $X_e \not\leqslant_T V_e$, whilst keeping $X_e \leqslant_{\mathrm{S}} A, B$ using essentially the *same* argument.

MORE HERE

Theorem 13.4.4 (Downey, Hirschfeldt and Nies [79])**.** *The upper semilattice of Solovay degrees of left-c.e. reals is dense.*

The proof of the Density Theorem splits into two cases. We treat these cases as separate theorems. This nonuniformity will later be proven to be necessary. The first proof establishes that each incomplete Solovay degrees splits over all lesser ones. It uses a new idea which will later be significantly generalized.

CHANGE HERE

Theorem 13.4.5. *Let $\gamma <_{\mathrm{S}} \alpha <_{\mathrm{S}} \Omega$ be left-c.e. reals. There are left-c.e. reals β^0 and β^1 such that $\gamma <_{\mathrm{S}} \beta^i <_{\mathrm{S}} \alpha$ for $i = 0, 1$ and $\beta^0 + \beta^1 = \alpha$.*

DELETED PROOF

We now show that the Solovay degrees are upwards dense, which together with the previous result implies that they are dense. This proof is a relatively straightforward finite injury argument based upon the fact that we don't need to make $\beta \leqslant_{\mathrm{S}} \Omega$.

Theorem 13.4.6. *Let $\gamma <_{\mathrm{S}} \Omega$ be a left-c.e. real. There is a left-c.e. real β such that $\gamma <_{\mathrm{S}} \beta <_{\mathrm{S}} \Omega$.*

DELETED AS ABOVE

Combining Theorems 13.4.5 and 13.4.6, we have the proven our density theorem.

Downey, Hirschfeldt and LaForte [77] later used similar techniques to other reducibilities to prove similar density theorems. This is no surprise as the following result shows.

Theorem 13.4.7. *Suppose that $\leqslant$ is any Σ_3^0 reducibility such that, on the left-c.e. reals, $+$ is a join, if A is computable, then $A \leqslant 1^\omega$ and such that $[\Omega]_\leqslant$ is the the top degree. Then the $\leqslant$ degrees of left-c.e. reals are dense.*

CHANGES

The hypothesis that $\alpha <_{\mathrm{S}} \Omega$ in the statement of Theorem 13.4.5 is necessary. This fact will follow easily from a stronger result which shows that, despite the upwards density of the Solovay degrees, there is a sense in which the complete Solovay degree is very much above all other Solovay degrees. We begin with a lemma giving a sufficient condition for domination.

Lemma 13.4.8. *Let α and β be left-c.e. reals and let $\alpha_0, \alpha_1, \dots$ and $\beta_0, \beta_1, \dots$ be computable increasing sequences of rationals converging to α and β, respectively. Let f be an increasing total computable function and let $k > 0$ be a natural number. If there are infinitely many $s \in \mathbb{N}$ such that $k(\alpha - \alpha_s) > \beta - \beta_{f(s)}$, but only finitely many $s \in \mathbb{N}$ such that $k(\alpha_t - \alpha_s) > \beta_{f(t)} - \beta_{f(s)}$ for all $t > s$, then $\beta \leqslant_{\mathrm{S}} \alpha$.*

Proof. By taking $\beta_{f(0)}, \beta_{f(1)}, \dots$ instead of $\beta_0, \beta_1, \dots$ as an approximating sequence for β, we may assume that f is the identity.

By hypothesis, there is an $r \in \mathbb{N}$ such that for all $s > r$ there is a $t > s$ with $k(\alpha_t - \alpha_s) \leqslant \beta_t - \beta_s$. Furthermore, there is an $s_0 > r$ such that $k(\alpha - \alpha_{s_0}) > \beta - \beta_{s_0}$. Given s_i, let s_{i+1} be the least number greater than s_i such that $k(\alpha_{s_{i+1}} - \alpha_{s_i}) \leqslant \beta_{s_{i+1}} - \beta_{s_i}$.

Assuming by induction that $k(\alpha - \alpha_{s_i}) > \beta - \beta_{s_i}$, we have

$$k(\alpha - \alpha_{s_{i+1}}) = k(\alpha - \alpha_{s_i}) - k(\alpha_{s_{i+1}} - \alpha_{s_i}) > \beta - \beta_{s_i} - (\beta_{s_{i+1}} - \beta_{s_i}) = \beta - \beta_{s_{i+1}}.$$

Thus $s_0 < s_1 < \cdots$ is a computable sequence such that $k(\alpha - \alpha_{s_i}) > \beta - \beta_{s_i}$ for all $i \in \mathbb{N}$.

Now define the computable function g by letting $g(n)$ be the least s_i that is greater than or equal to n. Then $\beta - \beta_{g(n)} < k(\alpha - \alpha_{g(n)}) \leqslant k(\alpha - \alpha_n)$ for all $n \in \mathbb{N}$, and hence $\beta \leqslant_{\mathrm{S}} \alpha$. $\square$

Theorem 13.4.9. *Let α and β be left-c.e. reals and let $\alpha_0, \alpha_1, \dots$ and $\beta_0, \beta_1, \dots$ be computable increasing sequences of rationals converging to α and β, respectively. Let f be an increasing total computable function and let $k > 0$ be a natural number. If β is random and there are infinitely many $s \in \mathbb{N}$ such that $k(\alpha - \alpha_s) > \beta - \beta_{f(s)}$ then α is random.*

Proof. As in Lemma 13.4.8, we may assume that f is the identity. If α is rational then we can replace it with a nonrational computable real α' such that $\alpha' - \alpha'_s \geqslant \alpha - \alpha_s$ for all $s \in \mathbb{N}$, so we may assume that α is not rational.

We assume that α is nonrandom and there are infinitely many $s \in \mathbb{N}$ such that $k(\alpha - \alpha_s) > \beta - \beta_s$, and show that β is nonrandom. The idea is to take a Solovay test $A = \{I_i : i \in \mathbb{N}\}$ such that $\alpha \in I_i$ for infinitely many $i \in \mathbb{N}$ and use it to build a Solovay test $B = \{J_i : i \in \mathbb{N}\}$ such that $\beta \in J_i$ for infinitely many $i \in \mathbb{N}$.

Let

$$U = \{s \in \mathbb{N} \mid k(\alpha - \alpha_s) > \beta - \beta_s\}.$$

Except in the trivial case in which $\beta \equiv_{\mathrm{S}} \alpha$, Lemma 13.12.1 guarantees that U is Δ^0_2. Thus a first attempt at building B could be to run the following procedure for all $i \in \mathbb{N}$ in parallel. Look for the least t such that there is an $s < t$ with $s \in U[t]$ and $\alpha_s \in I_i$. If there is more than one number s with this property then choose the least among such numbers. Begin to add the intervals

$$[\beta_s, \beta_s + k(\alpha_{s+1} - \alpha_s)], [\beta_s + k(\alpha_{s+1} - \alpha_s), \beta_s + k(\alpha_{s+2} - \alpha_s)], \ldots \quad (13.1)$$

to B, continuing to do so as long as s remains in U and the approximation of α remains in I_i. If the approximation of α leaves I_i then end the procedure. If s leaves U, say at stage u, then repeat the procedure (only considering $t \geqslant u$, of course).

If $\alpha \in I_i$ then the variable s in the above procedure eventually assumes a value in U. For this value, $k(\alpha - \alpha_s) > \beta - \beta_s$, from which it follows that $k(\alpha_u - \alpha_s) > \beta - \beta_s$ for some $u > s$, and hence that $\beta \in [\beta_s, \beta_s + k(\alpha_u - \alpha_s)]$. So β must be in one of the intervals (13.1) added to B by the above procedure.

Since α is in infinitely many of the I_i, running the above procedure for all $i \in \mathbb{N}$ guarantees that β is in infinitely many of the intervals in B. The problem is that we also need the sum of the lengths of the intervals in B to be finite, and the above procedure gives no control over this sum, since it could easily be the case that we start working with some s, see it leave U at some stage t (at which point we have already added to B intervals whose lengths add up to $\alpha_{t-1} - \alpha_s$), and then find that the next s with which we have to work is much smaller than t. Since this could happen many times for each $i \in \mathbb{N}$, we would have no bound on the sum of the lengths of the intervals in B.

This problem would be solved if we had an infinite computable subset T of U. For each I_i, we could look for an $s \in T$ such that $\alpha_s \in I_i$, and then begin to add the intervals (13.1) to B, continuing to do so as long as the approximation of α remained in I_i. (Of course, in this easy setting, we could also simply add the single interval $[\beta_s, \beta_s + k\,|I_i|]$ to B.) It is not hard to check that this would guarantee that if $\alpha \in I_i$ then β is in one of the intervals added to B, while also ensuring that the sum of the lengths of these intervals is less than or equal to $k\,|I_i|$. Following this procedure for all $i \in \mathbb{N}$ would give us the desired Solovay test B. Unless $\beta \leqslant_S \alpha$, however, there is no infinite computable $T \subseteq U$, so we use Lemma 13.4.8 to obtain the next best thing.

Let

$$S = \{s \in \mathbb{N} \mid \forall t > s(k(\alpha_t - \alpha_s) > \beta_t - \beta_s)\}.$$

If $\beta \leqslant_S \alpha$ then β is nonrandom, so, by Lemma 13.4.8, we may assume that S is infinite. Note that $k(\alpha - \alpha_s) \geqslant \beta - \beta_s$ for all $s \in S$. In fact, we may assume that $k(\alpha - \alpha_s) > \beta - \beta_s$ for all $s \in S$, since if $k(\alpha - \alpha_s) = \beta - \beta_s$ then $k\alpha$ and β differ by a rational amount, and hence β is nonrandom.

The set S is co-c.e. by definition, but it has an additional useful property. Let

$$S[t] = \{s \in \mathbb{N} \mid \forall u \in (s,t](k(\alpha_u - \alpha_s) > \beta_u - \beta_s)\}.$$

If $s \in S[t-1] - S[t]$ then no $u \in (s,t)$ is in S, since for any such u we have

$$k(\alpha_t - \alpha_u) = k(\alpha_t - \alpha_s) - k(\alpha_u - \alpha_s) \leqslant \beta_t - \beta_s - (\beta_u - \beta_s) = \beta_t - \beta_u.$$

In other words, if s leaves S at stage t then so do all numbers in (s,t).

To construct B, we run the following procedure P_i for all $i \in \mathbb{N}$ in parallel. Note that B is a multiset, so we are allowed to add more than one copy of a given interval to B.

1. Look for an $s \in \mathbb{N}$ such that $\alpha_s \in I_i$.
2. Let $t = s+1$. If $\alpha_t \notin I_i$ then terminate the procedure.
3. If $s \notin S[t]$ then let $s = t$ and go to step 2. Otherwise, add the interval
$$[\beta_s + k(\alpha_{t-1} - \alpha_s), \beta_s + k(\alpha_t - \alpha_s)]$$
to B, increase t by one, and repeat step 3.

This concludes the construction of B. We now show that the sum of the lengths of the intervals in B is finite and that β is in infinitely many of the intervals in B.

For each $i \in \mathbb{N}$, let B_i be the set of intervals added to B by P_i and let l_i be the sum of the lengths of the intervals in B_i. If P_i never leaves step 1 then $B_i = \emptyset$. If P_i eventually terminates then $l_i \leqslant k(\alpha_t - \alpha_s)$ for some $s,t \in \mathbb{N}$ such that $\alpha_s, \alpha_t \in I_i$, and hence $l_i \leqslant k\,|I_i|$. If P_i reaches step 3 and never terminates then $\alpha \in I_i$ and $l_i \leqslant k(\alpha - \alpha_s)$ for some $s \in \mathbb{N}$ such that $\alpha_s \in I_i$, and hence again $l_i \leqslant k\,|I_i|$. Thus the sum of the lengths of the intervals in B is less than or equal to $k\sum_{i\in\mathbb{N}}|I_i| < \infty$.

To show that β is in infinitely many of the intervals in B, it is enough to show that, for each $i \in \mathbb{N}$, if $\alpha \in I_i$ then β is in one of the intervals in B_i.

Fix $i \in \mathbb{N}$ such that $\alpha \in I_i$. Since α is not rational, $\alpha_u \in I_i$ for all sufficiently large $u \in \mathbb{N}$, so P_i must eventually reach step 3. By the properties of S discussed above, the variable s in the procedure P_i eventually assumes a value in S. For this value, $k(\alpha - \alpha_s) > \beta - \beta_s$, from which it follows that $k(\alpha_u - \alpha_s) > \beta - \beta_s$ for some $u > s$, and hence that $\beta \in [\beta_s, \beta_s + k(\alpha_u - \alpha_s)]$. So β must be in one of the intervals (13.1), all of which are in B_i. □

Corollary 13.4.10. *If α^0 and α^1 are left-c.e. reals such that $\alpha^0 + \alpha^1$ is random then at least one of α^0 and α^1 is random.*

Proof. Let $\beta = \alpha^0 + \alpha^1$. For each $s \in \mathbb{N}$, either $3(\alpha^0 - \alpha^0_s) > \beta - \beta_s$ or $3(\alpha^1 - \alpha^1_s) > \beta - \beta_s$, so for some $i < 2$ there are infinitely many $s \in \mathbb{N}$ such that $3(\alpha^i - \alpha^i_s) > \beta - \beta_s$. By Theorem 13.4.9, α^i is random. □

Combining Theorem 13.4.5 and Corollary 13.4.10, we have the following results, the second of which also depends on Theorem 13.3.4.

Theorem 13.4.11 (Downey, Hirschfeldt and Nies [79])**.** *A left-c.e. real γ is random if and only if it cannot be written as as $\alpha + \beta$ for left-c.e. reals α and β.*

Theorem 13.4.12. *Let* **d** *be a Solovay degree of left-c.e. reals. The following are equivalent:*

1. **d** *is incomplete.*
2. **d** *splits.*
3. **d** *splits over any lesser Solovay degree.*

We point out that Theorem 13.4.1 only applies to left-c.e. reals. Consider, for instance, if $\Omega = .a_0a_1....$ then if we put $\alpha = .a_0 0 a_2 0 a_4 0 \dots$ and $\beta = .0 a_1 0 a_3 0 \dots$, then clearly neither α nor β can be random yet $\alpha + \beta = \Omega$, but they are not left-c.e..

Before we leave the Solovay degrees of left-c.e. reals, we note that the structure must be very complicated.

Theorem 13.4.13 (Downey, Hirschfeldt, LaForte [78])**.** *The Solovay degrees of left-c.e. reals have an undecidable first order theory.*

The proof of theorem 13.4.13 uses Nies's method of interpreting effectively dense boolean algebras, together with a technical construction of a certain class of (strongly) left-c.e. reals. Calude and Nies [37] have proven that the random reals are all *wtt*-complete. Very little else is known about the Solovay degrees of left-c.e. reals.

13.5 sw-reducibility

We have seen that $\leqslant_{\mathrm{S}}$ on reals is a *measure of relative randomness* since it satisfies the *Solovay property*:

$$\text{If } \beta \leqslant \alpha \text{ then } \exists c\ (\forall n\ (E(\beta \upharpoonright n) \leqslant E(\alpha \upharpoonright n) + c)),$$

for $E \in \{C, K\}$. Actually, Solovay reducibility is a natural example of a reducibility that has this property, but as we see in the present chapter, and in some later ones, it also has a number of problems. They include the following.

(i) Restricted to left-c.e. reals.

(ii) Too fine.

(iii) Too uniform.

To see that (i) holds, we note the following.

Lemma 13.5.1. *There is a d.c.e. real β that is not S-above any left-c.e. real (including the computable reals).*

Proof. We build a real β, making sure that it is noncomputable, and trying to defeat all φ_e, c_e potential Solovay reductions. We are slowly making $\beta_s > \beta_t$ for $s > t$. Additionally, we are building a computable nonrational real $\alpha = \lim_s \alpha_s$. At some stage s, we get that $\varphi_{e,s}(\beta_t)\downarrow$, and

$$c_e(\beta_s - \beta_t) > \alpha_s - \varphi_{e,s}(\beta_t).$$

Then at stage $s+1$, we simply make β_{s+1} sufficiently close to β_t to make

$$c_e(\beta_{s+1} - \beta_t) < \alpha_s - \varphi_{e,s}(\beta_t).$$

□

Thus Solovay reducibility fails to be useful for classifying relative complexity as soon as we leave the left-c.e. reals. This is not to say that *when* $\alpha \leqslant_{\mathrm{S}} \beta$ then $\alpha \leqslant_K \beta$, it is just that there will many circumstances where there is no hope of applying anything close to Solovay reducibility.

Even on the left-c.e. reals Solovay reducibility fails badly to encompass relative complexity. In [77], Downey, Hirschfeldt, and LaForte introduced another measure of relative complexity called *sw-reducibility* (strong weak truth table reducibility):

Definition 13.5.2. $\beta \leqslant_{\mathrm{sw}} \alpha$ if there is a functional Γ such that $\Gamma^\alpha = \beta$ and the use of Γ is bounded by $x + c$ for some c.

It is easy to see that by Lemma 13.2.4, for any (not necessarily left-c.e.) reals $\alpha \leqslant_{\mathrm{sw}} \beta$, for all n, and $E = K$ or $E = C$,

$$E(\alpha \upharpoonright n) \leqslant E(\beta \upharpoonright n) + \mathcal{O}(1).$$

Notice that *sw*-reducibility is an example of a partial continuous transformation Γ acting on $2^{<\omega}$, in that $\sigma \preccurlyeq \tau$ implies $\Gamma(\sigma) \preccurlyeq \Gamma(\tau)$. Such transformations have been widely classically studied, and of relevance to us here are Hölder transformations $f : 2^\omega \mapsto 2^\omega$, that is

$$\forall \alpha, \beta (d(f(\alpha), f(\beta)) \leqslant c d(\alpha, \beta)^r,$$

where $c, r > 0$ and d is the Hausdorff metric. These are in turn induced by maps $\widehat{f}$ on $2^{<\infty}$

$$\liminf_{n\to\infty} \frac{\widehat{f}(\alpha \upharpoonright n)}{n} \leqslant r.$$

These will re-examined in Chapter 17, where we look at Hausdorff dimension. The case we are interested in is where $c = r = 1$, which are called Lipshitz transformations , and in the partial computable case with the use of the corresponding *sw*-reduction is the identity. Such primitive *sw*-reductions have also turned up in differential geometry as studied by Soare, Nabutovsky and Weinberger as described in [281].

Solovay reducibility says that sequences converging to one real can be effectively converted to ones converging to the other. *sw*-reducibility says that the *bits* of β can essentially be used to *efficiently* generate the *bits* of α in the sense that $n + c$ bits of β generate n bits of α.

This is an apparently essential difference. But Solovay reducibility and *sw*-reducibility do agree on at least one important class of sets.

Theorem 13.5.3 (Downey, Hirschfeldt, LaForte [77]). *Suppose that A is a left-c.e. real and B is a c.e.* set. *Then $A \leqslant_{\text{S}} B$ iff $A \leqslant_{\text{sw}} B$.*

Proof. Let A and B be as above and suppose that $\Gamma^B = A$ with use $x + c$. We may assume that we have the approximations of A and B sped up so that every stage is expansionary. That is, for all stages s and all $z \leqslant s$, we have $\Gamma_s^{B_s}(z) = A_s(z)$. We may also assume that if z enters A at stage s then $s \geqslant z$. Now if z enters A at stage s then some number less than or equal to $z + c$ must enter B at stage s. Since B is c.e., this means that $B_s - B_{s-1} \geqslant 2^{-(z+c)}$. But z entering A corresponds to a change of at most 2^{-z} in the value of α, so $B_s - B_{s-1} \geqslant 2^{-c}(A_s - A_{s-1})$. Thus for all s we have $A - A_s \leqslant 2^c(B - B_s)$, and hence $A \leqslant_{\text{S}} B$. $\square$

Theorem 13.5.4 (Downey, Hirschfeldt, LaForte [77]). *If A is strongly c.e. and B is c.e. then $A \leqslant_{\text{S}} B$ implies $A \leqslant_{\text{sw}} B$.*

Proof. Let A and B satisfy the hypotheses of the theorem. Note that, since A is strongly c.e., for all k and s we have $A \upharpoonright k = A_s \upharpoonright k$ if and only if $A - A_s \leqslant 2^{-(k+1)}$. Let f and d be as in Lemma 13.2.7 and let k be such that $d \leqslant 2^{k-2}$. To decide whether $x \in A$ using the first $x+k$ bits of B, find the least stage s such that $B_s \upharpoonright x + k = B \upharpoonright x + k$. We claim that $x \in A$ if and only if $x \in A_{f(s)}$. To verify this claim, first note that $B - B_s < 2^{-(x+k)}$, since otherwise B_s would have to change on one of its first $x + k$ places after stage s. Thus $A - A_{f(s)} \leqslant 2^{k-2}2^{-(x+k)} = 2^{-(x+2)}$, and hence, as noted above, A has stopped changing on the numbers $0, \dots, x$ by stage $f(x)$. $\square$

Thus *sw* and *S*-reducibilities agree on c.e. *sets*.

Corollary 13.5.5 (Downey, Hirschfeldt, LaForte [77]). *For c.e.* sets *A and B, $A \leqslant_{\text{S}} B$ iff $A \leqslant_{\text{sw}} B$.*

sw-reducibility also shows that if we want a general reducibility capturing relative randomness, then Solovay reducibility is too fine.

Theorem 13.5.6 (Downey, Hirschfeldt, LaForte [77]). *There exist left-c.e. reals $\alpha \leqslant_{\text{sw}} \beta$ such that $\alpha \nleqslant_{\text{S}} \beta$. Moreover, α can be chosen to be strongly c.e..*

Proof. We must build α and β so that $\alpha \leqslant_{\text{sw}} \beta$ and α is strongly c.e., while satisfying the following requirements for each $e, c \in \omega$.

$$\mathcal{R}_{e,c} : \exists q \in \mathbb{Q}(c(\beta - q) \ngeqslant \alpha - \Phi_e(q)),$$

where Φ_e is the eth partial computable function. We do this with a straightforward finite injury argument.

We discuss the strategy for a single requirement $\mathcal{R}_{e,c}$. Let k be such that $c \leqslant 2^k$. We must make the difference between β and some rational q quite small while making the difference between α and $\Phi_e(q)$ relatively large. At a stage t we pick a new big number d. For the sake of $\mathcal{R}_{e,c}$, we will control the first $d + k + 3$ places of (the binary expansion of) β_s and α_s for $s \geqslant t$. We set $\beta_t(x) = 1$ for all x with $d \leqslant x \leqslant d + k + 2$, while at the same time keeping $\alpha_s(x) = 0$ for all such x. We let $q = \beta_t$. Note that, since we are restraining the first $d + k + 3$ places of β_s, we know that, unless this restraint is lifted, β_s can only change on positions greater than or equal to $2^{-(d+k+3)} = 2^{-(d+3)}$.

We now need do nothing until we come to a stage $s \geqslant t$ such that $\Phi_{e,s}(q)\downarrow$ *and* $0 < \alpha_s - \Phi_{e,s}(q) \leqslant 2^{-(d+3)}$. Our action then is the following. First we add $2^{-(d+k+2)}$ to β_s. Then we restrain β_u for $u > s+1$ on its first $d + k + 3$ places. Assuming that this restraint is successful, it follows that $c(\beta - q) \leqslant 2^{-(d+3)} + 2^{-(d+2)} < 2^{-(d+1)}$.

Finally we win by our second action, which is to add 2^{-d} to α_{s+1}. Then $\alpha - \alpha_s \geqslant 2^{-d}$, so $\alpha - \Phi_e(q) \geqslant 2^{-d} > c(\beta - q)$, as required.

The theorem now follows by a simple application of the finite injury priority method.

It is easy to see that $\alpha \leqslant_{\text{sw}} \beta$. When we add $2^{-(d+k+2)}$ to β_s, since $\beta_t(x) = 1$ for all x with $d \leqslant x \leqslant d + k + 2$, the effect is to make position $d-1$ of β change from 0 to 1. On the α side, the only change is that position $d - 1$ changes from 0 to 1. Hence we keep $A \leqslant_{\text{sw}} B$ (with constant 0). It is also clear that α is strongly c.e.. $\square$

Using this result and the following, we see that *sw*-reducibility and Solovay reducibility are incomparable even on the left-c.e. reals.

Theorem 13.5.7 (Downey, LaForte, Hirschfeldt [77]). *There exist left-c.e. reals $\alpha \leqslant_{\text{S}} \beta$ such that $\alpha \nleqslant_{\text{sw}} \beta$ (in fact, even $\alpha \nleqslant_{wtt} \beta$). Moreover, β can be chosen to be strongly c.e..*

Proof. The proof is a straightforward diagonalization argument, similar to the previous proof, but even easier. The strategy is described below. We build sets A and B and let $\alpha = 0.\chi_A$ and $\beta = 0.\chi_B$. We must meet the following requirements.

$$\mathcal{R}_{e,c} : \text{If } \Gamma_e \text{ has use } x + c \text{ then } \Gamma_e^B \neq A.$$

The idea is quite simple. We need only make B "sparse" and A "sometimes thick". That is, for the sake of $\mathcal{R}_{e,c}$, we set aside a block of $c+2$ positions of the binary expansion of β, say $n, n+1, \ldots, n+c+1$. Initially we have *none* of these numbers in B, but we put *all* of $n+1, \ldots, n+c+1$ into A. If we ever see a stage s where $\Gamma_{e,s}^{B_s}(n)\downarrow= 0$ with use $n+c$, we can satisfy the requirement by adding $2^{-(n+c+1)}$ to both α_s and β_s, the effect being that $B_s(n+c+1)$ changes from 0 to 1, $A_s(n+i)$ for $1 \leqslant i \leqslant c+1$ changes from 1 to 0, and $A_s(n)$ changes from 0 to 1.

It is easy to check that $\alpha \leqslant_{\mathrm{S}} \beta$ and that β is strongly c.e.. □

sw degrees are useful for many proofs on especially c.e. sets. Here is one example. We have seen that Ω defines the top Solovay degree for left-c.e. *reals*. We see that there is no biggest Solovay degree for c.e. *sets*.

There is a greatest S-degree of left-c.e. reals, namely that of Ω, but the situation is different for strongly c.e. reals.

Theorem 13.5.8. *Let α be strongly c.e.. There is a strongly c.e. real that is not sw-below α, and hence not S-below α.*

Proof. The argument is nonuniform, but is still finite injury. Since sw-reducibility and S-reducibility coincide for strongly c.e. reals, it is enough to build a strongly c.e. real that is not sw-below α. Let A be such that $\alpha = 0.\chi_A$. We build c.e. sets B and C to satisfy the following requirements.

$$\mathcal{R}_{e,i} : \Gamma_e^A \neq B \vee \Gamma_i^A \neq C,$$

where Γ_e is the eth wtt reduction with use less than $x+e$. It will then follow that either $0.\chi_B \not\leqslant_{\mathrm{sw}} \alpha$ or $0.\chi_C \not\leqslant_{\mathrm{sw}} \alpha$.

The idea for satisfying a single requirement $\mathcal{R}_{e,i}$ is simple. Let $l(e,i,s) = \max\{x : \forall y \leqslant x(\Gamma_{e,s}^{A_s}(y) = B_s(y) \wedge \Gamma_{i,s}^{A_s} = C_s(y))\}$. Pick a large number $k >> e, i$ and let $\mathcal{R}_{e,i}$ assert control over the interval $[k, 3k]$ in both B and C, waiting until a stage s such that $l(e,i,s) > 3k$.

First work with C. Put $3k$ into C, and wait for the next stage s' where $l(e,i,s') > 3k$. Note that some number must enter $A_{s'} - A_s$ below $3k+i$. Now repeat with $3k-1$, then $3k-2, \ldots, k$. In this way, $2k$ numbers are made to enter A below $3k+i$. Now we can win using B, by repeating the process and noticing that, by the choice of the parameter k, A cannot respond another $2k$ times below $3k+e$.

The theorem now follows by a standard application of the finite injury method. □

Given a left-c.e. real α with computable approximation $\alpha_s \to \alpha$, we can associate a set α^* as follows. Begin with $\alpha_0^* = \emptyset$. For all x and s, if either

$\alpha_s(x) = 0$ and $\alpha_{s+1}(x) = 1$, or $\alpha_s(x) = 1$ and $\alpha_{s+1}(x) = 0$, then for the least j with $\langle x, j\rangle \notin \alpha_s^*$, put $\langle x, j\rangle$ into α_{s+1}^*. Notice that α^* is a c.e. set.

Lemma 13.5.9. *$\alpha^* \leqslant_{\mathrm{sw}} \alpha$ and $\alpha \leqslant_{tt} \alpha^*$.*

Proof. Since α is nearly c.e., $\langle x, j\rangle$ enters α^* at a given stage only if some $y \leqslant x$ enters α at that stage. Such a y will also be below $\langle x, j\rangle$. Hence $\alpha^* \leqslant_{\mathrm{sw}} \alpha$ with use x. Clearly, $x \in \alpha$ if and only if α^* has an odd number of entries in row x, and furthermore, since α is nearly c.e., the number of entries in this row is bounded by x. Hence $\alpha \leqslant_{tt} \alpha^*$. $\square$

By the Lemma above, we see that another nice aspect of *sw*-reducibility is that if α is a left-c.e. real which is noncomputable, then there is a noncomputable strongly left-c.e. real $\beta \leqslant_{\mathrm{sw}} \alpha$, and this is *not* true in general, for $\leqslant_{\mathrm{S}}$. We have the following theorem.

Theorem 13.5.10 (Downey, Hirschfeldt, LaForte [77])**.** *There is a noncomputable left-c.e. real α such that all strongly c.e. reals dominated by α are computable.*

Proof. Recall that if we have left-c.e. reals $\beta \leqslant_{\mathrm{S}} \alpha$ then there are a left-c.e. real γ and a positive $c \in \mathbb{Q}$ such that $\alpha = c\beta + \gamma$.

Now let α be the noncomputable left-c.e. real α such that if A presents α then A is computable. We claim that, for this α, if $\beta \leqslant_{\mathrm{S}} \alpha$ is strongly c.e. then β is computable.

To verify this claim, let $\beta \leqslant_{\mathrm{S}} \alpha$ be strongly c.e.. We know that there is a positive $c \in \mathbb{Q}$ such that $\alpha = c\beta + \gamma$. Let $k \in \omega$ be such that $2^{-k} \leqslant c$ and let $\delta = \gamma + (c - 2^{-k})\beta$. Then δ is a left-c.e. real such that $\alpha = 2^{-k}\beta + \delta$.

It is easy to see that there exist computable sequences of natural numbers $b_0, b_1, \ldots$ and $d_0, d_1, \ldots$ such that $2^{-k}\beta = \sum_{i\in\omega} 2^{-b_i}$ and $\delta = \sum_{i\in\omega} 2^{-d_i}$. Furthermore, since β is strongly c.e., so is $2^{-k}\beta$, and hence we can choose $b_0, b_1, \ldots$ to be pairwise distinct, so that the nth bit of the binary expansion of $2^{-k}\beta$ is 1 if and only if $n = b_i$ for some i.

Since $\sum_{i\in\omega} 2^{-b_i} + \sum_{i\in\omega} 2^{-d_i} = 2^{-k}\beta + \delta = \alpha < 1$, Kraft-Chaitin inequality tells us that there is a prefix-free c.e. set $A = \{\sigma_0, \sigma_1, \ldots\}$ such that $|\sigma_0| = b_0$, $|\sigma_1| = d_0$, $|\sigma_2| = b_1$, $|\sigma_3| = d_1$, etc.. Now $\sum_{\sigma\in A} 2^{-|\sigma|} = \sum_{i\in\omega} 2^{-b_i} + \sum_{i\in\omega} 2^{-d_i} = \alpha$, and thus A presents α.

By our choice of α, this means that A is computable. But now we can compute the binary expansion of $2^{-k}\beta$ as follows. Given n, compute the number m of strings of length n in A. If $m = 0$ then $b_i \neq n$ for all i, and hence the nth bit of binary expansion of $2^{-k}\beta$ is 0. Otherwise, run through the b_i and d_i until either $b_i = n$ for some i or $d_{j_1} = \cdots = d_{j_m} = n$ for some $j_1 < \cdots < j_m$. By the definition of A, one of the two cases must happen. In the first case, the nth bit of the binary expansion of $2^{-k}\beta$ is 1. In the second case, $b_i \neq n$ for all i, and hence the nth bit of the binary expansion of $2^{-k}\beta$ is 0. Thus $2^{-k}\beta$ is computable, and hence so is β. $\square$

13.6 The bits of Ω and the Yu-Ding Theorem

This section could be called "when reducibilities go bad." We will see that in spite of a number of nice features mentioned in the previous section, *sw*-reducibility has a number of very undesirable features. By Theorem 13.4.3, and the note following it, *sw*-reducibility on the left-c.e. reals has is not a lower semilattice. However, it is also not an upper semilattice either, in that there is no *join* operation. This was first proven directly by Downey, LaForte, and Hirschfeldt in [77], but follows from the *proof* of the next result of Yu and Ding.

Theorem 13.6.1 (Yu and Ding [326]). *There is no sw-complete left-c.e. real.*

Actually, they prove something stronger:

Corollary 13.6.2 (Yu and Ding [326]). *There are two left-c.e. reals β and γ so that there is no left-c.e. real α with $\beta \leqslant_{\text{sw}} \alpha$ and $\gamma \leqslant_{\text{sw}} \alpha$.*

Proof. The proof we follow is due to Barmpalias and Lewis [21]. The procedure they use is idenatical to the Yu-Ding method, but the verification is a much smoother indiction and the proof significantly cleaner. Intuitively, a real α computing another real β with use x for every x means that as soon as β changes at position x, α must change at a position less than or equal to x. That is, if β can be computed with oracle α and use x, then α is not less than β. So if there were a largest c.e. *sw*-degree α, we could select two reals β_0 and β_1 and change them alternatively to drive α to be very large. Before proceeding with our proof, we would like to isolate a procedure which implements this idea.

It suffices to construct left-c.e. reals β, γ such that sor all *sw* precedures Φ, Ψ and left-c.e. reals α:

$$\mathcal{Q}^{\alpha}_{\Phi,\Psi} : \beta \neq \Phi^{\alpha} \ \vee \ \gamma \neq \Psi^{\alpha}$$

Definition 13.6.3. The positions on the right of the decimal point in a binary expansion are numbered as $1, 2, 3, \ldots$ from left to right. The positions on the left of the decimal point are numbered as $0, -1, -2, \ldots$.

Consider the following sw-game between α, γ, β. These numbers have initial values and during the stages of the game they can only increase. If β increases and i is the leftmost position where a β-digit change occurred, then α has to increase in such a way that some α-digit at a position $\leqslant i$ changes. This game describes an sw-reduction. If α has to code two reals β, γ then we get a similar game (were, say, at each stage only one of β, γ can change). We say that α follows the *least effort strategy* if at each stage it increases by the least amount needed. The following observation is in some sense at the heart of the Yu-Ding strategy.

Lemma 13.6.4. (Passing through lemma) *Suppose that in some game (e.g. like the above) α has to follow instructions of the type 'change a digit at position $\geqslant n$'. Although $\alpha_0 = 0$, some α' plays the same game while starting with $\alpha'_0 = \sigma$ for a finite binary expansion σ. If the strategies of α, α' are the same (i.e. the 'least effort' strategy described above) and the sequence of instructions only ever demand change at positions $> |\sigma|$ then at every stage s,*

$$\alpha'_s = \alpha_s + \sigma. \tag{13.2}$$

Proof. By induction on s we show that (13.2) holds and α'_s, α_s have the same expansions after position $|\sigma|$. For $s = 0$ it is obvious. Suppose that this double hypothesis holds at stage s. At $s + 1$, some demand for a change at some position $> |\sigma|$ appears and since α, α' look the same on these positions, α'_s will need to increase by the same amount that α_s needs to increase. So $\alpha'_{s+1} = \alpha_{s+1} + \sigma$ and one can also see that α, α' will continue to look the same at positions $> |\sigma|$ (consider cases whether the change occurred at positions $> |\sigma|$ or not). □

Definition 13.6.5 (The Yu-Ding Procedure)**.** Given $n > 0$ and $t \in \mathbb{Z}$ we are going to define the *Yu-Ding procedure* amongst α, β, γ with attack interval $(t - n, t]$. We assume that α, β, γ have initial value 0. Repeat the following instructions (at expansionary stages s) until $\beta(i) = \gamma(i) = 1$ for all $i \in (t - n, t]$.

s odd
1. let $\beta = \beta + 2^{-t}$ and let b equal the highest (i.e. leftmost) position where a digit change occurs in β.
2. Add to α the least amount which causes a change in a digit at position b or higher.

s even
1. let $\gamma = \gamma + 2^{-t}$ and let g equal the highest (i.e. leftmost) position where a digit change occurs in γ.
2. Add to α the least amount which causes a change in a digit at position g or higher.

Before continuing with our proof, we give an example for n=2.
stage 1: $\beta_1 = 0.001$, $\gamma_1 = 0$ and $\alpha_1 = 0.001$
stage 2: $\beta_2 = 0.001$, $\gamma_2 = 0.001$ and $\alpha_2 = 0.010$
stage 3: $\beta_3 = 0.010$, $\gamma_3 = 0.001$ and $\alpha_3 = 0.100$
stage 4: $\beta_4 = 0.010$, $\gamma_4 = 0.010$ and $\alpha_4 = 0.110$
stage 5: $\beta_5 = 0.011$, $\gamma_5 = 0.010$ and $\alpha_5 = 0.111$
stage 6: $\beta_6 = 0.011$, $\gamma_6 = 0.011$ and $\alpha_6 = 1.000$
stage 7: $\beta_7 = 0.100$, $\gamma_7 = 0.011$ and $\alpha_7 = 1.100$
stage 8: $\beta_8 = 0.100$, $\gamma_8 = 0.100$ and $\alpha_8 = 10.000$

It is not hard to see that the above procedure describes how α evolves when it tries to code β, γ via sw-reductions with identity use and it uses the *least effort strategy* (provided that the changes in β, γ occur at expansionary stages). Player α follows the *least effort strategy* when it increases by the least amount which can rectify the functionals holding its computations of β, γ.

Lemma 13.6.6. *Let $n > 0$. For any $k \in \mathbb{Z}$ the Yu-Ding procedure amongst α, β, γ with attack interval $(k, k+n]$ ends up with $\alpha = n2^{-k}$.*

Proof. By induction: for $n = 1$ it is obvious. Assume that it holds for n. Now pick $k \in \mathbb{Z}$ and consider the attack using $(k-1, k+n]$. It is clear that up to a stage s_0 this will be identical to the procedure with attack interval $(k, k+n]$. By the induction hypothesis $\alpha_{s_0} = n2^{-k}$ and $\beta(i) = \gamma(i) = 1$ for all $i \in (k, k+n]$, while $\beta(k) = \gamma(k) = 0$. According to the next step β changes at position k and this forces α to increase by 2^{-k} ($\alpha = \alpha + 2^{-k}$) since α has no 1s lower than position k. Then γ does the same move and since α still has no 1s lower than position k, $\alpha = \alpha + 2^{-k}$ once again. So far

$$\alpha = n2^{-k} + 2^{-k} + 2^{-k} = n2^{-k} + 2^{-(k-1)}$$

and $\beta(i) = \gamma(i) = 0$ for all $i \in (k, k+n]$ while $\beta(k) = \gamma(k) = 1$. By applying the induction hypothesis again and the passing through lemma 13.6.4 the further increase of α will be exactly $n2^{-k}$. So

$$\alpha = n2^{-k} + 2^{-(k-1)} + n2^{-k} = (n+1)2^{-(k-1)}$$

as required. □

Now let us call *Yu-Ding strategy* with attack interval $(t-n, t]$ the enumerations of β, γ as in the Yu-Ding procedure. In the context of a requirement $\mathcal{Q}^{\alpha}_{\Phi,\Psi}$ we assume that each step is performed only when the reductions $\Phi^{\alpha} = \beta, \Psi^{\alpha} = \gamma$ are longer than ever before (i.e. at an expansionary stage).

Lemma 13.6.7. *In a game where α has to follow instructions of the type 'change a digit at position $\geqslant n$' (e.g. an sw-game between α and β, γ) the least effort strategy is a best strategy for α. In other words if a different strategy produces α' then at each stage s of the game $\alpha_s \leqslant \alpha'_s$.*

Proof. By induction on the stages s. At stage 0, $\alpha \leqslant \alpha'$. If $\alpha_s \leqslant \alpha'_s$ then there will be a position n such that $0 = \alpha_s(n) < \alpha'_s(n) = 1$ and $\alpha_s \upharpoonright n = \alpha'_s \upharpoonright n$. When β or γ make a move, α, α' will have to change on position t or higher in stage $s+1$; if $t < n$ it is clear that $\alpha_{s+1} \leqslant \alpha'_{s+1}$. Otherwise the highest change α will be forced to do is on n and so again $\alpha_{s+1} \leqslant \alpha'_{s+1}$. □

Although we implicitely assumed that the use in the functionals of $\mathcal{Q}$ is the identity function x, the case when it is $x + c$ is not different: the Yu-Ding strategy with attack interval $(k, t]$ against $\mathcal{Q}'$ where the use of both functionals is (bounded by) $x + c$ gives the same result (assuming the least effort strategy on the part of α) with the Yu-Ding strategy in $(k+c, t+c]$ against $\mathcal{Q}$ where the use of both functionals is the identity. So, from lemma 13.6.7 and proposition 13.6.6 we get

Corollary 13.6.8. *If β, γ follow the Yu-Ding strategy in attacking $\mathcal{Q}^\alpha$ (where the functionals have use bounded by $x+c$) with attack interval $(k, k+n]$ then either $\mathcal{Q}^\alpha$ is satisfied or $\alpha \geqslant n2^{-(k+c)}$.*

The above corollary is all we need to prove the theorem. Assume an effective list of all requirements and successively assign attack intervals to them. We will diagonalize against all $\alpha \leqslant 1$. Tnat will suffice. The methods above allows us to drive α to large numbers. If the attack interval for $\mathcal{Q}_i$ is $(k, n]$ define the one for $\mathcal{Q}_{i+1}$ to be $(n, t]$ where t is the least such that the estimation of corollary 13.6.8 gives $\alpha \geqslant 1$. Now assume that $\alpha \in [0, 1)$ and apply the Yu-Ding strategy for each of the requirements on the relevant intervals in a global construction. There is no interaction amongst the strategies and the satisfaction of all the requirements follows from corollary 13.6.8.

□

The Kučera-Slaman Theorem says that all 1-random reals are the same in terms of their complexity oscillations, and have sequences converging to them at esssentially the same rates. The Yu-Ding Theorem says that, in general, there is no *efficient* algorithm (in terms of the number of bits used) to take the *bits* of one left-c.e. real to the bits of any particular version of Ω.

We remark it is still unknown if there are *sw* incomparible versions *of* Ω.

In spite of the fact that there is no maximal c.e. *sw* degree, we can say a little about the *sw* degrees of 1-random left-c.e. reals.

Theorem 13.6.9. *Suppose that A is a c.e. set and α is a 1-random left-c.e. real. Then $A \leqslant_{\text{sw}} \alpha$, and this is true with identity use.*

Proof. Given A. α as above we must construct $\Gamma^\alpha = A$, where $\gamma(x) = x$. Fix a universal prefix-free machine U and using Kraft-Chaitin, and the Recursion Theorem, we will be building a part of U, so that if we enumerate a Kraft-Chaitin axiom $\langle 2^n, \sigma \rangle$ we will know that some τ, σ enters U for some τ of length $e+n$. Since α is 1-random we know that for all n, $K_U(\alpha \restriction n) \geqslant n - c$ for a fixed c, which we will know for the sake of this construction.

Let $\alpha_s \to \alpha$ be a computable sequence converging to α and let $A = \cup_s A_s$. Initially, we will define $\Gamma^{\alpha_s}(x) = 0$ for all x, and maintain this unless x enters $A_{s+1} - A_s$. As usual at such a stage, we would like to change the answer from 0 to 1. To do this we will need $\alpha \restriction x \neq \alpha_s \restriction x$. Should we see a stage $t \geqslant s$ with $\alpha_t \restriction x \neq \alpha_s \restriction x$ then we can so change the answer. For $x > e + c + 2$, we can force this to happen. We simply enumerate an axiom $\langle 2^{x-c-e-1}, \alpha_s \restriction x \rangle$ into our machine, calusing a description of $\alpha_s \restriction x$ of length $x - c - 1$ to enter $U - U_s$, and hence $\alpha \restriction x \neq \alpha_s \restriction x$. We can simply wait for this to happen, correct Γ and move to the next stage. □

Corollary 13.6.10 (Calude and Nies [37]). *Ω is wtt-complete.*

13.7 Relative K-reducibility

We would like a measure of relative randomness combining the best of S-reducibility and sw-reducibility.

Both S-reducibility and sw-reducibility are uniform in a way that relative initial-segment complexity is not. This makes them too strong, in a sense, and it is natural to wish to investigate nonuniform versions of these reducibilities. Motivated by this consideration, as well as by the problems with sw-reducibility, we introduce another measure of relative randomness, called relative K reducibility, which can be seen as a nonuniform version of both S-reducibility and sw-reducibility, and which combines many of the best features of these reducibilities. Its name derives from a characterization, discussed below, which shows that there is a very natural sense in which it is an *exact* measure of relative randomness.

Definition 13.7.1 (Downey, Hirschfeldt, LaForte [77])**.** Let α and β be reals. We say that β is *relative K reducible* (rK-reducible) to α, and write $\beta \leqslant_{rK} \alpha$, if there exist a partial computable binary function f and a constant k such that for each n there is a $j \leqslant k$ for which $f(\alpha \upharpoonright n, j)\downarrow = \beta \upharpoonright n$.

Clearly $\leqslant_{rK}$ is transitive. It might seem like a weird definition at first, but the actual motivation came from the consideration of Lemma 13.2.4. There we argued that if two strings are very close and of the same length then they have essentially the same complexity no matter whether we use K or C. Note that *sw* reducibility gives a method of taking an initial segment of length n of β to one of length $n-c$ of α. However, it would be enough to take *some* string k-close to an initial segment or β to one similarly close to one of α. This idea gives a notion equivalent to rK reducibility and leads to the definition above.

There are, in fact, several characterizations of rK-reducibility, each revealing a different facet of the concept. We mention three, beginning with a "relative entropy" characterization whose proof is quite straightforward. For a left-c.e. real β and a fixed computable approximation $\beta_0, \beta_1, \dots$ of β, we will let the mind-change function $m(\beta, n, s, t)$ be the cardinality of

$$\{u \in [s,t] : \beta_u \upharpoonright n \neq \beta_{u+1} \upharpoonright n\}.$$

Lemma 13.7.2 (Downey, Hirschfeldt, LaForte [77])**.** *Let α and β be left-c.e. reals. The following condition holds if and only if $\beta \leqslant_{rK} \alpha$. There are a constant k and computable approximations $\alpha_0, \alpha_1, \dots$ and $\beta_0, \beta_1, \dots$ of α and β, respectively, such that for all n and $t > s$, if $\alpha_t \upharpoonright n = \alpha_s \upharpoonright n$ then $m(\beta, n, s, t) \leqslant k$.*

The following is a more analytic characterization of rK-reducibility, which clarifies its nature as a nonuniform version of both S-reducibility and sw-reducibility.

Lemma 13.7.3 (Downey, Hirschfeldt, LaForte [77]). *For any reals α and β, the following condition holds if and only if $\beta \leqslant_{rK} \alpha$. There are a constant c and a partial computable function φ such that for each n there is a τ of length $n+c$ with $|\alpha - \tau| \leqslant 2^{-n}$ for which $\varphi(\tau)\downarrow$ and $|\beta - \varphi(\tau)| \leqslant 2^{-n}$.*

Proof. First suppose that $\beta \leqslant_{rK} \alpha$ and let f and k be as in Definition 13.7.1. Let c be such that $2^c \geqslant k$ and define the partial computable function φ as follows. Given a string σ of length n, whenever $f(\sigma, j)\downarrow$ for some new $j \leqslant k$, choose a new $\tau \supseteq \sigma$ of length $n+c$ and define $\varphi(\tau) = f(\sigma, j)$. Then for each n there is a $\tau \supseteq \alpha \upharpoonright n$ such that $\varphi(\tau)\downarrow = \beta \upharpoonright n$. Since $|\alpha - \tau| \leqslant |\alpha - \alpha \upharpoonright n| \leqslant 2^{-n}$ and $|\beta - \beta \upharpoonright n| \leqslant 2^{-n}$, the condition holds.

Now suppose that the condition holds. For a string σ of length n, let S_σ be the set of all μ for which there is a τ of length $n+c$ with $|\sigma - \tau| \leqslant 2^{-n+1}$ and $|\mu - \varphi(\tau)| \leqslant 2^{-n+1}$. It is easy to check that there is a k such that $|S_\sigma| \leqslant k$ for all σ. So there is a partial computable binary function f such that for each σ and each $\mu \in S_\sigma$ there is a $j \leqslant k$ with $f(\sigma, j)\downarrow = \mu$. But, since for any real γ and any n we have $|\gamma - \gamma \upharpoonright n| \leqslant 2^{-n}$, it follows that for each n we have $\beta \upharpoonright n \in S_{\alpha \upharpoonright n}$. Thus f and k witness the fact that $\beta \leqslant_{rK} \alpha$. □

The most interesting characterization of rK-reducibility (and the reason for its name) is given by the following result, which shows that there is a very natural sense in which rK-reducibility is an exact measure of relative randomness. Recall that the prefix-free complexity $K(\tau \mid \sigma)$ of τ relative to σ is the length of the shortest string μ such that $U^\sigma(\mu)\downarrow = \tau$, where U is a fixed universal self-delimiting machine (and similarly for $C(\tau \mid \sigma)$).

Theorem 13.7.4 (Downey, Hirschfeldt, LaForte [77]). *Let α and β be reals. Then $\beta \leqslant_{rK} \alpha$ if and only if $K(\beta \upharpoonright n \mid \alpha \upharpoonright n) = \mathcal{O}(1)$.*

Proof. First suppose that $\beta \leqslant_{rK} \alpha$ and let f and k be as in Definition 13.7.1. Let m be such that $2^m \geqslant k$ and let $\tau_0, \ldots, \tau_{2^m-1}$ be the strings of length m. Define the self-delimiting machine N to act as follows with σ as an oracle. For all strings μ of length not equal to m, let $N^\sigma(\mu)\uparrow$. For each $i < 2^m$, if $f(\sigma, i)\downarrow$ then let $N^\sigma(\tau_i)\downarrow = f(\sigma, i)$, and otherwise let $N^\sigma(\tau_i)\uparrow$. Let e be the coding constant of N and let $c = e + m$. Given n, there exists a $j \leqslant k$ for which $f(\alpha \upharpoonright n, j)\downarrow = \beta \upharpoonright n$. For this j we have $N^{\alpha \upharpoonright n}(\tau_j)\downarrow = \beta \upharpoonright n$, which implies that $K(\beta \upharpoonright n \mid \alpha \upharpoonright n) \leqslant |\tau_j| + e \leqslant c$.

Now suppose that $K(\beta \upharpoonright n \mid \alpha \upharpoonright n) \leqslant c$ for all n, where K is defined using the universal self-delimiting machine U. Let $\tau_0, \ldots, \tau_k$ be a list of all strings of length less than or equal to c and define f as follows. For a string σ and a $j \leqslant k$, if $U^\sigma(\tau_j)\downarrow$ then $f(\sigma, j)\downarrow = U^\sigma(\tau_j)$, and otherwise $f(\sigma, j)\uparrow$. Given n, since $K(\beta \upharpoonright n \mid \alpha \upharpoonright n) \leqslant c$, it must be the case that $U^{\alpha \upharpoonright n}(\tau_j)\downarrow = \beta \upharpoonright n$ for some $j \leqslant k$. For this j we have $f(\alpha \upharpoonright n, j)\downarrow = \beta \upharpoonright n$. Thus $\beta \leqslant_{rK} \alpha$. □

Note that $K(\beta \upharpoonright n \mid \alpha \upharpoonright n) = \mathcal{O}(1)$ if and only if $C(\beta \upharpoonright n \mid \alpha \upharpoonright n) = \mathcal{O}(1)$, so there is no separate notion of rC-reducibility.

An immediate consequence of Theorem 13.7.4 is that rK-reducibility satisfies the Solovay property.

Corollary 13.7.5. *If $\beta \leqslant_{rK} \alpha$ then $K(\beta \upharpoonright n) \leqslant K(\alpha \upharpoonright n) - \mathcal{O}(1)$ and $C(\beta \upharpoonright n) \geqslant C(\alpha \upharpoonright n)\mathcal{O}(1)$.*

Additionally, rK-reducibility is indeed a simultaneous generalization of both S and sw-reducibility. This follows by Lemma 13.7.3.

Theorem 13.7.6 (Downey, Hirschfeldt, LaForte [77])**.** *Let α and β be left-c.e. reals. If $\beta \leqslant_{\mathrm{S}} \alpha$ or $\beta \leqslant_{\mathrm{sw}} \alpha$, then $\beta \leqslant_{rK} \alpha$.*

We remark that sw-reducibility and S-reducibility do not coincide with rK-reducibility on even the c.e. sets.

Theorem 13.7.7 (Downey, Hirschfeldt, LaForte [77])**.** *There exist strongly left-c.e. reals α and β such that $\beta \leqslant_{rK} \alpha$ but $\beta \nleqslant_{\mathrm{sw}} \alpha$ (equivalently, $\beta \nleqslant_{\mathrm{S}} \alpha$).*

Proof. We build c.e. sets A and B to satisfy the following requirements.

$$\mathcal{R}_e : \Gamma_e^A \neq B,$$

where Γ_e is the eth wtt reduction with use less than $x + e$. We think of α and β as $0.\chi_A$ and $0.\chi_B$, respectively, and we build A and B in such as way as to enable us to apply Lemma 13.7.2 to conclude that $\beta \leqslant_{rK} \alpha$.

The construction is a standard finite injury argument. We discuss the satisfaction of a single requirement $\mathcal{R}_e$. For the sake of this requirement, we choose a large n, restrain n from entering B, and restrain $n+e+1$ from entering A. If we find a stage s such that $\Gamma_{e,s}^{A_s}(n)\downarrow= 0$ then we put n into B, put $n+e+1$ into A, and restrain the initial segment of A of length $n+e$. Unless a higher priority strategy acts at a later stage, this guarantees that $\Gamma_e^A(n) \neq B(n)$.

Furthermore, it is not hard to check that, because of the numbers that we put into A, for each n and $t > s$, if $\alpha_t \upharpoonright n = \alpha_s \upharpoonright n$ then $m(\beta, n, s, t) \leqslant 2$ (where $m(\beta, n, s, t)$ is as defined before Lemma 13.7.2). Thus, by Lemma 13.7.2, $\beta \leqslant_{rK} \alpha$. $\square$

Despite the nonuniform nature of its definition, rK-reducibility implies Turing reducibility.

Theorem 13.7.8 (Downey, Hirschfeldt, LaForte [77])**.** *If $\beta \leqslant_{rK} \alpha$ then $\beta \leqslant_T \alpha$.*

Proof. Let k be the least number for which there exists a partial computable binary function f such that for each n there is a $j \leqslant k$ with $f(\alpha \upharpoonright n, j)\downarrow= \beta \upharpoonright n$. There must be infinitely many n for which $f(\alpha \upharpoonright n, j)\downarrow$ for all $j \leqslant k$, since otherwise we could change finitely much of f to contradict the minimality of k. Let $n_0 < n_1 < \cdots$ be an α-computable sequence of such n. Let T be the α-computable subtree of 2^ω obtained by pruning, for each i, all the strings of length n_i except for the values of $f(\alpha \upharpoonright n_i, j)$ for $j \leqslant k$.

If γ is a path through T then for all i there is a $j \leqslant k$ such that γ extends $f(\alpha \upharpoonright n_i, j)$. Thus there are at most k many paths through T, and hence each path through T is α-computable. But β is a path through T, so $\beta \leqslant_T \alpha$. □

Notice that, since any computable real is obviously rK-reducible to any other real, the above theorem shows that the computable reals form the least rK-degree. We can use Theorem 13.4.3 to show that the rK-degrees of left-c.e. reals is not a lattice. However, the structure of the rK-degrees og left-c.e. reals is rather more tractable that that of the sw-degrees. The fact that Solovay reducibility implies rK-reducibility means that the Kučera-Slaman Theorem shows that $[\Omega]_{rK}$ is the top degree. Thus to apply Thoeorem **??**, we only need to prove that $+$ is a join.

Theorem 13.7.9 (Downey, Hirschfeldt, LaForte [77]). *The rK-degrees of left-c.e. reals form an uppersemilattice with least degree that of the computable sets and highest degree that of* Ω.

Proof. All that is left to show is that addition is a join. Since $\alpha, \beta \leqslant_S \alpha+\beta$, it follows that $\alpha, \beta \leqslant_{rK} \alpha+\beta$. Let γ be a left-c.e. real such that $\alpha, \beta \leqslant_{rK} \gamma$. Then Lemma 13.7.2 implies that $\alpha + \beta \leqslant_{rK} \gamma$, since for any n and $s < t$ we have $m(\alpha + \beta, n, s, t) \leqslant 2(m(\alpha, n, s, t) + m(\beta, n, s, t)) + 1$. □

We remark that the remaining part of Theorem 13.4.1 was that $\leqslant_S$ is distributive on the left-c.e. reals. Whether distributitivy holds for $\leqslant_{rK}$ is open at present.

One nice property observed by Raichev is that the reals $\leqslant_{rK}$ below a fixed one are well behaved.

Theorem 13.7.10 (Raichev [241, 242]). *For any fixed real* y*, the collection of reals* $\mathbb{R}_y = \{x : x \leqslant_{rK} y\}$ *forms a real closed field.*

This result can be proven using similar approximation methods to the proof of Theorem 8.5.8. We refer the reader to Raichev [**?**] for details.

We have seen that rK-reducibility shares many of the nice structural properties of S-reducibility on the left-c.e. reals, while still being a reasonable reducibility on non-left-c.e. reals. Together with its various characterizations, especially the one in terms of relative K-complexity of initial segments, this makes rK-reducibility a tool with great potential in the study of the relative randomness of reals. As one would expect, little else is known about the structure of rK degrees. For example, it is still an open question if every real is rK reducible to a random one.

As we see in Chapter 14, rK-reducibility plays an important role in the study of the relative randomness of *random* reals.

13.8 A Minimal rK-degree.

In this section, we will prove the result below.

Theorem 13.8.1 (Raichev and Stephan [**?**]). *There is a minimal rK-degree.*

The reason that this result is interesting is that rK is one of the few measures of relative randomness for which we know whether or not there is a minimal degree. The proof below uses the fact that on very sparse sets, rK behaves in a reasonably weel-behaved way. Merkle and Stephan [**?**] have exploited this idea of using sparse sets to establish results about measures of relative randomness quite fruitfully, as we see in the next sections.

Proof. We build a Π^0_1 class $[T]$ for a computable tree T with no computable paths. T will have the property that for all total functionals Φ, for all $X \in [T]$, there is a string $\sigma \prec X$, such that one of the following holds:

(i) Φ^X and Φ^Y are compatible for all $Y \in [T]$ extending σ or

(ii) For all $Z \neq Y$ extending σ in $[T]$, Φ^Z and Φ^Y are incompatible.

We make the set S of splitting nodes of $[T]$ computably sparse. That is, for all computable functions g,

$$(a.a.\sigma \in S)(\forall \tau \in S)[\sigma \prec \tau \rightarrow g(|\sigma|) < |\tau|].$$

Raichev and Stephan's proof uses movable markers. For a string ν, m_ν denotes a position of a splitting node on T_s. At stage 0, $T_0 = 2^{<\omega}$ and $m_\nu = \nu$. At each stage s we will prune T_s to make T_{s+1}. The basic action is called procedure $\mathrm{CUT}(\sigma, \tau)$. This can be invoked for $\sigma \prec \tau$. This prunes all paths that extend $m_{\sigma'}$ for all $\sigma \preccurlyeq \sigma' \prec \tau$, but not m_τ Then all markers are moved accordingly. Namely, we move m_σ to m_τ, and $m_{\sigma\nu}$ to $m_{\tau\nu}$.

The construction at stage s works for nodes of size $\leqslant s$ and has the following actions.

(a) If ther eis a σ, $i < 2$ and $e \leqslant |\sigma|$ with $\Phi_e(x) = m_{\sigma i}(x)$ for all $x \leqslant |\sigma|$, then invoke $\mathrm{CUT}(\sigma, \sigma(1-i))$.

(b) If there is a σ, δ, ν, and $e \leqslant |\sigma|$, with $\Phi_e^{m_{\sigma 0}}$ and $\Phi_e^{m_{\sigma 1}}$ compatible for all arguments $y \leqslant |\sigma|$, but $\Phi_e^{m_{\sigma 0 \delta}}$ and $\Phi_e^{m_{\sigma 1 \nu}}$ are incompatible at some srgument $y \leqslant |\sigma|$, then $\mathrm{CUT}(\sigma 0, \sigma 0 \delta)$ and $\mathrm{CUT}(\sigma 1, \sigma 1 \nu)$.

(c) Finally, if there exist σ, τ, ν and $e \leqslant |\sigma|$, with $\sigma \prec \tau \prec \nu$, and $|m_\tau| \leqslant \Phi_e(|m_\sigma|) < m_\nu$, invoke $\mathrm{CUT}(\tau, \nu)$.

It is easy to show that all the markers stop moving at some stage. Additionally, at each stage the tree T_s is perfect, and hence $\cap_s T_s$ is a perfect tree. By (a), T has no computable members and by (b) and (c), the tree satisfies (i) or (ii), and the splitting nodes are computably sparse. Now let A be a path on T of hyperimmune free degree, using the Hyperimmune-free basis theorem Now let A be a path on T of hyperimmune free degree, obtained using the Hyperimmune-Free Basis Theorem, Theorem 5.16.9. We claim that A has minimal rK degree. Suppose that $i\emptyset \neq_T B \leqslant_{rK} A$. Then $B \leqslant_T A$ also and hence, as A is hyperimmune free, $B \leqslant_{tt} A$. Let $\Phi^A = B$ be the witnessing truth table reduction with computable use φ.

We show that $A \leqslant_{rK} B$. Let σ be the relevant string of (i) for A. We need the following lemma.

Lemma 13.8.2. *For almost all n, and almost all stages t, T_t has at most two extensions of σ of length n with extensions in T_t, that map to $B \upharpoonright n$ under Φ.*

Proof. Let f be the function defined on $k > |\sigma|$, as follows. $f(k)$ is the first stage such that for all strings ν with $A \upharpoonright (k-1)\widehat{\ }(1-A(k)) \prec \nu$ of length $\varphi(s)$ on T_s, there exists $x \leqslant s$ with $\Phi^\nu(x) \downarrow\neq \Phi^A(x)$. Let $f(k) = 1$ for $k \leqslant |\sigma|$. f is total, else there for all s there is a string ν_s with $A \upharpoonright (k-1)\widehat{\ }(1-A(k)) \prec \nu_s$ of length $\varphi(s)$ on T_s with $\Phi^{\nu_s}(x) = \Phi^A(x)$ for all $x \leqslant s$. Then $Y = \liminf_s \nu_s \in [T]$ is distinct from A and $\Phi^A = \Phi^Y$. However, (ii) holds as B is noncomputable, and this is a contradiction.

Notice that f is A-computable, and hence as A is hyperimmune free, there is a computable function g majorizing f.

Now choose n larger than $|\sigma|$, the length that T becomes g-sparse, and the length of the first splitting node of A on T. Let τ be the last splitting node on $A \upharpoonright n$ and $\nu \prec \tau$ any splitting node extending σ. Then by sparseness, we know that for $s = f(|\sigma|)$,

$$s \leqslant g(|\sigma|) < |tau| \leqslant n.$$

Thus by stage s, every $\rho \in T_s$ extending $A \upharpoonright (|\nu|-1)\widehat{\ }(1-A(|nu|)) = \nu\widehat{\ }(1-A(|nu|)$, will have an argument $x \leqslant s$ with $\Phi^\rho(x) \neq \Phi^A(x) = B(x)$. Therefore ρ cannot map to $B \upharpoonright n$ under Φ. As ν is an arbitary splitting node of T below the last splitting node of $A \upharpoonright n$, only strings extending the last splitting node of $A \upharpoonright n$ can map to $B \upharpoonright n$ under Φ. □

To complete the proof, given $B \upharpoonright n$, run through computable approximations of T_s until a sufficiently large stage is found where T_t has at most two extensions of σ that can map to $B \upharpoonright n$ under Φ. t exist by Lemma 13.8.2. We can find these extensions effectively from $B \upharpoonright n$ as Φ is a tt-reduction. The output the two strings found. One will be $A \upharpoonright n$. This is an rK reduction. □

Raichev and Stephan [**?**] establish varions facts about minimal rK degrees. For example, they show that there are $2^{\aleph_0}$ many of them, as the Hyperimmune-Free Basis Theorem shows that every Π^0_1 class with no computable members has $2^{\aleph_0}$ many hyperimmune-free members. Additionally, they show that minimal rK degrees have fairly low initial segment compelxity. To wit, the show the following.

Theorem 13.8.3 (Raichev and Stephan [?]). *If A has minimal rK degree, then for any computable order g, and Q either C or K,*

$$Q(A \upharpoonright n) \leqslant^+ Q(n) + g(n).$$

Proof. Given any computable order h, define the *h-dilution* of A as $A_h(h(n)) = A(n)$ and $A_h(k) = 0$ for $k \neq h(n)$ for any n. Notice that $A_h \leqslant_S A$ and hence $A \equiv_{rK} A$, if A has minimal rK degree. To describe $A_h(h(k)) \upharpoonright h(k)$ only needs $Q(h(k))$ plus k bits of information. Given any g, it is easy to choose a slowly enough growing h to make $Q(A \upharpoonright n) \leqslant^+ Q(n) + g(n)$. □

Of course this shows that if A has minimal rK degree it is far from random. By the Kučera-Gács Theorem, Theorem 11.3.2, we know that if there is a 1-random R with $A \leqslant_{wtt} R$. Now choose a dilution h growing much slower than the use of the reduction $A \leqslant_{wtt} R$. Then $A_h \leqslant_S R$ and hence we have the following.

Theorem 13.8.4 (Raichev and Stephan [?]). *Every real of minimal rK degree is rK reducible to a random real.*

Raichev and Stephan also show that there are random reals with no minimal rK degrees below them.

13.9 $\leqslant_K$ and $\leqslant_C$.

Recall that the most basic measures of relative initial segment complexity are $\leqslant_K$ and $\leqslant_C$, where, for instance, $\alpha \leqslant_K \beta$ iff for all n, $K(\beta \upharpoonright n) \geqslant K(\alpha \upharpoonright n) - \mathcal{O}(1)$. We will refer to $\leqslant_K$ and $\leqslant_C$ as "reducibilities" although on the face of it there is no reason they should be reducibilities in the computational sense. Indeed, as we soon see they are *not* reducibilities, and $\leqslant_K$ is not even a reducibility on the left-c.e. reals.

But this anticipates things. A basic question motivating the paper [77] was the following. Suppose that $\alpha \leqslant_K \beta$. Does it follow that $\beta \leqslant_{rK} \alpha$? Does it even follow that $\beta \leqslant_T \alpha$?

Although it might seem at first that the answer to this question should obviously be negative, at first glance, Theorem 13.9.1 would seem to indicate that any counterexample would probably have to be quite complicated, and gives us hope for a positive answer. Actually, sometimes $\leqslant_K$ does imply $\leqslant_{\mathrm{sw}}$ at least on the left-c.e. reals.

Theorem 13.9.1 (Downey, Hirschfeldt, LaForte [77]). *Let α and β be left-c.e. reals such that $\liminf_n K(\alpha \upharpoonright n) - K(\beta \upharpoonright n) = \infty$. Then $\beta <_{\mathrm{sw}} \alpha$.*

Proof. Let $c_\alpha(n)$ be the least s such that $\alpha_s \upharpoonright n = \alpha \upharpoonright n$, and define $c_\beta(n)$ analogously. Let U be a universal self-delimiting computer and define the self-delimiting computer N as follows. For each n, s, and σ, if $U(\sigma)[s]\downarrow=$

$\beta_s \upharpoonright n$ and $N(\sigma)$ has not been defined before stage s then let $N(\sigma)\downarrow = \alpha_s \upharpoonright n$. Let e be the coding constant of N. For each n, if $c_\beta(n) \geqslant c_\alpha(n)$ then $\forall\sigma(U(\sigma)\downarrow = \beta \upharpoonright n \Rightarrow N(\sigma)\downarrow = \alpha \upharpoonright n)$, which implies that $K(\alpha \upharpoonright n) \leqslant K(\beta \upharpoonright n) + e$. Thus our hypothesis implies that $c_\beta(n) < c_\alpha(n)$ for almost all n, which clearly implies that $\beta \leqslant_{\text{sw}} \alpha$. We note that, $\alpha \nleqslant_{\text{sw}} \beta$, so $\beta <_{\text{sw}} \alpha$ $\square$

Stephan [291] has shown that the Theorem above has limited use because it is hard to satisfy the hypotheses of the Theorem.

Theorem 13.9.2 (Stephan [291]). *Suppose that α and β satisfy the hypotheses of Theorem 13.9.1. Then α is wtt-complete.*

The following proof is due to the authors. Stephan's original proof can be found in Downey [71].

Proof. Suppose that α and β are left-c.e. reals such that $\liminf_n K(\alpha \upharpoonright n) - K(\beta \upharpoonright n) = \infty$. As usual we will affix $[s]$ to denote the stage s situation. We build a reduction $\Gamma^\alpha = H$ where H denotes the halting problem. The method here is similar to that of our proof of Theorem 9.2.5, and of the Kučera-Slaman Theorem.. We begin with $\Gamma^\alpha(x) = 0$ for all x. Again we will build part of the universal machine, U and our action will have coding constant c. For x, if $K(\beta \upharpoonright x) > x + c + 2[s]$, we work as follows. Should $x \in H_s$ and should no $y \geqslant x$ be pending (to be defined), our action is as follows. We will want to change $\Gamma^\alpha(x)$ from 0 to 1. If $K(\alpha \upharpoonright x) \geqslant K(\beta \upharpoonright x) + c + 2[s]$, enumerate a Kraft-Chaitin axiom $\langle 2^{-(K(\beta \upharpoonright x)[s])}, \alpha \upharpoonright x[s]\rangle$ into our part of the universal machine, and declare that x is pending. Our action puts a description of $\alpha \upharpoonright x[s]$ of length $K(\beta \upharpoonright x)[s] + c$ into $U[t] - U[s]$.

Now x remains pending until we get to change $\Gamma^\alpha(x)[t]$ at some $t \geqslant s$, if ever. While x is pending and $K(\beta \upharpoonright x)[t] < K(\beta \upharpoonright x)[s]$, *and additionally,* $K(\alpha \upharpoonright x) \geqslant K(\beta \upharpoonright x) + c + 2[t]$, we will repeat this action. Notice that our action is okay as we are "pushing the opponent's quanta." That is the total amount we put into our machine is bound by the measure of the descriptions of $\beta \upharpoonright x[s]$. Thus Kraft-Chaitin will apply.

Now should it be the case that α remains pending forever, we see that $K(\alpha \upharpoonright x) < K(\beta \upharpoonright x) + c + 2$. However, for almost all x, $K(\alpha \upharpoonright x) \geqslant K(\beta \upharpoonright x) + c + 2$. Hence for almost all x, we will get to correct $\Gamma^\alpha(x) = 1$, where necessary. $\square$

The reader should note that the use of the procedure above is the identity. Applying this result to the special case that $\beta = 1^\omega$, we see the following.

Corollary 13.9.3. *Suppose that α is a left-c.e. real, and not sw-above all c.e. sets with identity use. In particular suppose that α is strongly c.e. or that it has incomplete wtt-degree. Then for some c,*

$$\exists^\infty n[K(\alpha \upharpoonright n) < K(1^n) + c].$$

Stephan also points out the following analogous situation for $\leqslant_C$, which he attributes to folklore.

Theorem 13.9.4. *Suppose that* $\exists^\infty n[C(A \upharpoonright n) \leqslant C(1^n) + \mathcal{O}(1)]$ *and that* A *is a left-c.e. real. Then* A *is wtt-complete.*

Proof. Let $\psi(x)$ be the partial computable function of the least stage that $x \in K_s$. If a left-c.e. real A is not wtt-complete then the computation function c_A defined via $c_A(x) = \mu s[A \upharpoonright x = A_s \upharpoonright x]$ of A does not dominate ψ. Thus $\exists^\infty x[\psi(x) \downarrow > c_A(x)]$. Therefore

$$C(A \upharpoonright n) \leqslant C(n) = \mathcal{O}(1),$$

for those infinitely many n in K where $\psi(n) > c_A(n)$. □

Stephan has clarified the situation for the relationship between $\leqslant_C$ and $\leqslant_T$.

Corollary 13.9.5 (Stephan [291])**.** *Suppose that we have c.e.* α, β *with* $\alpha \leqslant_C \beta$*. Then* $\alpha \leqslant_T \beta$*.*

Proof. So we suppose that there is a c such that for all n,

$$C(\alpha \upharpoonright n) \leqslant C(\beta \upharpoonright n) + c.$$

IF $\beta \equiv_T \emptyset'$, then there is nothing to prove. So we suppose that $\beta <_T \emptyset'$. In that case, for any total β-computable function g, we know

$$\exists^\infty x[x \in \emptyset' - \emptyset'_{g(x)}].$$

Let $\psi(x)$ be the partial computable function of the least stage that $x \in \emptyset'_s$. Then there are infinitely many x with $\psi(x) \downarrow > g(x)$. So let g be the computation function of β. Then there is an infinite $D \leqslant_T \beta$ with $D \subset \emptyset'$ and $\psi(x) > g(x)$ for all $x \in D$.

For any $x \in D$ we have the following program:

$$\varphi_e(x) = \beta_{\psi(x)} \upharpoonright x.$$

For this set of x we have $C(\beta \upharpoonright x|x) \leqslant e$, and hence

$$C(\alpha \upharpoonright x|x) \leqslant e + \mathcal{O}(1).$$

Now relativizing Loveland's theorem 6.4.1, we see that $\alpha \leqslant_T \beta$. □

The situation for $\leqslant_K$ is quite different. The argument of Stephan above shows that $\alpha \leqslant_K \beta$ implies that for all $x \in D$, $K(\beta \upharpoonright x|x) \leqslant e$, and hence $K(\beta \upharpoonright x) \leqslant e + K(x) + \mathcal{O}(1)$, for this set of x. All would be sweet if the following statement, true for C was also true for K: $K(\alpha \upharpoonright x) \leqslant K(x) + \mathcal{O}(1)$ for all (a computable set of) x, implied that α is computable. Chaitin (Theorem 15.1.3) observed using a relativized form of Loveland's observation that

$$K(\alpha \upharpoonright x) \leqslant K(x) + \mathcal{O}(1) \text{ implies } \alpha \leqslant_T \emptyset'.$$

Surprisingly we cannot replace $\emptyset'$ *by* $\emptyset$ *for* K*. That is even though* α *looks identical to* ω *we cannot conclude that* α *is computable even for strongly c.e. reals* α*.*

This was proved by Solovay in his 1974 manuscript. We will look at these "K-trivial" reals in Chapter 15. These remarkable reals have enabled a fascinating interaction between randomness and computability. But we will defer this discussion until that Chapter, since their theory is sufficiently deep that they deserve a Chapter entirely to themselves.

The structure of the K- or C-degrees of c.e reals is wide open. It is currently unknown if the K-degrees form a *lattice*! We do know that both are uppersemilattices with join induced by $+$.

Theorem 13.9.6 (Downey, Hirschfeldt, Nies, Stephan [81])**.** *(i) If α, β are computably enumerable reals then the K-complexity of $\alpha + \beta$ is – up to an additive constant – the maximum of the K-complexities of α and β. In particular, $\alpha + \beta$ represents the join of α and β with respect to K-reducibility.*

(ii) The same is true of $\leqslant_C$.

Proof. We do (i), the same proof working also for (ii). Let $\gamma = \alpha + \beta$. Without loss of generality, the reals represented by α, β are in $(0, 1/2)$, so that we do not to have to care about the problem of representing digits before the decimal point. Furthermore, we have programs i, j, k which approximate α, β, γ, respectively, from below, such that at every stage and also for the limit the equation $\alpha + \beta = \gamma$ holds.

First we show that $K(\gamma \upharpoonright n) \leqslant \max\{K(\alpha \upharpoonright n), K(\beta \upharpoonright n)\} + c$ for some constant c. Fix a universal prefix-free machine U. It is sufficient to produce a prefix-free machine V that for each n computes $(\alpha + \beta) \upharpoonright n$ from some input of length up to $\max\{K(\alpha \upharpoonright n), K(\beta \upharpoonright n)\} + 2$.

The machine V receives as input eab where $a, b \in \{0, 1\}$ and $e \in \{0, 1\}^*$. The length of the input is $|e| + 2$. First V simulates $U(e)$. In the case that this simulation terminates with some output σ, let $n = |\sigma|$. Now V simulates the approximation of α and β from below until it happens that either

- $a = 0$ and $\sigma = \alpha \upharpoonright n$ or
- $a = 1$ and $\sigma = \beta \upharpoonright n$.

Let $\tilde{\alpha}, \tilde{\beta}$ be the current values of the approximations of α and β, respectively, when the above simulation is stopped. Now V outputs the first n bits of the real $\tilde{\alpha} + \tilde{\beta} + b \cdot 2^{-n}$.

In order to verify that this works, given n, let a be 0 if the approximation of β is correct on its first n bits before the one of α and let a be 1 otherwise. Let e be the shortest program for $\alpha \upharpoonright n$ in case $a = 0$ and for $\beta \upharpoonright n$ in case $a = 1$. Then $U(e)$ terminates and $|e| \leqslant \max\{K(\alpha \upharpoonright n), K(\beta \upharpoonright n)\}$. In addition, we know both values $\alpha \upharpoonright n$ and $\beta \upharpoonright n$ once $U(e)$ terminates. So $\tilde{\alpha}$ and $\tilde{\beta}$ (defined as above) are correct on their first n bits, but it might be that bits beyond the first n cause a carry to exist which is not yet known. But we can choose b to be that carry bit and have then that $V(eab) = \gamma \upharpoonright n$.

For the other direction, we construct a machine W that computes $(\alpha \upharpoonright n, \beta \upharpoonright n)$ from any input e with $U(e) = \gamma \upharpoonright n$. The way to do this is to simulate $U(e)$ and, whenever it gives an output σ, to simulate the enumerations of α, β, γ until the current approximation $\tilde{\gamma} \upharpoonright n = \sigma$. As $\tilde{\alpha} + \tilde{\beta} = \tilde{\gamma}$, it is impossible that the approximations of α, β will later change on their first n bits if $\gamma \upharpoonright n = \sigma$. So the machine W then just outputs $(\tilde{\alpha} \upharpoonright n, \tilde{\beta} \upharpoonright n)$, which is correct under the assumption that e, and therefore also σ, are correct. □

Again we note that both $\leqslant_C$ and $\leqslant_K$ are Σ^0_3 pre-orderings whose induced degree structure on the left-c.e. reals has top degree that of $[\Omega]$, the bottom includes the computable reals (and in the case of $\leqslant_C$ *is* the computable reals, and $+$ is a join. Hence, again, Theorem **??** applies to prove density.

Little else is known about the degree structure of the $\leqslant_C$ or $\leqslant_K$ degrees of left-c.e. reals.

Interestingly, and mainly through the efforts of Yu Liang and Joe Miller, we know a lot more about the degree structure of $\leqslant_K$ and $\leqslant_C$ degrees of *random* reals. More on this in Chapter 14.

13.10 K-degrees, C-degrees, and Turing degrees

The Slaman-Kučera Theorem says that $[\Omega]$ is special at least in terms of Solovay reducibility. The complete K-degree of left-c.e. reals collapses to a single Solovay degree. On the other hand, in Chapter 15 we will construct a c.e. nonzero Turing degree $\mathbf{a}$ such that the K-degrees of reals below $\mathbf{a}$ are all the same. The ultimate extension of the Slaman-Kučera Theorem would be to say that the only c.e. K-degree which collapses to a single Solovay degree is $[\Omega]$, and the only c.e. K-degree which collapses to a single Turing degree is $\mathbf{0}'$.

In Chapter 17, we see that if we look at all left-c.e. reals α of Kolmogorov complexity $\frac{1}{2}n$, that is $K(\alpha \upharpoonright n) = n + \mathcal{O}(1)$, then they are all of the same Turing degree $\mathbf{0}'$.

Recently, Merkle and Stephan [**?**] gave a simple and short (but clever) proofs about the structure of the C- and K-degrees, and their relationships with Turing degrees by using sparse sets. We will see other relationships between these measures of relative randomness and Turing degrees for *random* reals in the next Chapter, Chapter 14.

Theorem 13.10.1 (Merkle and Stephan [**?**])**.** *Suppose that* $Y \subseteq \{2^n : n \in \omega\}$*. Then any* X*,*

$$X \leqslant_K Y \text{ implies } X \leqslant_T Y \oplus \emptyset'.$$

Hence if $\emptyset' \leqslant Y$ *then* $Y \leqslant_K X$ *implies* $Y \leqslant_T X$*.*

Proof. The idea is code Y by certain sparse numbers. We define $g(n) = 2^n + \sum_{k<n} 2^k Y(2^{k+2})$. Then $g(n) \in [2^n, 2^{(n+1)} - 1]$ and we can compute $Y \upharpoonright 2^{(n+1)}$ from $g(n)$. Thus $K(Y \upharpoonright g(n)) \leqslant^+ K(g(n))$. If $X \leqslant_K Y$, we also see $K(X \upharpoonright g(n)) \leqslant^+ K(g(n))$. Thus X is (K, c) trivial on $I = \{g(n) : n \in \omega\}$, for some fixed c. By Theorem **??**, $X \leqslant I \oplus \emptyset' \leqslant_T Y \oplus \emptyset'$. $\square$

We remark that a similar proof establishes the following.

Theorem 13.10.2 (Merkle and Stephan [**?**])**.** *Let $Y \subseteq \{2^{2^n} : n \in \omega\}$. The for any X, $X \leqslant_C Y$ implies $X \leqslant_T Y$.*

Proof. The method is similar. We assume that the first two values are known. Then define $g(n) = 2^n + \sum_{k<n} 2^k Y(2^{2^{k+2}})$. Then again $g(n) \in [2^n, 2^{(n+1)} - 1]$, and hence $2^{g(n)+1} \leqslant 2^{2^{n+1}}$, for all n. Hence, give a number $m \in J_n = [2^{g(n)}, 2^{g(n)+1} - 1]$, we can fiurst compute $g(n)$ and then compute $Y(2^{2^k})$ for all k with $2^{2^{k+2}} < 2^{g(n)+1}$, from $g(n)$. Then there is a constant c such that for all n and all $m \in J_n$,

$$C(Y \upharpoonright m) \leqslant g(n) + c_1.$$

Thus $C(X \upharpoonright m) \leqslant g(n) + c_2$, for some constant c_2. Now let T be the set of strings $w \in J_n$ for some n and w has an extension w' of length $\max J_{n+1}$ such that for all prefixes u of w',

$$u \in J_n \text{ implies } C(u) < g(n) + c_2.$$

Then $T \leqslant_T Y$ is a tree and $X \in [T]$. Furthermore for each n there are at most $2^{g(n)}$ numbers in J_n. Hence for any n, there is a number $i_n \in J_n$ such that for some c_3,

$$g(n) \leqslant C(i_n) + c_3.$$

Let $I = \{i_n : n \in \omega\}$, and let $c_4 = c_3 + c_2$. We see that for all n,

$$C(X \upharpoonright i_n) \leqslant C(i_n) + c_4.$$

Thus X is (C, c_4) trivail on I, and hence $X \leqslant_T I$. But more importantly, since the tree of extendably (C, c_4) trivial strings has constant width, so too must T. Hence as $T \leqslant_T Y$, we see that $X \leqslant_T Y$. $\square$

An immediate corollary to this last result is the following.

Corollary 13.10.3 (Merkle and Stephan [**?**])**.** *There exist a minimal pair of C degrees of computably enumerable sets.*

Proof. Choose $A, B \subseteq \{2^{2^n} : n \in \omega\}$ whose Turing degrees form a minimal pair and apply Theorem 13.10.2. $\square$

Another corollary examines the quantity of F-degrees within a Turing degree.

Corollary 13.10.4 (Merkle and Stephan [**?**])**.** *A C-degree contains at most one Turing degree.*

Proof. Suppose that A and B are sets in a C-degree $\mathbf{a}$. Consider $\widehat{A} = \{2^{2^n} : n \in A\}$ and $\widehat{B} = \{2^{2^n} : n \in B\}$. By Theorem 13.10.2, we would have $\widehat{A} \equiv_T \widehat{B}$ and hence $A \equiv_T B$. □

We remark that not all C-degrees can contain Turing degrees. Recall that A is *complex* iff there is an order h with $C(A \upharpoonright h(n)) >^+ n$ for all n (or, equivalently, an order $\widehat{h}$ with $K(A \upharpoonright \widehat{h}(n)) >^+ n$ for all n.)

Theorem 13.10.5 (Merkle and Stephan [**?**])**.** *Suppose that A is complex. Then neither the C-degree nor the K-degree of A contains a Turing degree.*

Proof. (We do the case for C, the K-case being more or less identical.) Fix h as above. Let $B \equiv_C A$. Let $D = \{h(4n) : n \in B\}$. Then $C(B \upharpoonright h(4n)) > 3n$, and $C(D \upharpoonright h(4n)) < 2n$ for almost all n, yet $D \equiv_T B$. □

For the K-degrees, we can apply similar methods to get minimal pairs. We will say that a real A is *K-trivial* iff for all n, $K(A \upharpoonright n) \leqslant^+ K(n)$. K-trivials are a very important class and will be studied more fully in Chapter 15. The form the **0**-degree for $\leqslant_K$.

Theorem 13.10.6 (Merkle and Stephan [**?**])**.** *Suppose that Y is not K-trivial and $Y \leqslant_K X$. Then there is a function $f \leqslant_T Y \oplus \emptyset'$ such that, for each n, the set X has at least one element between n and $f(n)$.*

Proof. Let c be a constant with $Y \leqslant_K X$ via c. Let $f(n)$ be the least number $m > n$ such that for all subsets $D \subset \{0, \dots, n\}$, $K(D \upharpoonright m) + c < K(Y \upharpoonright m)$. This exists as Y is not K-trivial and f is computable from $\emptyset'$ and Y. As $Y \leqslant X$, X must differ below m from all subsets $D \subset \{0, \dots, n\}$, and the result follows. □

Theorem 13.10.7 (Merkle and Stephan [**?**])**.** *Suppose that A and B are $\emptyset'$-hyperimmune sets, forming a Turing minmal pair relative to $\emptyset'$. Then the K-degrees of A and B form a minimal pair. Hence there are Σ^0_2 minimal pairs of K-degrees.*

Proof. Choose A and B as above. Without loss of generality we can assume that $A, B \subseteq \{2^n : n \in \omega\}$. Let $Y \leqslant_K A, B$. By Theorem 13.10.1, $Y \leqslant_T A \oplus \emptyset', B \oplus \emptyset'$. Thus $Y \leqslant_T \emptyset'$ as A and B form a minimal pair above $\emptyset'$. But by Theorem 13.10.6, if Y is not K-trivial, then A and B could not be hyperimune relative to $\emptyset'$, and the result follows. □

Theorem 13.10.7 improves a basic result of Csima and Montalbán [**?**] who constructed a minimal pair of K-degrees. The method of Csima and Montalbán is interesting and informative in its own right, and will be dealt with in Chapter 15. In particular, their method uses a gap behaviour for the K-trivials.

We remark that using similar techniques, Merkle and Stephan have constructed nontrivial branching degrees in the C and K degrees.

13.11 A Minimal C-degree

Theorem 13.11.1 (Merkle and Stephan [?]). *There exists a minimal C-degree.*

Proof. We use perfect set forcing where the branches of the trees are subsets of $\{2^{2^n} : n \in \omega\}$. We set T_0 to be the computable tree of branches of this form. Haning defined T_e we define T_{e+1} to be the computable subtree which forces Φ_e^A to be partial for all branches A of T_{e+1}, or we will force Φ_e^A to be computable for all A on T_{e+1}, *or* we will construct a *very sparse* splitting tree for T_{e+1}. In particular, we will ask that for all e, and for all branches A on T_{e+1}, and all x, there is at most one branching node σ on T_{e+1} with $x \leqslant |\sigma| \leqslant \varphi_e(x)$, where $\varphi_e^A(x)$, where $\varphi_e^A(x)$ denotes the use of the computation $\Phi^A(x)$.

Then the usual arguments show that $A = \cap_e T_e$ is noncomputable and has minimal Turing degree. Since $A \subset \{2^{2^n} : n \in \omega\}$, $B \leqslant_C A$ implies $B \leqslant_T A$. Choose $B \leqslant_C A$ with B noncomputable. Then $B \leqslant_T A$ via some Φ_e and lies in the range of T_{e+1}, with $A \leqslant_T B$ as give by the usual inductive procedure for minimal degrees. *However*, we note that by the φ_e^A sparseness of A, there is at most one (largest) place x such that

(i) $B \upharpoonright n$ computes $A \upharpoonright z$ for all $z \leqslant x$, and

(ii) $A \upharpoonright n$ is at worst one of two extensions (computably determined by the construction) of length n on T_{e+1} of $A \upharpoonright x$.

hence $C(A \upharpoonright n) \leqslant^+ C(B \upharpoonright n)$. □

Using more intricate methods, it is possible to prove the following.

Theorem 13.11.2. *Every c.e. C-degree bounds a minimal C-degree. (and a minimal rK-degree.)*

It is unknown whether there is a minimal K-degree.

13.12 Density and Splittings

Lemma 13.12.1. *Let $\alpha \nleqslant_{\mathrm{S}} \beta$ be left-c.e. reals. The following hold for all total computable functions f and all $k \in \omega$.*

1. *For each $n \in \omega$ there is an $s \in \omega$ such that either*
   (a) $\alpha_t - \alpha_{f(n)} < k(\beta_t - \beta_n)$ for all $t > s$ or
   (b) $\alpha_t - \alpha_{f(n)} > k(\beta_t - \beta_n)$ for all $t > s$.
2. *There are infinitely many $n \in \omega$ for which there is an $s \in \omega$ such that $\alpha_t - \alpha_{f(n)} > k(\beta_t - \beta_n)$ for all $t > s$.*

Proof. If there are infinitely many $t \in \mathbb{N}$ such that $\alpha_t - \alpha_{f(n)} \leqslant k(\beta_t - \beta_n)$ and infinitely many $t \in \mathbb{N}$ such that $\alpha_t - \alpha_{f(n)} \geqslant k(\beta_t - \beta_n)$ then

$$\alpha - \alpha_{f(n)} = \lim_t \alpha_t - \alpha_{f(n)} = \lim_t k(\beta_t - \beta_n) = k(\beta - \beta_n),$$

which implies that $\alpha \equiv_{\mathrm{S}} \beta$.

If there are infinitely many $t \in \mathbb{N}$ such that $\alpha_t - \alpha_{f(n)} \leqslant k(\beta_t - \beta_n)$ then

$$\alpha - \alpha_{f(n)} = \lim_t \alpha_t - \alpha_{f(n)} \leqslant \lim_t k(\beta_t - \beta_n) = k(\beta - \beta_n).$$

So if this happens for all but finitely many n then $\alpha \leqslant_{\mathrm{S}} \beta$. (The finitely many n for which $\alpha - \alpha_{f(n)} > k(\beta - \beta_n)$ can be brought into line by increasing the constant k.) $\square$

Lemma 13.12.2. *Let $\beta \leqslant_{\mathrm{S}} \alpha$ be left-c.e. reals and let $\alpha_0, \alpha_1, \ldots$ be a computable increasing sequence of rationals converging to α. There is a computable increasing sequence $\widehat{\beta}_0, \widehat{\beta}_1, \ldots$ of rationals converging to β such that for some constant c and all $s \in \omega$,*

$$\widehat{\beta}_s - \widehat{\beta}_{s-1} < c(\alpha_s - \alpha_{s-1}).$$

Let C_i be the ith left-c.e. real.

Definition 13.12.3. Let r be a reducibility on the left-c.e. reals. We say that r is Σ^0_3 if there is a total computable 5-ary function Φ such that for all $a, b \in \omega$, we have $C_a \leqslant_r C_b$ iff $\exists k \, \forall m \, \exists n \, \Phi(a, b, k, m, n)$.

The reducibility r is *standard* if r is Σ^0_3, every computable real is r-reducible to any given left-c.e. real, and addition is a join in the r-degrees of left-c.e. reals.

Theorem 13.12.4. *Let r be a standard reducibility on the left-c.e. reals. Let $\gamma <_r \alpha <_{\mathrm{S}} \Omega$ be left-c.e. reals. There are left-c.e. reals β^0 and β^1 such that $\gamma <_r \beta^0, \beta^1 <_r \alpha$ and $\beta^0 + \beta^1 = \alpha$.*

Proof. Let a be an index for α and let Φ be as in Definition 13.12.3.

We want to build β^0 and β^1 so that $\gamma \leqslant_r \beta^0, \beta^1 \leqslant_r \alpha$, $\beta^0 + \beta^1 = \alpha$, and the following requirement is satisfied for each $e \in \omega$ and $i < 2$:

$$R_{i,e} : \exists n \, \forall m \, \neg\Phi(a, b^i, e, n, m),$$

where b^i is an index for β^i given by the Recursion Theorem.

By Lemma 13.12.2 and the fact that $\gamma/c \equiv_r \gamma$ for any rational c, we may assume without loss of generality that $2(\gamma_s - \gamma_{s-1}) \leqslant \alpha_s - \alpha_{s-1}$ for each $s \in \omega$. (Recall our convention that $\mu_0 - \mu_{-1} = \mu_0$ for any left-c.e. real μ.)

In the absence of requirements of the form $R_{1-i,e}$, it is easy to satisfy simultaneously all requirements of the form $R_{i,e}$: for each $s \in \omega$, simply let $\beta^i_s = \gamma_s$ and $\beta^{1-i}_s = \alpha_s - \gamma_s$. In the presence of requirements of the form $R_{1-i,e,k}$, however, we cannot afford to be quite so cavalier in our treatment of β^{1-i}; enough of α has to be kept out of β^{1-i} to guarantee that β^{1-i} is not r-above α.

Most of the essential features of our construction are already present in the case of two requirements $R_{i,e}$ and $R_{1-i,e'}$, which we now discuss. We assume that $R_{i,e}$ has priority over $R_{1-i,e'}$. We will think of the β^j as being built by adding amounts to them in stages. Thus β^j_s will be the total amount added to β^j by the end of stage s. At each stage s we begin by adding $\gamma_s - \gamma_{s-1}$ to the current value of each β^j; in the limit, this ensures that $\beta^j \geqslant_r \gamma$.

We will say that $R_{i,e}$ is satisfied through n at stage s if $\forall m < s \, \neg\Phi(a, b^i, e, n, m)$. The strategy for $R_{i,e}$ is to act whenever either it is not currently satisfied or the least number through which it is satisfied changes. Whenever this happens, $R_{i,e}$ initializes $R_{1-i,e'}$, which means that the amount of $\alpha - 2\gamma$ that $R_{1-i,e'}$ is allowed to funnel into β^i is reduced. More specifically, once $R_{1-i,e'}$ has been initialized for the mth time, the total amount that it is thenceforth allowed to put into β^i is reduced to 2^{-m}.

The above strategy guarantees that if $R_{1-i,e'}$ is initialized infinitely often then the amount put into β^i by $R_{1-i,e'}$ (which in this case is all that is put into β^i except for the coding of γ) adds up to a computable real. In other words, $\beta^i \equiv_r \gamma <_r \alpha$. But this means that there is a stage s after which $R_{i,e}$ is always satisfied and the least number through which it is satisfied does not change. So we conclude that $R_{1-i,e'}$ is initialized only finitely often, and that $R_{i,e}$ is eventually permanently satisfied.

This leaves us with the problem of designing a strategy for $R_{1-i,e'}$ that respects the strategy for $R_{i,e}$. The problem is one of timing. To simplify notation, let $\widehat{\alpha} = \alpha - 2\gamma$ and $\widehat{\alpha}_s = \alpha_s - 2\gamma_s$. Since $R_{1-i,e'}$ is initialized only finitely often, there is a certain amount 2^{-m} that it is allowed to put into β^i after the last time it is initialized. Thus if $R_{1-i,e'}$ waits until a stage s such that $\widehat{\alpha} - \widehat{\alpha}_s < 2^{-m}$, adding nothing to β^i until such a stage is reached, then from that point on it can put all of $\widehat{\alpha} - \widehat{\alpha}_s$ into β^i, which of course guarantees its success. The problem is that, in the general construction, a strategy working with a quota 2^{-m} cannot effectively find an s such that $\widehat{\alpha} - \widehat{\alpha}_s < 2^{-m}$. If it uses up its quota too soon, it may find itself unsatisfied and unable to do anything about it.

The key to solving this problem (and the reason for the hypothesis that $\alpha <_{\mathrm{S}} \Omega$) is the observation that, since the sequence $\Omega_0, \Omega_1, \dots$ converges much more slowly than the sequence $\widehat{\alpha}_0, \widehat{\alpha}_1, \dots$, Ω can be used to modulate the amount that $R_{1-i,e'}$ puts into β^i. More specifically, at a stage s, if $R_{1-i,e'}$'s current quota is 2^{-m} then it puts into β^i as much of $\widehat{\alpha}_s - \widehat{\alpha}_{s-1}$ as possible, subject to the constraint that the total amount put into β^i by $R_{1-i,e'}$ since the last stage before stage s at which $R_{1-i,e'}$ was initialized must not exceed $2^{-m}\Omega_s$. As we will see below, the fact that $\Omega >_{\mathrm{S}} \alpha$ implies that there is a stage v after which $R_{1-i,e'}$ is allowed to put in all of $\widehat{\alpha} - \widehat{\alpha}_v$ into β^i.

In general, at a given stage s there will be several requirements, each with a certain amount that it wants (and is allowed) to direct into one of the

β^j. We will work backwards, starting with the weakest priority requirement that we are currently considering. This requirement will be allowed to direct as much of $\widehat{\alpha}_s - \widehat{\alpha}_{s-1}$ as it wants (subject to its current quota, of course). If any of $\widehat{\alpha}_s - \widehat{\alpha}_{s-1}$ is left then the next weakest priority strategy will be allowed to act, and so on up the line.

We now proceed with the full construction. We say that $R_{i,e}$ has stronger priority than $R_{i',e'}$ if $2e + i < 2e' + i'$.

We say that $R_{i,e}$ is satisfied through n at stage s if

$$\forall m < s \, \neg\Phi(a, b^i, e, n, m).$$

Let $n_s^{i,e}$ be the least n through which $R_{i,e}$ is satisfied at stage s, if such an n exists, and let $n_s^{i,e} = \infty$ otherwise.

We say that $R_{i,e}$ requires attention at stage s if either $n_s^{i,e} = \infty$ or $n_s^{i,e} \neq n_{s-1}^{i,e}$.

If $R_{i,e}$ requires attention at stage s then we say that each requirement of weaker priority than $R_{i,e}$ is initialized at stage s.

Each requirement $R_{i,e}$ has associated with it a left-c.e. real $\tau^{i,e}$, which records the amount put into β^{1-i} for the sake of $R_{i,e}$.

We decide how to distribute $\delta = \alpha_s - \alpha_{s-1}$ between β^0 and β^1 at stage s as follows.

1. Let $j = s$ and $\varepsilon = 2(\gamma_s - \gamma_{s-1})$, and add $\gamma_s - \gamma_{s-1}$ to the current value of each β^i.

2. Let $i < 2$ and $e \in \omega$ be such that $2e + i = j$. Let m be the number of times $R_{i,e}$ has been initialized and let t be the last stage at which $R_{i,e}$ was initialized. Let

$$\zeta = \min(\delta - \varepsilon, 2^{-(j+m)}\Omega_s - (\tau_{s-1}^{i,e} - \tau_t^{i,e})).$$

   (It is not hard to check that ζ is non-negative.) Add ζ to ε and to the current values of $\tau^{i,e}$ and β^{1-i}.

3. If $\varepsilon = \delta$ or $j = 0$ then add $\delta - \varepsilon$ to the current value of β^0 and end the stage. Otherwise, decrease j by one and go to step 2.

This completes the construction. Clearly, $\gamma \leqslant_r \beta^0, \beta^1 \leqslant_r \alpha$ and $\beta^0 + \beta^1 = \alpha$.

We now show by induction that each requirement initializes requirements of weaker priority only finitely often and is eventually satisfied. Assume by induction that $R_{i,e}$ is initialized only finitely often. Let $j = 2e + i$, let m be the number of times $R_{i,e}$ is initialized, and let t be the last stage at which $R_{i,e}$ is initialized. The following are clearly equivalent:

1. $R_{i,e}$ is satisfied,

2. $\lim_s n_s^{i,e}$ exists and is finite, and

3. $R_{i,e}$ eventually stops requiring attention.

Assume for a contradiction that $R_{i,e}$ requires attention infinitely often. Since $\Omega \nleqslant_S \alpha$, part 2 of Lemma 13.12.1 implies that there are $v > u > t$ such that for all $w > v$ we have $2^{-(j+m)}(\Omega_w - \Omega_u) > \alpha_w - \alpha_u$. Furthermore, by the way the amount ζ added to $\tau^{i,e}$ at a given stage is defined in step 2 of the construction, $\tau^{i,e}_u - \tau^{i,e}_t \leqslant 2^{-(j+m)}\Omega_u$ and $\tau^{i,e}_{w-1} - \tau^{i,e}_u \leqslant \alpha_{w-1} - \alpha_u$. Thus for all $w > v$,

$$\alpha_w - \alpha_{w-1} = \alpha_w - \alpha_u - (\alpha_{w-1} - \alpha_u) <$$
$$2^{-(j+m)}(\Omega_w - \Omega_u) - (\alpha_{w-1} - \alpha_u) = 2^{-(j+m)}\Omega_w - (2^{-(j+m)}\Omega_u + \alpha_{w-1} - \alpha_u) \leqslant$$
$$2^{-(j+m)}\Omega_w - (\tau^{i,e}_u - \tau^{i,e}_t + \tau^{i,e}_{w-1} - \tau^{i,e}_u) = 2^{-(j+m)}\Omega_w - (\tau^{i,e}_{w-1} - \tau^{i,e}_t).$$

From this we conclude that, after stage v, the reverse recursion performed at each stage never gets past j, and hence everything put into β^i after stage v is put in either to code γ or for the sake of requirements of weaker priority than $R_{i,e}$.

Let τ be the sum of all $\tau^{1-i,e'}$ such that $R_{1-i,e'}$ has weaker priority than $R_{i,e}$. Let $s_l > t$ be the lth stage at which $R_{i,e}$ requires attention. If $R_{1-i,e'}$ is the pth requirement on the priority list and $p > j$ then $\tau^{i',e'} - \tau^{i',e'}_{s_l} \leqslant 2^{-(p+l)}\Omega$. Thus

$$\tau - \tau_{s_l} \leqslant \sum_{p \geqslant 1} 2^{-(p+l)}\Omega = 2^{-l}\Omega \leqslant 2^{-l},$$

and hence τ is computable.

Putting together the results of the previous two paragraphs, we see that $\beta^i \leqslant_r \gamma$. Since $\alpha \nleqslant_r \gamma$, this means that $\alpha \nleqslant_r \beta^i$. It now follows that there is an $n \in \omega$ such that $R_{i,e}$ is eventually permanently satisfied through n, and such that $R_{i,e}$ is eventually never satisfied through any $n' < n$. Thus $\lim_s n^{i,e}_s$ exists and is finite, and hence $R_{i,e}$ is satisfied and eventually stops requiring attention. □

Theorem 13.12.5. *Let r be a standard reducibility on the left-c.e. reals. Let $\gamma <_r \Omega$ be a left-c.e. real. There is a left-c.e. real β such that $\gamma <_r \beta <_r \Omega$.*

Proof. Let a and c be indices for γ and Ω, respectively, and let Φ be as in Definition 13.12.3.

We want to build $\beta \geqslant_r \gamma$ to satisfy the following requirements for each $e \in \omega$:

$$R_e : \Phi_e \text{ total } \Rightarrow \exists n \, \forall m \, \neg\Phi_(b, a, e, n, m)$$

and

$$S_e : \Phi_e \text{ total } \Rightarrow \exists n \, \forall m \, \neg\Phi_(c, b, e, n, m)$$

where b is an index for β given by the Recursion Theorem.

As in the previous proof, the analysis of an appropriate two-strategy case will be enough to outline the essentials of the full construction. Let us consider the strategies S_e and $R_{e'}$, the former having priority over the latter.

The strategy for S_e is basically to make β look like γ. At each point of the construction, $R_{e'}$ has a certain fraction of Ω that it is allowed to put into β. (This is in addition to the coding of γ into β, of course.) We will say that S_e is satisfied through n at stage s if $\forall m < s\, \neg\Phi(c, b, e, n, m)$. Whenever either it is not currently satisfied or the least number through which it is satisfied changes, S_e initializes $R_{e'}$, which means that the fraction of Ω that $R_{e'}$ is allowed to put into β is reduced.

As in the previous proof, if S_e is not eventually permanently satisfied through some n then the amount put into β by $R_{e'}$ is computable, and hence $\beta \equiv_r \gamma$. But, as before, this implies that there is a stage after which S_e is permanently satisfied through some n and never again satisfied through any $n' < n$. Once this stage has been reached, $R_{e'}$ is free to code a fixed fraction of Ω into β, and hence it too succeeds.

We now proceed with the full construction. We say that a requirement X_e has stronger priority than a requirement $Y_{e'}$ if either $e < e'$ or $e = e'$, $X = R$, and $Y = S$.

We say that R_e is satisfied through n at stage s if

$$\forall m < s\, \neg\Phi(b, a, e, n, m).$$

We say that S_e is satisfied through n at stage s if

$$\forall m < s\, \neg\Phi(c, b, e, n, m).$$

For a requirement X_e, let $n_s^{X_e}$ be the least n through which X_e is satisfied at stage s, if such an n exists, and let $n_s^{X_e} = \infty$ otherwise.

We say that the requirement X_e requires attention at stage s if either $n_s^{X_e} = \infty$ or $n_s^{X_e} \neq n_{s-1}^{X_e}$.

At stage s, proceed as follows. First add $\gamma_s - \gamma_{s-1}$ to the current value of β. If no requirement requires attention at stage s then end the stage. Otherwise, let X_e be the strongest priority requirement requiring attention at stage s. We say that X_e acts at stage s. If $X = S$ then initialize all weaker priority requirements and end the stage. If $X = R$ then let m be the number of times that R_e has been initialized. If s is the first stage at which R_e acts after the last time it was initialized then let t be the last stage at which R_e was initialized, and otherwise let t be the last stage at which R_e acted. Add $2^{-(e+m)}(\Omega_s - \Omega_t)$ to the current value of β and end the stage.

This completes the construction. Since β is bounded by $\gamma + \sum_{i \geqslant 0} 2^{-i}\Omega = \gamma + 2\Omega$, it is a well-defined left-c.e. real. Furthermore, $\gamma \leqslant_r \beta$.

We now show by induction that each requirement initializes requirements of weaker priority only finitely often and is eventually satisfied. Assume by induction that there is a stage u such that no requirement of stronger priority than X_e requires attention after stage u. The following are clearly equivalent:

1. X_e is satisfied,
2. $\lim_s n_s^{X_e}$ exists and is finite,
3. X_e eventually stops requiring attention, and
4. X_e acts only finitely often.

First suppose that $X = R$. Let m be the number of times that R_e is initialized. (Since R_e is not initialized at any stage after stage u, this number is finite.) Suppose that R_e acts infinitely often. Then the total amount added to β for the sake of R_e is $2^{-(e+m)}\Omega$, and hence $\beta \equiv_r 2^{-(j+m)}\Omega \equiv_r \Omega \nleqslant_r \gamma$. It now follows from Lemma 13.12.1 that there is an $n \in \omega$ such that R_e is eventually permanently satisfied through n, and such that R_e is eventually never satisfied through $n' < n$. Thus $\lim_s n_s^{R_e}$ exists and is finite, and hence R_e is satisfied and eventually stops requiring attention.

Now suppose that $X = S$ and S_e acts infinitely often. If $v > u$ is the mth stage at which S_e acts then the total amount added to β after stage v for purposes other than coding γ is bounded by $\sum_{i \geqslant 0} 2^{-(i+m)}\Omega < 2^{-m+1}$. This means that $\beta \equiv_r \gamma \ngeqslant_r \Omega$. It now follows from Lemma 13.12.1 that there is an $n \in \omega$ such that S_e is eventually permanently satisfied through n, and such that S_e is eventually never satisfied through $n' < n$. Thus $\lim_s n_s^{S_e}$ exists and is finite, and hence S_e is satisfied and eventually stops requiring attention. $\square$

Combining Theorems 13.12.4 and 13.12.5, we have the following result.

Theorem 13.12.6. *Let r be a standard reducibility on the left-c.e. reals that is at least as strong as Solovay reducibility. Then the r-degrees of left-c.e. reals are dense. In particular, the following reducibilities are standard: $\leqslant_S$, $\leqslant_{rK}$, $\leqslant_C$, and $\leqslant_K$.*

DENIS can you remember the proof that $+$ is a join for $\leqslant_{sm}$?

One notable reducibility missing from the list above is $\leqslant_{Km}$, monotone reducibility. Certainly this is a Σ_3^0 reducibility where $[\Omega]$ is the top degree and the trivials are the computable sets. *However*, it is not known if $+$ induces a join on this degree structure.

Question 13.12.7. *Does $+$ induce a join on the monotone degrees of left c.e. reals?*

We conjecture that the answer is no. In spite of this lack of knowledge, there is still a downward density theorem.

Theorem 13.12.8 (Calhoun [**?**])**.** *The monotone degrees of left c.e. reals are downward dense, meaning that if* $\mathbf{b}$ *is a nonzero Km degree of a left c.e. real then there is a Km degrees* $\mathbf{a}$ *of a left c.e. real with* $\mathbf{b} > \mathbf{a} > \mathbf{0}$..

The proof uses the following useful lemma says we can use simple permitting..

Lemma 13.12.9 (Calhoun [?])**.** *Suppose that* $A = \lim_s A_s$*, and* $B = \lim_s B_s$ *are montonic approximations to left c.e. reals* A *and* B *and* f *is a computable function such that for all* s, n*,*

$$B_s \upharpoonright n = B \upharpoonright n \text{ implies } A_{f(s)} \upharpoonright n = A \upharpoonright n.$$

Then $A \leqslant_{Km} B$*.*

Proof. By speeding things up, we will suppose that $f(s) = s$. Let U denote the universal monotone machine. We build a montone machine M. The point is that whenever $U(\sigma) = B_s \upharpoonright n$, we can set $M(\sigma) = A_s \upharpoonright n$. The given condition guarantees that this definition is monotone. □

Proof. (of Theorem 13.12.8) The argument is a finite injury one. We are given $B = \lim_s B_s$. We keep $A \leqslant_{Km} B$ by Lemma 13.12.9 and simple permitting. We must meet the requirements

$$R_{2e+1} : \overline{A} \neq W_e$$

$$R_{2e} : B \not\leqslant_{Km} A \text{ with constant } e.$$

The R_{2e+1} are enough to make A nontrivial as the trivial Km degree consists of the computable sets. Associates with the R_j are movable markers $n(j,s)$, with $n(-1,s) = 0$ for all s. We will be meeting $R)j$ between $n(j-1,s)$ and $n(j,s)$. We will show that $\lim_s j(j,s) = n(j)$ exists.

To meet R_0 (and more generally R_{2e}) we will allow R_0 to assert control of various locations of A. If R_e asserts control of position n at stage s, we will ensure, with the appropriate priority, that $A_t(n) = A_s(n)$ for all $t > s$. At stage s, R_0 will have control of $A \upharpoonright [n(-1,s), n(0,s)]$. At stage $s+1$ if we see $Km(B_s \upharpoonright n(0,s)) \leqslant Km(A_s \upharpoonright n(0,s))[s]$, then we allow R_0 to assert control of the next position by setting $n(0,s+1) = n(0,s)+1$. Notice that this can happen only finitely often lest B be computable.

Meeting R_1 we use a simple permitting argument. Once R_1 has priority, if it has control of position n, when we see, and $\overline{A}_s = W_{e,s} \upharpoonright n(1,s)$ we will similarly set $n(1,s+1) = n(1,s+)+1$, and if ever B_t permits $n(1,t)$ we can make a disagreement in the usual way changing $A_t(n(1,s))$. □

Question 13.12.10. *Are the* Km *degrees of left c.e. reals dense?*

13.13 Schnorr reducibility

Once we have a machine characterization of Schnorr randomness, we can define a calibrating pre-ordering along the lines of $\leqslant_K$.

Definition 13.13.1 (Downey and Griffiths [72])**.** We say a real α is *Schnorr reducible* to β

$$\alpha \leqslant_{Sch} \beta,$$

iff for all computable machines M, there is a constant c and a computable machine $\widehat{M}$ such that for all n,

$$K_M(\beta \upharpoonright n) \geqslant K_{\widehat{M}}(\alpha \upharpoonright n) - c.$$

(Here we regard the left hand side as infinite if $\beta \notin \mathrm{ra}(M)$.)

We remark that actually we could have apparently two varieties of Schnorr reducibility a *uniform* version where $M \mapsto \widehat{M}$ is computable, and a *nonuniform* version where this map is arbitrary. We do not explore this here.

Clearly, if α is Schnorr random, and $\alpha \leqslant_{Sch} \beta$, then β is random. Little is known. We finish this section with some easy observations from Downey, Griffiths, LaForte [74]

We would like to know if $\leqslant_K$ implies $\leqslant_{Sch}$. The answer is no in quite a strong way.

Theorem 13.13.2 (Downey, Griffiths, LaForte [74])**.** *There are c.e. sets A and B such that $B \leqslant_{\mathrm{sw}} A$, but $B \not\leqslant_{Sch} A$.*

Proof. By Observation 10.3.8, we need only consider prefix-free machines M_e such that $\mu(\mathrm{dom}(\mathrm{M}_e)) = 1$, since any computable machine is equivalent to such a one. We therefore build a computable machine M, and c.e. A, B, to satisfy the requirements

$$R_e: \text{ if } \mu(\mathrm{M}_e) = 1, \text{ then } (\exists n) K_M(A \upharpoonright n) < K_{\mathrm{M}_e}(B \upharpoonright n) - e.$$

To satisfy requirement R_e, we will set aside a block of numbers $\{n, n{+}1, ..., n{+}d\}$ where d is some number greater than 2^{e+2}. Note that $2 < 2^{e+2}$, so that $d^2 < 2^{d+2} - 2$. Of the numbers in the block $\{n, ..., n{+}d\}$, we will allow no $n{+}j$ for $j > 0$ to ever enter A, but we may possibly put n itself into A. Thus we enumerate 2 axioms of the form $\langle 2{+}\log d, \tau\rangle$, one for each of the two possibilities for $\tau = A \upharpoonright n + j{+}1$ with $j \leqslant d$. This adds $2(d{+}1)2^{-2-\log d} \leqslant 2^{-1} + 2^{-1-\log d} < 1$ to $\mu(\mathrm{M})$. We now wait for a stage s such that $1 - \mu(\mathrm{M}_e)[s] < 2^{-e-2-\log d}$. Since $\mu(\mathrm{M}_e) \leqslant 1$, there can be at most $d \cdot 2^{e+2} \leqslant d^2 < 2^{d+2} - 2$ strings of length less than or equal to $e{+}2{+}\log d$ on which M_e converges. However, the number of axioms required by M_e to cover all possibilities of members of the block $\{n, \ldots, n{+}d\}$ being in or out of B, for the $d{+}1$ strings, $B \upharpoonright n, \ldots B \upharpoonright (n{+}d)$ is $2^1 + 2^2 + ... + 2^{d+1} = 2^{d+2} - 2$. Hence, at least one possibility is not in the range of M_e restricted to strings in its domain of length less than or equal to $e{+}2{+}\log d$. At this point, we choose such a combination of elements of $\{n, \ldots, n + d\}$ and enumerate them into $B[s{+}1]$, simultaneously enumerating n into $A[s{+}1]$. Any new axioms for M_e must cause convergence on strings of length greater than $e{+}2{+}\log d$. Since $K_{\mathrm{M}}(n{+}j) = 2{+}\log d$ for every $j \leqslant d$, the requirement is satisfied. Also, the membership of all elements of $\{n, \ldots, n + d\}$ in B can be calculated just by checking whether or not $n \in A$, so $B \leqslant_{\mathrm{sw}} A$. Not also that $\mu(\mathrm{M})$ is computable, since the enumeration of its axioms do not depend on waiting for any condition to be satisfied.

We can combine all strategies by assigning them intervals $\{x_0, ..., x_0+d_0\}$, $\{x_1, ..., x_1+d_1\}$, $\{x_2, \ldots, x_2+d_2\}$, and so on, where $x_0 = 0$ and $x_{i+1} = x_i + d_i + 1$. Since for each i, any combination of the values x_j for $j < i$ could show up in A, we must in general use 2^{i+1} axioms of the form $\langle m, \tau\rangle$, 2^i for each of the two possibilities for $\tau = A \upharpoonright x_i+j+1$ with $j \leqslant d_i$. This means we must enumerate axioms of the form $\langle 2+2i+\log d_i, \tau\rangle$ into M. In this case, the axioms enumerated into M for the sake of requirement R_i will add exactly

$$2^i(d_i+1)2^{-2-2i-\log d_i} \leqslant 2^{-i-2} + 2^{-i-2-\log d_i} \leqslant 2 \cdot 2^{-i-2} < 2^{-i-1}$$

to the measure of M, so that $\mu(\mathrm{M}) = \sum_{i \geqslant 0} 2^{-i-1} = 1$, as required for a computable machine. In order to satisfy the requirement, we can therefore choose d_i to be the least number greater than 2^{2+3i} Note again that $d_i^2 < 2^{d_i+2} - 2$, since $d_i \geqslant 2$. Then there can be at most $d \cdot 2^{2+3i} \leqslant d^2 < 2^{d_i+2} - 2$ strings of length less than or equal to $2+3i+\log d_i$ on which M_e converges. We take action for R_i at the first stage s such that $1 - \mu(\mathrm{M}_i)[s] < 2^{-2-3i-\log d_i}$, enumerating the elements of an appropriate subset of $\{x_i, ..., x_i+d_i\}$ into $B[s+1]$ and enumerating x_i into $A[s+1$. This satisfies requirement R_i permanently, which suffices to prove the result. $\square$

Recall that A is truth-table reducible to B ($A \leqslant_{tt} B$) if and only if $A \leqslant_{\mathrm{wtt}} B$ via a reduction Γ such that $\Gamma(\sigma, n)\downarrow$ for all $\sigma \in 2^{<\omega}$ and $n \in \mathbb{N}$. It turns out that tt-reducibility is related to Schnorr reducibility somewhat as wtt-reducibility is to K reducibility. This is not surprising, since the essential difference between a tt reduction and an ordinary wtt reduction is that the former has a computable domain, and this is what distinguishes a computable machine from an ordinary prefix-free machine.

To imply Schnorr reducibility, we will need to restrict the search in the domain of the reduction in the same way that *sw* refined *wtt*. This is the intuition behind the following definition.

Definition 13.13.3 (Downey, Griffiths, LaForte [74]). A is *strongly truth-table reducible* to B (written $A \leqslant_{\mathrm{st}} B$) if and only if $A \leqslant_{tt} B$ via a truth table reduction Γ with a constant c such that for all n, the use $\gamma(n) \leqslant n+c$.

Theorem 13.13.4. *If $A \leqslant_{st} B$, then $A \leqslant_{Sch} B$.*

Proof. Let $A \leqslant_{\mathrm{st}} B$ via some st reduction Γ with use $\gamma(n)$ bounded by $n+c$. Let $\zeta_0, \ldots, \zeta_{2^c-1}$ be the 2^c different elements of $2^{<\omega}$ of length c. Let M be any computable prefix-free machine. For every σ and τ such that $\mathrm{M}(\sigma) = \tau$, add $\langle |\sigma| + c, \Gamma^{\tau^\frown\langle\zeta\rangle_j} \upharpoonright |\tau|\rangle$ for each $j < 2^c$ to a c.e. set, thereby defining a machine M' via the Kraft-Chaitin Theorem. Since Γ is a tt-reduction, all these computations converge, and so this adds $2^c \cdot 2^{-|\sigma|-c} = 2^{-|\sigma|}$ to $\mu(\mathrm{M}')$. Hence, $\mu(\mathrm{M}') = \mu(\mathrm{M})$, making M' a computable machine. Then, if $\mathrm{M}(\sigma) = B \upharpoonright n$, we have, for some string σ' of length $|\sigma|+c$, $\mathrm{M}'(\sigma') = (\Gamma^{B\upharpoonright(n+c)} \upharpoonright n) = A \upharpoonright n$, so that $K_{\mathrm{M}'}(A \upharpoonright n) = K_{\mathrm{M}}(B \upharpoonright n)+c$.

□

13.13.1 Degrees of Schnorr random left-c.e. reals

We saw in Theorem 13.3.4, that all left-c.e. random reals were complete under Solovay reducibility, and hence were of Turing degree $\mathbf{0}'$. Later we will see that this result fails for Schnorr random reals. But there is a characterization of the Turing degrees of Schnorr random reals. The first part is the following.

Theorem 13.13.5 (Downey and Griffiths [72])**.** *All Schnorr random left-c.e. reals are of high Turing degree.*

Proof. Given an arbitrary left-c.e. real α we show using the limit lemma and the Solovay-type characterization of Schnorr randomness that if α is Schnorr random then *Tot*, the index set of total computable functions, is Turing reducible to α'.

Let α_s be a strictly increasing c. e. sequence of rationals with limit α, a Schnorr random real. Let T be a c. e. set such that $\langle i, j\rangle \in T$ iff $|W_i| > j$. We show that there exists a computable reduction $\widehat{\Gamma}$ with $\widehat{\Gamma}^{\alpha'} = Tot$, by constructing Γ such that $\Gamma^{\alpha}(i,k) = T(\langle i,k\rangle)$.

We define a single reduction Γ, but define for each $i \in \omega$ a total Solovay test: a collection of open rational intervals $Q^i = \{Q^i_0, Q^i_1, Q^i_2, ...\}$.

Proceed in stages $s \in \omega$. At each stage s there are two main steps: defining an axiom for the least argument $\langle i,k\rangle$ not yet dealt with, and checking whether α_s has changed on the use of previously defined axioms.

- Find the least pair $\langle i,k\rangle$ for which Γ has no axiom. Set $\Gamma^{\alpha_s}(i,k) = T_s(\langle i,k\rangle)$ with use $\gamma(i,k) = k$. If this value is 1, that is $\langle i,k\rangle$ has already entered T_s, then add the interval $(0, 2^{-k})$ to Q^i (these intervals are added simply to make it easy to show later that Q^i is a *total* Solovay test).

- Consider all Γ axioms that have just been "injured" by a change from α_{s-1} to α_s. Reset Γ on the injured arguments i,k to the current value $T_s(\langle i,k\rangle)$ with use $\gamma(i,k) = k$. If $\langle i,k\rangle$ has just entered T_s, then add the interval $(0,2^{-k})$ to Q^i. If $\Gamma^{\alpha_s}(i,k) = 0$ for a pair i,k, but $T_s(\langle i,k\rangle)$ has changed from 0 to 1, then add the interval $(\alpha_s \upharpoonright \langle i,k\rangle, \alpha_s \upharpoonright \langle i,k\rangle + 2^{-k})$ to the test Q^i (these are the really crucial intervals in relation to α).

If $i \notin Tot$ then we will define $\Gamma^{\alpha}(i,k)$ to be zero for all but finitely many k, and the limit in k exists and is correct, equal to 0.

Each Solovay test Q^i contains either finitely many intervals or, if $Tot(i) = 1$, an interval of width 2^{-k} for each $k \in \omega$. In either case it is a total Solovay test.

If $i \in Tot$ then as α is Schnorr random it will move out of all but finitely many of our sets $\{Q^i_n\}_{n\in\omega}$ so we will get the chance to reset $\Gamma(\langle i, k\rangle)$ to 1 if necessary in all but finitely many cases. Hence Γ will have the correct limit in k.

$$(\forall i)\ \lim_k \Gamma^\alpha(i, k) = \lim_k T(\langle i, k\rangle) = Tot(i)$$

$\square$

14
The Quantity of K- and other Degrees, $K(\Omega)$ and $K(\Omega^{(n)})$

14.1 Introduction

In this chapter we will examine basic questions about the function K as well as related questions about the orderings $\leqslant_C$ and other measures of relative complexity. Also we will introduce some other natural maeasures of relative randomness apparently unrelated to initial segment complexity, but naturally relative Kolmogorov complexity to the Turing degrees, and other measures of relative computational complexity. These include the pre-ordering $\leqslant_{\mathcal{L}R}$ and $\leqslant_{vL}$, *low for random* and *van Lambalgen* reducibilities respectively.

14.2 Uncountably cones of K- and C-degrees

For $\leqslant_K$, and $\leqslant_C$, we will prove that there are $2^{\aleph_0}$ many degrees of random reals, and look at comparing the initial segment complexities of natural n-random sets. This fact that the quantity of K-degrees is the continuum is not completely obvious, since the usual argument (every degree is countable and hence there are continuum many degrees of reals) does not obviously apply. For instance the following argument shows that lower cones can be uncountable.

Theorem 14.2.1 (Yu, Ding, Downey [328])**.** *Suppose that α is 1-random. There are $2^{\aleph_0}$ many reals which are K-reducible to α. Indeed, there are sets of every m-degree K-reducible to α.*

Proof. Define $\mathcal{A} = \mathcal{P}(\{2^n : n \in \mathbb{N}\})$. Evidently, $|\mathcal{A}| = 2^{\aleph_0}$ (Note for every $X \subseteq \mathbb{N}$, there is a set $A \in \mathcal{A}$ so that $X \equiv_m A$). For every set A, define $B(A) = \{n : 2^n \in A\}$. Then for every $A \in \mathcal{A}$ and n, $K(A \upharpoonright n) \leqslant K(\log n) + K(B(A) \upharpoonright \log n) + c$

$\leqslant 2\log n + 4\log\log n + c'$

$\leqslant n + c''$

$\leqslant K(\alpha \upharpoonright n) + c'''$.

Thus $A \leqslant_K \alpha$ for every $A \in \mathcal{A}$. □

Essentially the same proof shows the following.

Corollary 14.2.2 (Yu, Ding, Downey [328]). *Suppose that α is 1-random. There are $2^{\aleph_0}$ many reals which are C-reducible to α. Indeed, there are sets of every m-degree C-reducible to α.*

We remark that whilst the *cardinality* of sets K-reducible to α might be large, the *measure* of sets K-reducible to α is small, since we will prove the Yu, Ding Downey Theorem from [328] that *for all α, $\mu(\{\beta : \beta \leqslant_K \alpha\}) = 0$, (Theorem 14.4.13), and hence there are uncountably many K-degrees, and the Yu-Ding Theorem from [327] that there are $2^{\aleph_0}$ many K-degrees (Theorem 14.4.14).*

We remark that it is unknown if there is an *uncountable K-degree.* NOT TRUE TO BE FIXED

However, we will begin with important results concerning the initial segment complexity of Ω itself. We will then detour through yet further measures of relative randomness, especially van Lambalgen reducibility introduced by Miller and Yu.

14.3 On $K(\Omega)$ and other 1-randoms

14.3.1 $K(\alpha \upharpoonright n)$ vs $K(n)$ for α random

Earlier we have seen a number of results, especially of Miller ans Yu such as Theorems 9.4.1 and 9.4.2, were we obtained relatively precise bounds on the behaviour of $K(\alpha)$ for a 1-random α. In the present section, we will present results of Solovay from [284], where results looking at the relationship between $K(n)$ and $K(\alpha \upharpoonright n)$ are established. The main result is summarized as follows.

Theorem 14.3.1 (Solovay [284]). *Let α be Martin-Löf random. Suppose that g is computable and $\sum_{n=1}^{\infty} 2^{-g(n)} = \infty$, and ℓ is computable monotone and tends to ∞ with n.*[1] *Then there exist infinitely many n with*

(i) $K(n) \geqslant g(n) - \mathcal{O}(1)$, and one of the following holding:

[1] The prototypes here would be $g(n) = \log n$ and $\ell(n) = \log\log n$.

(iia) $K(\alpha \upharpoonright n) \geqslant n + g(n) - \mathcal{O}(1)$.

(iib) $K(\alpha \upharpoonright n) \leqslant n + \ell(n) - \mathcal{O}(1)$.

Actually, using Corollary 9.4.2, we can improve (iia) as follows.

Theorem 14.3.2. *Suppose that g is an arbitrary function with $\sum_{m \in \mathbb{N}} 2^{-g(m)} = \infty$. Suppose that α is 1-random. Then there are infinitely many m with*

(i) $K(n) \geqslant g(n) - \mathcal{O}(1)$ *and*

(iia)' $K(\alpha \upharpoonright n) > m + g(n) - \mathcal{O}(1)$.

Proof. (of Theorem 14.3.2) This is an easy Corollary to Theorem 9.4.2, which gives $(iia)'$. Now suppose that $K(\alpha \upharpoonright n) > n + g(n) - \mathcal{O}(1)$. We know by Chaitin's Counting Theorem, $K(\alpha \upharpoonright n) \leqslant n + K(n) - \mathcal{O}(1)$. Thus, for such n,

$$n + g(n) - \mathcal{O}(1) < n + K(n) - \mathcal{O}(1),$$

and hence $g(n) \leqslant K(n) - \mathcal{O}(1)$, as required. □

Proof. (of (iib))

We prove that $K(n)$ can be large and yet $K(\alpha \upharpoonright n)$ is small for infinitely many n. Solovay remarks that his proof is in turn based on the proof, akin to Theorem 9.2.2, due to Martin-Löf that $C(\alpha \upharpoonright n) << n$ infinitely often.

So suppose that g is computable with $\sum_{n=1}^{\infty} 2^{-g(n)} = \infty$, and and ℓ is computable monotone and tends to ∞ with n. Solovay begins by making some reductions. Let $\widehat{g}(n) = \min\{g(n), \rfloor 2 \log n \lfloor\}$. Since $\sum\{2^{-g(j)} : g(j) \geqslant 2 \log n\} < \infty$, we have that $\sum 2^{-\widehat{g}(n)} = \infty$. Thus if we can prove the desired result with $\widehat{g}$ in place of g, then since $K(n) < 2 \log n$ for almost all n, we must have $g(n) = \widehat{g}(n)$ for almost all n for which (i) of the statement of theorem holds, with $\widehat{g}$ in place of g. The upshot of all of this is that, henceforth, we can assume that $g(n) \leqslant 2 \log n$, which we shall do.

The first lemma allows us to assume that $K(n) \leqslant g(n) + \ell(n)$ for all n.

Lemma 14.3.3. *There is a computable function h with the following properties:*

(i) $g(n) \leqslant h(n) \leqslant 2 \log n$.

(ii) $\sum_{n=1}^{\infty} 2^{-h(n)} = \infty$.

(iii) $\sum_{n=1}^{\infty} 2^{-h(n)-\ell(n)} < \infty$.

Proof. We will define $h(n)$ so that it is either $g(n)$ or $2 \log n$. Then (i) of Lemma 14.3.3 will be clear. We define $h(n)$ in stages. At stage i we will define $h(n)$ for $m_i \leqslant n < m_{i+1}$. Matters are arranged so that

(a) if $\ell(n) < i$, $h(n) = 2 \log n$, and

(b) $1 \leqslant \sum\{2^{-h(n)} : m_i \leqslant n < m_{i+1} \wedge \ell(n) \geqslant i\} < 2$.

Since $\ell(n) \to \infty$ with n and $\sum_{n=1}^{\infty} 2^{-g(n)} = \infty$, there is no difficulty doing this. Note that (b) ensures that Lemma 14.3.3 (ii) holds, and (b) ensures that lemma 14.3.3 (iii) holds. (ADD DETASILS) $\square$

Thus we can now replace g by h, and we will rechristen h by g, so that

$$\sum_{n=1}^{\infty} 2^{-g(n)-\ell(n)} < \infty.$$

This is okay since $g \leqslant h$, and hence if we prove $K(n) > g(n)$ with the new g, it holds *a fortiori* with the old one.

Finally, by replacing g by g_1 with $g_1 - g$ tending to inftinity with n and $g_1(n) < 3\log n$, we reduce The task of proving Theorem 14.3.1 (i) to showing that

$$K(\alpha \upharpoonright n) \leqslant n + g(n) + \mathcal{O}(1).$$

We claim that that this and Theorem 14.3.1 (ii) will follow from

$$(c)\ K(\alpha \upharpoonright n) \leqslant n + k(n) - g_1(n) + \mathcal{O}(1)$$

holding infinitely often. Indeed (c) and Theorem 14.3.1 (i) entail

$$K(n) \geqslant g_1(n) + \mathcal{O}(1).$$

On the other hand, $n + K(n) - g_1(n) \leqslant n + g(n) + \ell(n) - g_1(n) + \mathcal{O}(1)$ which is below $n + \ell(n)$ for sufficiently large n.

The conclusion is that we have reduced things to the following: We are given computable g with $\sum_{n=1}^{\infty} 2^{-g(n)} = \infty$ with $g(n) \leqslant 3\log n$. And we must prove that for some D and for infinitely many n

$$K(\alpha \upharpoonright n) \leqslant n + K(n) - g(n) + D.$$

It remains to construct a Martin-Löf test to show that this happens. As Solovay remarks, this construction emulates the one of Martin-Löf where he shows that $C(\alpha \upharpoonright n) << n$ infinitely often. (Martin-Löf [199]) We construct a computable sequence of c.e. open sets U_n with

(1) $\mu(U_n) = 2^{-g(n)}$, and

(2) Each $x \in 2^{\omega}$ lies in U_n for infinitely many n.

We may also assume that that each U_n is represented by a set of strings W_{e_n} with all the strings in W_{e_n} having length presicely $3\log n$. Then since $g(n) \leqslant 3\log n$, for all reals x,

$$K(x \upharpoonright 3\log n | e_n, g(n)) \leqslant 3\log n - g(n).$$

Now, *a fortiori*, $K(x \upharpoonright 3\log n | n) \leqslant 3\log n - g(n)$. Also, since $K(n - 3\log n | n) = \mathcal{O}(1)$, we see

$$K(x \upharpoonright n | (x \upharpoonright 3\log n), n) \leqslant n - 3\log n + \mathcal{O}(1).$$

hence, for infinitely many n,

$$K(x \upharpoonright n|n) \leqslant n - g(n) + \mathcal{O}(1).$$

The conslusion is that for any x, and particularly for α, we have

$$K(x \upharpoonright n) \leqslant n + K(n) - g(n) + \mathcal{O}(1),$$

completing the proof. $\square$

14.3.2 The growth rate of $K(\Omega)$ and the α function

An important function if the following "monotone approximation" to K^{-1}.

Definition 14.3.4 (Chaitin)**.** Define

$$\alpha(n) = \mu(m)\forall\widehat{m} > m(K(\widehat{m}) > n).$$

Of course, we can similarly define $\alpha^{(n)}$ as α for $U^{\emptyset^{(n)}}$, and denote $\alpha^{(1)}$ by α'. Note that if $n > m$ then $\alpha(n) \geqslant \alpha(m)$. This will be important in the next section.

We conside the rate of convergence of the series $\Omega = \sum_{n=0}^{\infty} 2^{-K(n)}$. Let

$$s(n) = -\log(\sum_{j=n}^{\infty} 2^{-K(j)}).$$

Notice that $s(m) \to \infty$.

Theorem 14.3.5 (Solovay [284])**.** $s(m) = \alpha(m) + \mathcal{O}(\log \alpha(m))$.

Proof. We begin by recalling the Chaitin's proof that is g is computable nondecreasing and tending to infinity with n, then $\alpha(n) < g(n)$ almost always. If we replace g by $\lfloor \log g(n) \rfloor$ then we see that it is enough to prove that for all n,

$$\alpha(n) \leqslant g(n) + \mathcal{O}(1).$$

Then if we put $h(n) = \mu j(g(j) \geqslant n)$, h is computable and monotonic. Moreover, $h(g(n)+1) > n$. Also

$$\alpha(h(n)) \leqslant K(h(n)) \leqslant K(n) + \mathcal{O}(1) \leqslant n + \mathcal{O}(1).$$

Hence it follows that $\alpha(n) \leqslant \alpha(h(g(n)+1)) \leqslant g(n) + \mathcal{O}(1)$. Thereore α grows slower than any function tending to infinity with n.

Solovay's proof continues by establishing the inequality

$$s(n) \leqslant \alpha(n).$$

To see this, let $\alpha(n) = j$. Then for some $m \geqslant n$, $K(m) = j$. Consequently,

$$\sum_{i=n}^{\infty} 2^{-K(i)} \geqslant 2^{-K(m)} = 2^{-\alpha(n)}.$$

Taking logs, this gives $s(n) \leqslant \alpha(n)$.

To complete the result, we need to establish the converse inequality. From $\Omega \upharpoonright m$ we shall demonstrate how to compute an integer t_m such that

(i) $H(t_n|\Omega \upharpoonright n) = \mathcal{O}(1)$.

(ii) $s(t_n) > n$.

Granted this, we prove that

$$(iii)\ \alpha(n) \leqslant s(n) + K(s(n) + \mathcal{O}(1),$$

from which we derive the theorem. Assuming (i) and (ii), to show that (iii) holds, let $s(n) = k$. Consider t_k. Then $s(t_k) > k = s(n)$. Hence $t_k > n$. On the other hand, by (i),

$$K(t_k) \leqslant K(\Omega \upharpoonright k) + \mathcal{O}(1) \leqslant K(k) + k + \mathcal{O}(1).$$

Therefore, $\alpha(n) \leqslant K(t_k) \leqslant K(k) + k + \mathcal{O}(1)$, which gives (iii). Now since $s(n) \leqslant \alpha(n)$, we get

$$\alpha(n) = s(n) + \mathcal{O}(\log s(n)).$$

It follows that $\alpha(n)$ is asymptotically equal to $s(n)$, and hence $\log \alpha(n) = \log s(n) + \mathcal{O}(1)$, giving the desired result.

Now we turn to the construction of t_n satisfying (i) and (ii). Our result uses Theorem **??** which the reader should recall stated that

$$p_n =_{\text{def}} \mathit{Card}(\{x : |x| = n \wedge U(x)\downarrow\}) \sim 2^{n-K(n)}.$$

As usual, let $\Omega_s = \sum_{U(x)\downarrow[s]} 2^{-|x|}$, where we can assume that at each stage s exactly one new string x_s enters the domain of U. Of course $\Omega_s \to \Omega$. Now given $\Omega \upharpoonright n$, compute k_n minimal with $\Omega_{k_n}\Omega \upharpoonright n$. Let t'_n be the least integer greater than $|x_i|$ for all $i \leqslant k_n$. The

$$2^{-n} > \Omega - \Omega \upharpoonright n \geqslant \sum\{2^{-|x|} : |x| > t'_n \wedge U(x)\downarrow\}$$

$$\geqslant \sum_{j>t'_n} 2^{-j} p_j \geqslant c \sum_{j>t'_n} 2^{-j}.2^{j-K(j)} = c2^{-s(t'_n)}.$$

Taking logs, we see

$$s(t'_n) \geqslant n + \mathcal{O}(1).$$

Thus for suitable choice of k, if we put $t_n = t'_{n+k}$, we see $s(t_n) \geqslant n+1 > n$. On the other hand, $K(t_n|\Omega \upharpoonright n) \leqslant K(t_n|\Omega \upharpoonright n+k) + K(\Omega \upharpoonright n+k|\Omega \upharpoonright n)$. The first term is $\mathcal{O}(1)$ by the expicit description of t'_n from $\Omega \upharpoonright n$. The second is evidently $\mathcal{O}(1)$ since k is fixed independert of n. This concludes the proof of the result. $\square$

14.3.3 $K(\Omega)$ vs $K(\Omega')$

For the purposes of the present section, we will fix a standard universal machine U. The first person to look at the K-degrees of randoms was Solovay [284]. He analysed the relationshp between $K(\Omega)$ and $K(\Omega')$, where Ω' denotes $\Omega^{\emptyset'}$. We will let $U^{(n)}$ denote $U^{\emptyset^{(n)}}$ and U', $U^{\emptyset'}$. To do so he ised the α function of the previous section.

Theorem 14.3.6 (Solovay [284]). $\exists^\infty k(K(\Omega \restriction k) \leqslant k + \alpha'(k) + \mathcal{O}(\log \alpha'(k))$.

Proof. For each n we will define $m_n = k$ as above. This is done as follows. We will define a prefix-free machine M that works as follows. On any input it tries to parse as xyz with $U(x) = |y|$. Then M tries to interpret z as an inital segment of Ω, specifically the use of the computation of $U'^{\Omega}(y)$. M will halt with value k if $U'^{\Omega}(y) = k$, precisely if z is the use of this computation such that $|z| = \max\{k, \textit{active use}\}$. Then if M halts it will output z.

Note $K(z) \leqslant |x| + |y| + |z|$.

Let $a'(n) = \mu j(\alpha'(j) > n)$. Note that $K'(a'(n)) = n + \mathcal{O}(1)$. To see this, we note that $K'(a'(n))$ cannot be smaller than n by its definition. But since it is the *least*, then $a'(n) - 1$ must have $K'(a'(n)) = n$, and hence $K'(a'(n)) \leqslant n + \mathcal{O}(1)$. Thus we can pick y so that $|y| = n + \mathcal{O}(1)$ and $U'(y) = a'(n)$. Now pick x such that $U(x) = |y|$. We need $\mathcal{O}(\log n)$ many bits to specify y. Let z be the unique initial segment of Ω such that $M(xyz)\downarrow$. (That is $U'^{z}(y)\downarrow$.)

Now let $m_n = |z|$. Since $m_n \geqslant U'(y) = a'(n)$, we see $\alpha'(m_n) > n$. But $\alpha'(m_n) \leqslant K'(m_n) \leqslant |y| + \mathcal{O}(1) = n + \mathcal{O}(1)$. (Since using y we can simulate $U'(y)$ to get z. Therefore $\alpha'(m_n) = n + \mathcal{O}(1)$.

We are finished since $K(\Omega \restriction m_n) \leqslant |x| + |y| + |z| \leqslant \mathcal{O}(\log \alpha'(m_n)) + \alpha'(m_n) + m_n$. $\square$

We can now show that the K-degrees of Ω and Ω' differ.

Theorem 14.3.7 (Solovay [284]). *Suppose that X is 2-random. Then*

$$K(X \restriction n) \geqslant n + \alpha(n) + \mathcal{O}(\log(\alpha(n))).$$

Evidently α' grows more slowly than any function computable in $\emptyset'$, such as $\alpha(\alpha(n))$, and hence we see the following[2].

Theorem 14.3.8 (Solovay [284]). *Suppose that X is 2-random, then $X \restriction n \not\leqslant_K \Omega$.*

Lemma 14.3.9 (Solovay [284]). $K'(n) \leqslant K(n) - \alpha(n) + \mathcal{O}(\log(\alpha(n)))$.

[2] Actually, Solovay did not state this result in terms of 2-randoms, but in terms of α arithmetically random. However, the stronger statement can be extracted from Solovay's proof.

Proof. (of Lemma 14.3.9) The basic idea is that n gives $\alpha(n)+\mathcal{O}(\log(\alpha(n)))$ bits of information about $\Omega \upharpoonright \alpha(n)$. Thus, given an oracle for the dimain of U, we can compute $\Omega \upharpoonright \alpha(n)$ from $\alpha(n)$, impying $K'(\Omega \upharpoonright \alpha(n)) = \mathcal{O}(\log(\alpha(n)))$. hence using an oracle we can eliminate $\alpha(n)+\mathcal{O}(\log(\alpha(n)))$ bits from the description of n.

In detail, we have $K(\Omega \upharpoonright \alpha(n)|n) = \mathcal{O}(\log(\alpha(n)))$. Therefore, $K(n|\Omega \upharpoonright \alpha(n)) = K(n)+\mathcal{O}(\log(\alpha(n)))$. On the other hand, $K(n, \Omega \upharpoonright \alpha(n)) = K(\Omega \upharpoonright \alpha(n)) + K(n|\Omega \upharpoonright \alpha(n))$ which equals

$$\alpha(n) + \mathcal{O}(\log(\alpha(n))) + K(n|\Omega \upharpoonright \alpha(n)).$$

Hence, rearranging, $K(n|\Omega \upharpoonright \alpha(n)) = n - \alpha(n) + \mathcal{O}(\log(\alpha(n)))$. Therefore

$$K'(n) \leqslant K'(n|\Omega \upharpoonright \alpha(n))+K'(\Omega \upharpoonright \alpha(n)) \leqslant K(n|\Omega \upharpoonright \alpha(n))+K'(\Omega \upharpoonright \alpha(n))+\mathcal{O}(1)$$

$$= K(n) - \alpha(n) + K'(\Omega \upharpoonright \alpha(n)) + \mathcal{O}(\log(\alpha(n))).$$

But since $\Omega \leqslant_T \emptyset'$, we have that $K'(\Omega \upharpoonright \alpha(n)) = \mathcal{O}(\log(\alpha(n)))$, proving the lemma. $\square$

Proof. (of Theorem 14.3.8) We know that if α is 2-random, then

$$K'(X \upharpoonright n) \geqslant n - \mathcal{O}(1),$$

by relativizing Schnorr's Theorem. By Lemma 14.3.9, we have

$$K(X \upharpoonright n) \geqslant n - \mathcal{O}(1) + \alpha(X \upharpoonright n) + \mathcal{O}(\log(\alpha(X \upharpoonright n))).$$

By the monotonicity of α we have

$$K(X \upharpoonright n) \geqslant n + \alpha(n) + \mathcal{O}(\log(\alpha(n))).$$

$\square$

Solovay also proved a number of other results about the possible behaviour of $K(\Omega \upharpoonright n)$, and, as we have already seen, he showed that arithmetically random reals X (indeed 3-random reals) have

$$\exists^\infty n(K(X \upharpoonright n) > n + K(n) - \mathcal{O}(1).$$

If we could show that 2-randoms did not have this property, then we would know how to differentiate between 2-randoms and 3-randoms using only their initial segment complexities.

Unfortunately, we don't know the answer to this fundamantal question, nor is there is a clear way to extend Solovay's methods to $\Omega^{(n)}$ for $n \geqslant 2$. In the next chapter we will introduce new methods to show that, indeed, n-randomness can be defined using *some* properties of K, and that for all $n \neq m$,

$$\Omega^{(n)}|_K\Omega^{(m)}.$$

To finish this section, we include one final result of Solovay whose proof has never appeared, and one for which we cannot derive the result using the techniques of the later section. This result shows that $K(\Omega \restriction n)$ is sometimes very close to n in a particular way.

Theorem 14.3.10 (Solovay, [284])**.** *There exist an infinite series of pairs of numbers m_n^0, m_n^1 such that for $i = 0, 1$,*

(i) $\alpha'(m_n^i) = n + \mathcal{O}(1)$.

(ii) $K(\Omega \restriction m_n^i) \leqslant m_n + n + \mathcal{O}(1)$.

(iii) $K(m_n^0) = \log m_n^0 + K(\log m_n^0) + \mathcal{O}(1)$.

(iv) $K(m_n^1) = \alpha(m_n^1) + \mathcal{O}(1)$.

Proof. The key clauses are (i) and (ii). Having proven these modifications will be given to get the others. When the context is clear, we will drop the superscripts in the following.

We begin with (i) The idea of the construction is the following. Suppose that we are given Ω and a binary string x. Then, using Ω we can determine membership in $\mathrm{dom}(U)$, and hence simulate the action of U' upon the input x. We will continue to use the digits of U as needed in this simulation. If $U'(x) \downarrow = \ell$, say, then this simulation will end having read some k first bits of Ω. The we will read $\max\{0, \ell - k\}$ bits of Ω and halt.

To see this is can be implemented by a prefix-free machine C, we can regard C as receiving an input of the form xyz. We would regard this as

(1) $U(x) = n$.

(2) $|y| = n$.

(3) z is an initial segment of Ω of length $\max\{k, \ell\}$.

z will be used to similate $U'(y)$. The C's action is to read x (since U will only halt on x), interpret the next n symbols as y, and interpret z as an initial segment of Ω, and use this to simulate $U'(y)$, as outlined above, the output of $C(xyz)$ being z.

The point is, suppose that n is given. Select y of length $n + \mathcal{O}(1)$ so that $U'(y) = \alpha'(n)$. Let x be of length $\mathcal{O}(\log n)$ with $U(x) = |y|$. Finally, let z be the initial segment of Ω such that xyz is in $\mathrm{dom} z$. (Hence $|z| \geqslant \alpha'(n)$.)

We let $m_n = |z|$. Note that z is computable from y by a function computable from $\emptyset'$, and hence $|z| \leqslant \alpha'(n + \mathcal{O}(1))$, and $\alpha'(|z|) = \alpha'(m_n) = n + \mathcal{O}(1)$, as required.

To establish (ii), Solovay begins by proving the weaker version of (ii) :

$$(ii)' : K(\Omega \restriction m_n) \leqslant m_n + n + \mathcal{O}(\log n).$$

This is easy after the material so far. We simply note that

$$K(\Omega \restriction m_n) = K(z) \leqslant |\Pi_C| + |xyz| = m_n + n + \mathcal{O}(\log n).$$

To improve this estimate (ii)' to (ii), Solovay revises the definition of the machine C to a new one $\widehat{C}$. This revised version will simulate $U'(x)$. It does so in three phases:

(1) It reads a new bit of its data.

(2) It simulates the oracle portion of U'.

(3) It performs deterministic computations.

If it is in phase (1), it will assume that the next bit of data is a bit of x; if it is phase (2), it will assume that the next block of digits on its data tape is the next portion of Ω needed to simulate the oracle. Thus, there is some word

$$s = w_0 z_0 w_1 z_1 \dots w_n z_n$$

such that $w_0 w_1 \dots w_n = y$ and $z_0 z_1 \dots z_n = z$. The $\widehat{C}(s)$ is defined and equals z. Therefore

$$K(\Omega \upharpoonright |z|) = K(z) \leqslant |\pi_{\widehat{C}}| + |y| + |z| = m_n + n + \mathcal{O}(1).$$

Now we need to get m_n^0 and m_n^1 to satisfy (iii) and (iv) as well. By Chaitin's Counting Theorem there is a constant d such that

$$\forall n \exists m (n \leqslant m \leqslant 4n \wedge K(m) \geqslant |m| + K(|m|) - d).$$

Note the given $\Omega \upharpoonright n$ we can determine $x \in \mathrm{dom}(U)$ such that $|x| \leqslant n\}$, by running the enumeration of Ω_s until it is correct on the first n bits. Thus we can computer from $\Omega \upharpoonright n$ both $\alpha(n)$ and an $m \in [n, 4n]$ such that $K(m) \geqslant |m| + K(|m|) - d$.

We let m_n^0 be $\alpha(m_n)$ and m_n^1 be the least integer $\geqslant m_n$ such that $K(m_n^1) \geqslant |m_n^1| + K(|m_n^1|) - d$. Hence (ii) and (iv) hold by choice of the m_n^i.

Now $m_n \geqslant \alpha`(n)$ and hence $\alpha'(m_n^i) \geqslant n$. On the other hand, m_n^i can be obtained by a computable process using $\Omega \upharpoonright m_n$, and hence by a procedure applied to $\emptyset'$ and m_n. Thus

$$K'(m_n^i) \leqslant K(m_n) + \mathcal{O}(1),$$

and so $K`(m_n^i) = n + \mathcal{O}(1)$. Finally, $K(\Omega \upharpoonright m_n^i) \leqslant K(\Omega \upharpoonright m_n^i | m_n^i, \Omega \upharpoonright m_n) + K(m_n^i | \Omega \upharpoonright m_n) + k(\Omega \upharpoonright m_n)$ This is $\leqslant (m_n^i - m_n) + \mathcal{O}(1) + m_n + n + \mathcal{O}(1) = m_n^i + n + \mathcal{O}(1)$. Therefore (ii) also holds for the m_n^i and the result follows.

□

14.4 van Lambalgen reducibility, $\leqslant_{\mathcal{L}R}$, $\leqslant_C$, and $\leqslant_K$.

We have seen a number of different measures of relative randomness such as Solovay, sw-, rK-, K- and C-. In this section will will introduce yet another measure of relative randomness, interesting for two reasons. First for its own sake, and second as a tool for the analysis of $\leqslant_K$ and $\leqslant_C$ *and their interactions with* $\leqslant_T$.

The following definition is based upon the fundamental result of van Lambalgen we met in Chapter 11: *If $x, y \in 2^\omega$, then $x \oplus y$ is 1-random iff x is 1-random and y is 1-x-random.* (Theorems 11.6.2 and 11.6.5.)

Definition 14.4.1 (van Lambalgen reducibility, Miller and Yu [216])**.** (i) For $x, y \in 2^\omega$, write $x \leqslant_{vL} y$ if $(\forall z \in 2^\omega)$ $x \oplus z$ is 1-random $\implies$ $y \oplus z$ is 1-random.

(ii) We call the equivalence classes induced by this relation the *van Lambalgen degrees*.

There are other closely related reducibilities which are largely unexplored both in terms of themselves and in terms of their interrelations. For example by André Nies [226] defined the following:

Definition 14.4.2 (Low for random reducibility, Nies [226])**.** Define $y \leqslant_{\mathcal{L}R} x$ if $(\forall z \in 2^\omega)$ z is 1-x-random $\implies$ z is 1-y-random.

By van Lambalgen's Theorem, if x and y are both 1-random, then $y \leqslant_{\mathcal{L}R} x$ iff $x \leqslant_{vL} y$. Another related reducibility would be generated by monotone Kolmogorov complexity from Chapter 9.5.

Definition 14.4.3 (Extended monontone reducibility)**.** We say that $x \leqslant_{Km\oplus} y$ iff $(\forall z \in 2^\omega)$ $x \oplus z \leqslant_{Km} y \oplus z$.

Since all 1-randoms have the same Km degree by Corollary 9.5.8, $\leqslant_{Km\oplus}$ implies $\leqslant_{vL}$, but $\leqslant_{Km\oplus}$ also makes sense on non-random reals.

However, for our purposes, we will concentrate upon van Lambalgen reducibility.

The *vL*-degree turns out to be a weak measure of the degree of randomness. It is proven in Corollaries 14.4.11 and **??** that both $\leqslant_K$ and $\leqslant_C$ refine $\leqslant_{vL}$.

This is the key that allowed Miller and Yu [216] to transfer facts about the *vL*-degrees to the K- and C- degrees. This is useful because many basic properties of the *vL*-degrees are easy to prove from known results.

14.4.1 Basic properties of the van Lambalgen degrees

The basic properties of the van Lambalgen degrees use the fundamental Theorem 11.6.5, as well as Kučera's Theorem 11.4.1 *that every degree* $\mathbf{a} > \mathbf{0}'$ *is random*, and the Kučera-Terwijn Theorem **??** *that for every real x there is a real $y \not\leqslant_T x$ such that every $1 - x-$random real is $1 - x \oplus y-$random.*

Theorem 14.4.4 (Miller and Yu [216])**.**

(i) *If $x \leqslant_{vL} y$ and x is n-random, then y is n-random.*

(ii) *The least vL-degree is $\mathbf{0}_{vL} = \{x \mid x \text{ is not 1-random}\}$.*

(iii) *If $x \oplus y$ is 1-random, then x and y have no upper bound in the vL-degrees.*

(iv) *If $y \leqslant_T x$ and y is 1-random, then $x \leqslant_{vL} y$.*

(v) *There are 1-random reals $x \equiv_{vL} y$ but $x <_T y$.*

Proof. (i) Assume that $x \leqslant_{vL} y$ and x is n-random. First consider $n = 1$. Select a 1-x-random real z. Then $x \oplus z$ is 1-random, so $y \oplus z$ is 1-random. Thus y is 1-random.

Now take $n > 1$. By Kučera's Theorem, there is a 1-random $z \equiv_T \emptyset^{(n-1)}$. Then x is n-random $\implies$ x is 1-z-random $\implies$ $x \oplus z$ is 1-random $\implies$ $y \oplus z$ is 1-random $\implies$ y is 1-z-random $\implies$ y is n-random.

(ii) Clearly, if x is not 1-random then $x \leqslant_{vL} y$ for any real y. If x is 1-random, then $x \nleqslant_{vL} \emptyset$, or else $\emptyset$ would be 1-random by part (i). Therefore, $\mathbf{0}_{vL}$ consists exactly of the non-random reals.

(iii) Let $x \oplus y$ be 1-random and assume, for a contradiction, that $x, y \leqslant_{vL} z$. Because $x \leqslant_{vL} z$ and $x \oplus y$ is 1-random, $z \oplus y$ is 1-random too. Therefore, $y \oplus z$ is 1-random. But $y \leqslant_{vL} z$, so $z \oplus z$ must also be 1-random, which is a contradiction.

(iv) Assume that $y \leqslant_T x$ and y is 1-random. For any $z \in 2^\omega$, $x \oplus z$ is 1-random $\implies$ z is 1-x-random $\implies$ z is 1-y-random $\implies$ $y \oplus z$ is 1-random. Therefore, $x \leqslant_{vL} y$.

(v) Pick any random real $x \geqslant_T \emptyset'$ (e.g., $x = \Omega$). By Lemma **??**, there is a $w \nleqslant_T x$ such that every 1-x-random is 1-$x \oplus w$-random. Take a 1-random real $y \equiv_T x \oplus w$; this is guaranteed to exist by Lemma **??**(i). So $x <_T y$. Also, $z \oplus x$ is 1-random if and only if z is 1-x-random if and only if z is 1-$x \oplus w$-random if and only if z is 1-y-random if and only if $z \oplus y$ is 1-random. Therefore, $x \equiv_{vL} y$. □

Corollary 14.4.5 (More basic properties of the vL-degrees)**.**

(i) *There is no join in the vL-degrees.*

(ii) *If $x \oplus y$ is 1-random, then $x \oplus y <_{vL} x, y$ and $x \mid_{vL} y$.*

(iii) *There is no maximal vL-degree.*

(iv) *There is no minimal random vL-degree.*

(v) *The Σ^0_1 theory of $(R, \leqslant_{vL})$ is decidable, where R denotes the collection of vL-degrees of random sets.*

Proof. (i) Immediate from part (iii) of Theorem 14.4.4.

(ii) By part (iii) of Theorem 14.4.4, $x \mid_{vL} y$. By part (iv) of the same theorem, $x \oplus y \leqslant_{vL} x, y$. Therefore, $x \oplus y <_{vL} x, y$.

(iii) If $x = x_1 \oplus x_2$ is a 1-random real, then $x <_{vL} x_1, x_2$ by part (i). So vL-degree is maximal.

(iv) Assume x is 1-random. Take any 1-x-random real y; thus $x \oplus y$ is 1-random. By part (i), $\emptyset <_{vL} x \oplus y <_{vL} x$. So, there is no minimal random vL-degree.

(v) As in Lerman [176], it suffices to prove that every finite poset can be embedded into $(R, \leqslant_{vL})$. Suppose $\mathbb{P} = (P, \leqslant)$ and $P = \{p_i\}_{i<n}$. Pick a

1-random real $x = \oplus_{i<n} x_i$. For any $k < n$, define $F(k) = \{i \mid p_k \leqslant p_i\}$ and let $y_k = \oplus_{i \in F(k)} x_i$. Let $g\colon P \to R$ be defined by $g(p_k) = y_k$. It suffices to prove that $p_j \leqslant p_k$ if and only if $y_j \leqslant_{vL} y_k$. If $p_j \leqslant p_k$. Then $F(k) \subseteq F(j)$ so $y_j \leqslant_{vL} y_k$, by part (i). If $p_j \nleqslant p_k$, then $k \notin F(j)$ and so $x_k \oplus y_j$ is 1-random. But $x_k \oplus y_k$ is not 1-random since $k \in F(k)$. So $y_j \nleqslant_{vL} y_k$. □

The folllwing corollary improves earlier results of Yu, Ding and Downey [328] that for all $n > m$ $\Omega^{(m)} \nleqslant_K \Omega^{(n)}$.

Corollary 14.4.6 (Miller and Yu [216]). *If $m \neq n$, then $\Omega^{\emptyset^{(m)}}$ and $\Omega^{\emptyset^{(n)}}$ have no upper bound in the vL-degrees.*

It is also clear that Ω is the least Δ^0_2 1-random real in the *vL*-degrees and that no 2-random can be $\geqslant_{vL}$ a Δ^0_2 1-random.

The following is a beautiful result saying that there really are quite deep unapparent connections between computability and randomness.

Corollary 14.4.7 (Miller and Yu [216]). *If x is n-random and $y \leqslant_T x$ is 1-random, then y is n-random.*

Proof. Immediate from parts (i) and (iv) of Theorem 14.4.4. □

Again we see the theme that randomness is a *lowness* property. The point is that Corollary 14.4.7 does not work the *other* way. Kučera's Theorem 11.4.1 says that every degree above $\mathbf{0}'$ is 1-random, but none can by $n > 1$ random. Thus we have the following.

Observation 14.4.8. *For every n-random degree* **a** *there is a random degree* $\mathbf{b} > \mathbf{a}$ *which is not* 2*-random.*

14.4.2 Below a z-random and $\leqslant_T$

In this section we will prove the following strong extension of Theorem 14.4.7.

14.4.3 $\leqslant_{vL}$, $\leqslant_K$ and $\oplus$

The key ingredient to a nexus between $\leqslant_K$ and $\leqslant_{vL}$ was the realization by Miller and Yu that the initial segment complexity of x determines, for any $z \in 2^\omega$, whether $x \oplus z$ is 1-random. This then allowed them to prove the important result that $x \leqslant_K y$ implies $x \leqslant_{vL} y$, so the results of the previous section have consequences in the K-degrees. For example, Corollary 14.4.12(i) implies that there is a $\leqslant_K$ characterizationof, say, n-randomness, for each n.

Recall that strings of length n can be thought of as representing the numbers between 2^n-1 and $2^{n+1}-2$. We also need some additional notation for the proof of the theorem and of Theorem **??** below.

Definition 14.4.9 (Miller and Yu [216]). Define $x\widehat{\oplus}z$ to be

$$\langle z_0, x_0, x_1, z_1, x_2, x_3, x_4, x_5, z_2, x_6, \ldots\ldots, z_n, x_{2^n-2}, \ldots, x_{2^{n+1}-3}, z_{n+1}, \ldots\ldots\rangle.$$

Clearly, $x\widehat{\oplus}z$ is 1-random iff $x \oplus z$ is 1-random. In fact, $x\widehat{\oplus}z \equiv_{vL} x \oplus z$. We can also define $\sigma\widehat{\oplus}\tau$ for strings $\sigma, \tau \in 2^{<\omega}$, provided that $2^{|\tau|-1} - 2 \leqslant |\sigma| \leqslant 2^{|\tau|} - 2$.

Theorem 14.4.10. *If $x, z \in 2^\omega$, then $(\forall n)\ K(x \upharpoonright (z \upharpoonright n)) \geqslant z \upharpoonright n + n - \mathcal{O}(1)$ iff $x \oplus z$ is 1-random.*

Proof. First, assume that $x \oplus z$ is 1-random. Thus, $x\widehat{\oplus}z$ is also 1-random. Note that $K(x \upharpoonright (z \upharpoonright n)) = K((x \upharpoonright (z \upharpoonright n))\widehat{\oplus}(z \upharpoonright n+1)) + \mathcal{O}(1)$ (where it is necessary to use $z \upharpoonright n+1$ to make the $\widehat{\oplus}$ well defined). But $(x \upharpoonright (z \upharpoonright n))\widehat{\oplus}(z \upharpoonright n+1) = (x\widehat{\oplus}z) \upharpoonright (z \upharpoonright n+n+1)$, so $K(x \upharpoonright (z \upharpoonright n)) \geqslant K((x\widehat{\oplus}z) \upharpoonright (z \upharpoonright n+n+1)) - \mathcal{O}(1) \geqslant z \upharpoonright n + n - \mathcal{O}(1)$, for all n.

For the other direction, define a prefix-free machine $M\colon 2^{<\omega} \to 2^{<\omega}$ as follows. To compute $M(\tau)$, look for τ_1, τ_2, η_1 and η_2 such that $\tau = \tau_1\tau_2$, $\mathcal{U}(\tau_1) = \eta_1\widehat{\oplus}\eta_2$ and $|\eta_1\tau_2| = \eta_2$. If these are found, define $M(\tau) = \eta_1\tau_2$.

Assume that $x \oplus z$ is not 1-random. Then for each k, there is an m such that $K((x\widehat{\oplus}z) \upharpoonright m) \leqslant m - k$. Take strings η_1 and η_2 such that $\eta_1\widehat{\oplus}\eta_2 = (x\widehat{\oplus}z) \upharpoonright m$ and let τ_1 be a minimal $\mathcal{U}$-program for $\eta_1\widehat{\oplus}\eta_2$. Let $n = |\eta_2|$. Note that $|\eta_1| \leqslant 2^n - 2$ and that $\eta_2 \geqslant 2^n - 1$. So, there is a string τ_2 such that $\eta_1\tau_2 = x \upharpoonright \eta_2$. Then $M(\tau_1\tau_2) = x \upharpoonright \eta_2$. Therefore,

$$\begin{aligned}
K(x \upharpoonright (z \upharpoonright n)) &\leqslant K(x \upharpoonright \eta_2) \leqslant K_M(x \upharpoonright \eta_2) + \mathcal{O}(1) \leqslant |\tau_1\tau_2| + \mathcal{O}(1)\\
&\leqslant K(\eta_1\widehat{\oplus}\eta_2) + |\tau_2| + \mathcal{O}(1) \leqslant |\eta_1\eta_2| - k + |\tau_2| + \mathcal{O}(1)\\
= |\eta_1\tau_2| + |\eta_2| - k + \mathcal{O}(1) &= \eta_2 + |\eta_2| - k + \mathcal{O}(1) = z \upharpoonright n + n - k + \mathcal{O}(1),
\end{aligned}$$

where the constant depends only on M. Because k was arbitrary, $K(x \upharpoonright (z \upharpoonright n)) - z \upharpoonright n - n$ is not bounded below. Therefore, $(\forall n)\ K(x \upharpoonright (z \upharpoonright n)) \geqslant z \upharpoonright n + n - \mathcal{O}(1)$ implies that $x \oplus z$ is 1-random. ☐

Corollary 14.4.11 (Miller and Yu [216]). *$x \leqslant_K y \implies x \leqslant_{vL} y$.*

Combined with Theorem 14.4.4 and Corollary 14.4.5, this corollary has the following important consequence for the structure of the K-degrees.

Corollary 14.4.12 (Miller and Yu [216]). (i) *If $x \leqslant_K y$ and x is n-random, then y is n-random.*

(ii) *If $x \oplus y$ is 1-random, then $x \mid_K y$ and x and y have no upper bound in the K-degrees. Therefore, there is no join in the K-degrees.*

We remark that Miller ans Yu had direct proofs of some of these results, but they were much more complex.

[add results about Ω's; in particular, $\{\Omega^{\emptyset^{(n)}}\}_{n\in\mathbb{N}}$ is a K-antichain and the columns of Ω form an infinite K-antichain of Δ^0_2 random reals]

Another important consequence of Corollarycor:K-implies-vL is the following fundamental result showing that whilst the cardinality of a K-cone might be big, the measure is always small. Again the following result was orginally proven by direct and more complex methods.

Theorem 14.4.13 (Yu, Ding and Downey [328]). $\mu(\{\beta : \beta \leqslant_K \alpha\}) = 0$.

Proof. If β is 1-α-random, then $\beta \nleqslant_{vL} \alpha$ and hence, since $\mu(\{\beta : \beta$ is 1-α-random $\}) = 1$, we get $\mu(\{\beta : \beta \leqslant_K \alpha\}) = 0$, by applying Corollary 14.4.11. □

Similar methods also show that the cardinality of the K-degrees of random reals is as large as it can be. (Actually, since $\leqslant_K$ is Borel, the following is a consequence of Silver's Theorem **??** and Theorem 14.4.13.)

Theorem 14.4.14 (Yu and Ding [327]). *There are $2^{\aleph_0}$ K-degrees of random reals.*

Again Theorem 14.4.13 had a direct proof, but it is equivalent to another theorem from classical measure theory, as observed by Miller and Yu.

Theorem 14.4.15 (???). *For any relation $E \subseteq 2^\omega \times 2^\omega$, if $\mu(E) = 1$, then there is a perfect set $X \subseteq 2^\omega$ so that $\forall x, y \in X(x \neq y \implies (x, y) \in E)$.*

We can get Theorem 14.4.14 from Theorem 14.4.15 by considering the relation $E = \{\langle x, y\rangle :\}$.

14.4.4 On the structure of the monotone and process degrees

We have seen that there is a single Km and Km_D degree containing all random reals, namely those with $Km(\alpha \upharpoonright n) =^+ n$ for all n. Thus virtually all of the previous material does not apply to $\leqslant_{Km}$. Nevertheless we can show that that there are continuum many Km-degrees.

In the below we will let U_m denote the universal monotone machine and U_D the universal discrete (process) monotone machine.

We begin with a technical lemma about the behaviour of Km.

Definition 14.4.16 (Calhoun [**?**]). For $\sigma, \tau \in 2^{<\omega}$, let $\mu(\sigma, \tau)$ denote the least initial segment of σ (if any) incomparible with τ.

Lemma 14.4.17 (Calhoun [**?**]). *(i) For $\sigma|\tau$,*

$$K(\mu(\sigma, \tau) \leqslant^+ Km(\sigma) + Km(\tau) + 2\log Km(\sigma) + 2\log Km(\tau).$$

(ii) For any σ,

$$\max\{Km(\sigma0), Km(\sigma1)\} \geqslant \frac{K(|\sigma| + 1)}{3} - \mathcal{O}(1).$$

(iii) For any σ, τ,

$$Km(\tau) \leqslant^+ Km(\sigma\tau) + K(|\sigma|).$$

(iv) *For any σ, there is a constant c such that for any τ,*

$$Km(\tau) \leqslant^+ Km(\sigma\tau) + c.$$

(v) *For any computable real β, there is a constant c_β such that for any $\sigma \in 2^{<\omega}$ and n,*

$$Km(\sigma(\beta \upharpoonright n)) \leqslant K(\sigma) + c_\beta.$$

Proof. (i) Consider the prefix free machine M which on input $z = pq$ computes $U(p)U(q)$, after parsing z into pq, and then outputs $\mu(U(p), U(q))$. Notice that this shows that $M(\sigma^*\tau^*) = \mu(\sigma, \tau)$, and hence

$$K(\mu(\sigma,\tau)) \leqslant^+ K(\sigma)+K(\tau) \leqslant^+ K_m(\sigma)+2\log Km(\sigma)+k_m(\tau)+2\log Km(\tau).$$

(ii) Without loss of generality assume $Km(\sigma 0) \geqslant Km(\sigma 1)$. By (i), $K(\sigma 0) \leqslant^+ Km(\sigma 0) + 2\log Km(\sigma 0) + Km(\sigma 1) + 2\log Km(\sigma 1)$. Choosing σ sufficiently long, $K(\sigma 0) \leqslant 3Km(\sigma 0)$, we see $\max\{Km(\sigma 0), Km(\sigma 1)\} = Km(\sigma 0) \geqslant \frac{K(\sigma 0)}{3} - \mathcal{O}(1) \geqslant \frac{K(|\sigma|+1)}{3} - \mathcal{O}(1)$.

(iii) Define a monotone machine M by $M(pq) = \nu$ if $U(p) = 1^k$ and $U_m(q) = \nu$ with $|\nu| = k$. Apply this to $p = (1^{|\sigma|})^*$ and q the optimal monotone description of τ.

(iv) Apply (iii) with $c' = K(|\sigma|) + c$ with c the constant in (iii).

(v) By Theorem 9.5.12 we know that there is a constant c with $Km(\sigma(\beta \upharpoonright n) \leqslant K(\alpha) + Km(\beta \upharpoonright n) + c$, but since β is computable, $Km(\beta \upharpoonright n) \leqslant k$ for all n, and hence (v) follows. □

Theorem 14.4.18 (Calhoun [?]). *For any real α which is neither random nor computable, there is a real β with $\alpha|_{Km}\beta$.*

Proof. Let γ be a random real. Then β is constructed in stages by a finite extension argument. For even s, let $\beta_{s+1} = \beta_s 0^k$ for some k such that $Km(\beta_s 0^k) > Km(\alpha \upharpoonright (|\beta_s|+k) - s$. Such an s exists since $Km(\alpha \upharpoonright n) \to \infty$ and $Km(\beta_s 0^k)$ is bounded by Lemma 14.4.17 (v). At odd stages, let $\beta_{s+1} = \beta_s\frown(\gamma \upharpoonright k)$ for some k with $Km(\beta_s\frown(\gamma \upharpoonright k)) > Km(\alpha \upharpoonright (|\beta_s|+k)) + s$. Such a k exists since $\beta_s\gamma$ is random. Therefore $Km(\beta_s\gamma \upharpoonright n) =^+ n$ by Levin's Theorem. Since α is not random, $Km(\alpha \upharpoonright (|\beta_s|+n)$ is bounded away from n. The even stages will force $Km(\alpha \upharpoonright n) \not\leqslant Km(\beta \upharpoonright n)$ and the odd stages $Km(\beta \upharpoonright n) \not\leqslant Km(\alpha \upharpoonright n)$. □

Dovetaining the same technique as the proof of Theorem 14.4.18, it is easy to construct a prefect tree of such degrees. Hence the following is also true.

Theorem 14.4.19 (Calhoun [?]). *There exist an antichain of Km-degree of size $2^{\aleph_0}$.*

Since $\alpha \leqslant_{Km_D} \beta$ implies $\alpha \leqslant_{Km} \beta$, we have as corollaries that the two theorems above also hold of the discrete monotone degrees.

Calhoun was also able to prove some structural results about the monotone degrees. Using a similar finite extension argument we have the following.

Theorem 14.4.20 (Calhoun [?]). *There exist a minimal pair of Km-degrees.*

Proof. We constaruct α and β in stages. Let $\alpha_0 = \beta_0 = \lambda$, and $n_0 = 0$. At an even stage s, let α_s^* be an extension (say, by a random string) of α_s with $Km(\alpha_s^*) > Km(\alpha_s$. Following Calhoun [?], call a string σ *terminal* if there is no proper extension τ of σ with $Km(\tau) = Km(\sigma)$. Find $n = n_{s+1}$ so large that the following hold

(i) There is no terminal σ with $|\sigma| > n$ and $Km(\sigma) < Km(\alpha_s^*) + c_2 + s$, where c_2 is the constant from Lemma 14.4.17 (v), and applied to 0^ω.

(ii) $n \geqslant |\alpha_s^*|$, and,

(iii) For all $m \geqslant n$, $K(m) \geqslant 3(K(\sigma_s^*) + c_1 + c_2 + s$, where c_1 is the constant from Lemma 14.4.17 (ii).

Let $\beta_{s+1} = \beta_s\widehat{\ }0^{n-|\beta_s|}$ and $\alpha_{s+1} = \alpha_s^*\widehat{\ }0^{n-|\alpha_s^*|}$. At odd stages s we do the above switching the roles of α and β.

To see that α and β form a minimal pair, suppose that $\gamma \leqslant_{Km} \alpha, \beta$. We will say that s is an *increment stage* if $Km(\gamma \upharpoonright n_{s+1}) Km(\gamma \upharpoonright n_s)$. If there are only finitely many increment stages, then we are done as γ is computable. Thus we suppose that there are infinitely many increment stages. Without loss of generality there are infinitely many even increment stages. For each increment even stage s, let x be such that $n_s < x \leqslant n_{s+1}$ and $Km(\gamma \upharpoonright x - 1) < Km(\gamma \upharpoonright x)$. As $x > n_s$, if $\gamma \upharpoonright x - 1$ is terminal, then $Km(\gamma \upharpoonright) \geqslant K(\alpha_s^*) + c_2 + s$ by condition (i). If $\gamma \upharpoonright x - 1$ is not terminal, then $Km(\gamma \upharpoonright x) = \max\{Km((\gamma \upharpoonright x-1)\widehat{\ }0), Km((\gamma \upharpoonright x-1)\widehat{\ }1)\} \leqslant \frac{Km(x)}{3} - c_1$, by Lemma 14.4.17 (ii). By condition (iii), $K(m) \geqslant 3(K(\alpha_s^*) + c_1 + c_2 + s)$. As a consequence, $Km(\gamma \upharpoonright x) > K(\alpha_s^*) + c_2 + s$ whether $\gamma \upharpoonright x - 1$ is terminal or not. By Lemma 14.4.17 (v),

$$Km(\alpha \upharpoonright x) = Km(\alpha_s^*\widehat{\ }0^{x-|alpha_s^*|}) \leqslant K(\alpha_s^*) + c_2.$$

But this is a contradiction, since then $Km(\gamma \upharpoonright x) \geqslant Km(\alpha \upharpoonright x) + s$, and s is unbounded. □

We remark that Calhoun also shows how to construct many uncountable Km degrees. Given a real α and increasing function f, he defines $\alpha \otimes f(n)$ to be $\alpha(f^{-1}(n))$ for n the range of f, and 0 otherwise.

Theorem 14.4.21 (Calhoun [?]). *(i) If f is strictly increasing and computable, then $Km(\alpha \otimes f \upharpoonright n) =^+ Km(\alpha \upharpoonright f^{-1}[n])$ where $f^{-1}[n] = \max\{K : k \leqslant f(k) < n\}$.*

(ii) Hence, if α and β are reals if $Km(\alpha \upharpoonright n) =^+ Km(\beta \upharpoonright n)$ then $Km((\alpha \otimes f) \upharpoonright n) =^+ Km((\beta \otimes f) \upharpoonright n)$.

For the proofs below, for a string σ, define $\sigma \otimes f$ as above, except that $\sigma \otimes f$ has length $f(|\sigma|)$.

Proof. To show $Km((\alpha \otimes f) \upharpoonright n) \leqslant^+ Km(\alpha \upharpoonright f^{-1}[n])$, we build a monotone machine M. Put $M(p) = \sigma$ if $\sigma = (\tau \otimes f)\widehat{\ }0^k$ for some string τ, $k \in \omega$ with $U_m(p) = \tau$ and $k < f(|\tau| + 1) - f(|\tau|)$. To see that M is monotone, suppose that $M(p) = \sigma_1$ and $M(q) = \sigma_2$ with p and q compatible. Then the corresponding τ_p and τ_q must also be compatible as U_m is monotone. Assuming $\tau_p \preccurlyeq \tau_q$, we have that $\sigma_p, \sigma_q \preccurlyeq (\tau_q \otimes f)\widehat{\ }0^\omega$. Therefore σ_p and σ_q are compatible. Thus M is monotone. Now if $\sigma = \alpha \otimes f \upharpoonright n$, then $M(p) = \sigma$ where p is an optimal description of $\alpha \upharpoonright f^{-1}[n]$, and hence $Km(\sigma) \leqslant |p| = Km(\alpha \upharpoonright f^{-1}[n])$.

For the converse direction, we construct a monotone machine N proving $Km(\alpha \otimes f \upharpoonright n) \geqslant^+ Km(\alpha \upharpoonright f^{-1}[n])$. Define $N(p) = \sigma$ whenever $U_m(p) = \sigma \otimes f$. If p and q are comparable with $U_m(q) = \sigma_q$, then also $U_m(p) = \sigma_q \otimes f$. But as U_m is monotone, $\sigma \otimes f$ and $\sigma_q \otimes f$ are comparable, and we may suppose $\sigma \otimes f \preccurlyeq \sigma_q \otimes f$. But then $\sigma \preccurlyeq \sigma_q$ by definition of $\otimes$. Thus N is monotone. Finally, puttine $\sigma = \alpha \upharpoonright f^{-1}[n]$ yields $\sigma \otimes f \upharpoonright f(f^{-1}[n]) \preccurlyeq \alpha \otimes f \upharpoonright n$, and hence if p is an optimal description of $\sigma \otimes f$, $Km(\sigma) = |p| = Km(\sigma \otimes f) \leqslant Km(\alpha \otimes f \upharpoonright n)$, proving (i). We remark that (ii) is immediate from (i). □

Using this elegant result, Calhoun is able to prove the following.

Theorem 14.4.22 (Calhoun [**?**])**.** *There is an order preserving embedding from the rationals into the monotone degrees such that each degree in the image of the embedding has cardinality $2^{\aleph_0}$.*

Proof. For any rational number $r \in (0, 1)$, let $f_r(n) = \lfloor \frac{n}{r} \rfloor$. Let α be any random real. By Theorem 14.4.21 (ii), $Km(\alpha \otimes f_r \upharpoonright n) =^+ Km(\alpha \upharpoonright f_r^{-1}[n]) =^+ f_r^{-1}[n]$, as α is random. By construction, $f_r^{-1}[n] = \max\{k : \lfloor \frac{k}{r} \leqslant n\} =^+ rn$. Then the map $r \mapsto \alpha \otimes f_r$ induces the embedding as $r < s$ implies $rn < sn$, and if α and β and $\alpha \neq \beta$, we see $\alpha \otimes f_r \equiv_{Km} \beta \otimes f_r$ and yet $\alpha \otimes f_r \neq \beta \otimes f_r$, so that the degrees have cardinality $2^{\aleph_0}$ as there are $2^{\aleph_0}$ many random reals. □

14.4.5 Contrasting the K-degrees and vL-degrees

It might be hoped that, for instance, the K-degrees and vL-degrees might coincide. Alas this is not the case. The theorem below implies that $\leqslant_{vL}$ differs from $\leqslant_K$, even for 1-random Δ^0_2 reals.

Theorem 14.4.23 (Miller and Yu [**?**])**.** *For any 1-random real x, there is a 1-random $y \leqslant_T x \oplus \emptyset'$ such that $x \mid_K y$.*

The difficulty here is getting the precise degree bound. Easier methods construct a $y \leqslant_T x'$ (in fact with $y' \leqslant x'$) with $x|_K y$. To see this, use x to construct a $\Pi_1^{0,x}$ class of x-random reals. Now use the Low Basis Theorem to get a 1-x-random real low over x. Then by van Lambalgen's Theorem, $x \oplus y$ is random and $x|_K y$, by Corollary 14.4.12. This would suffice to construct a K-antichain in the Δ_2^0 degrees starting with a low random real.

Proof. (of Theorem 14.4.23.) Let $\mathcal{R} = \{z \mid (\forall n)\ K(z \upharpoonright n) \geqslant n\}$ and note that $\mu(\mathcal{R}) \geqslant 1/2$. We define two predicates:

$$
\begin{aligned}
&A(\tau, p) \text{ if and only if } \mu(\{z \succ \tau \mid z \notin \mathcal{R}\}) > p \\
\text{and } &B(\sigma, s) \text{ if and only if } (\exists n < |\sigma|)\ K(\sigma \upharpoonright n) > K(x \upharpoonright n) + s \\
&\qquad\qquad\qquad\qquad \wedge (\exists m < |\sigma|)\ K(\sigma \upharpoonright m) < K(x \upharpoonright m) - s.
\end{aligned}
$$

where $\sigma, \tau \in 2^{<\omega}$, $p \in \mathbb{Q}$ and $s \in \mathbb{N}$. It should be clear that $B(\sigma, s)$ is uniformly decidable from $x \oplus \emptyset'$. To see that $A(\tau, p)$ can be decided by $\emptyset'$, note that it is equivalent to $(\exists s)\ \mu(\{z \succ \tau \mid (\exists n \leqslant s)\ K_s(z \upharpoonright n) < n\}) > p$. We construct $y = \bigcup_{s \in \mathbb{N}} \sigma_s$ by finite initial segments $\sigma_s \in 2^{<\omega}$ such that $B(\sigma_{s+1}, s)$ holds. This guarantees that $x \mid_K y$. We also require the inductive assumption that $\mu(\{z \mid z \succ \sigma_s\} \cap \mathcal{R}\}) > 0$. This ensures that $y \in \mathcal{R}$ because $\mathcal{R}$ is closed. Therefore, y is 1-random. Finally, the construction will be done relative to the oracle $x \oplus \emptyset'$, so $y \leqslant_T x \oplus \emptyset'$.

Stage $s = 0$: Let $\sigma_0 = \emptyset$. Note that $\mu(\{z \mid z \succ \sigma_0\} \cap \mathcal{R}\}) = \mu(\mathcal{R}) \geqslant 1/2 > 0$, so the inductive assumption holds for the base case.

Stage $s+1$: We have constructed σ_s such that $\mu(\{z \mid z \succ \sigma_s\} \cap \mathcal{R}\}) > 0$. Using the oracle $x \oplus \emptyset'$, search for $\tau \succ \sigma_s$ and $p \in \mathbb{Q}$ such that $B(\tau, s)$, $p < 2^{-|\tau|}$ and $\neg A(\tau, p)$. If these are found, then set $\sigma_{s+1} = \tau$ and note that it satisfies our requirements. In particular, $\mu(\{z \mid z \succ \sigma_{s+1}\} \cap \mathcal{R}\}) \geqslant 2^{-|\sigma_{s+1}|} - p > 0$.

All that remains is to verify that the search succeeds. We know by Corollary 14.4.12(ii) that $\mu(\{z \mid x \mid_K z\}) = 1$. Therefore, $\mu(\{z \succ \sigma_s \mid z \mid_K x\} \cap \mathcal{R}) > 0$. So there is a $\tau \succ \sigma_s$ such that $B(\tau, s)$ and $\mu(\{z \mid z \succ \tau\} \cap \mathcal{R}) > 0$. This implies that there is a $p \in \mathbb{Q}$ such that $p < 2^{-|\tau|}$ and $\neg A(\tau, p)$.

This completes the construction. □

Essentially the same proof gives the following result.

Theorem 14.4.24. *For any finite collection $x_0, \ldots, x_n$ of 1-random reals, there is another 1-random real $y \leqslant_T x_0 \oplus \cdots \oplus x_n \oplus \emptyset'$ such that, for every $i \leqslant n$, y and x_i have no upper bound in the K-degrees.*

As was noted above, Ω is vL-below every Δ_2^0 1-random. On the other hand, Theorem 14.4.23 implies that there is a Δ_2^0 1-random $y \in 2^\omega$ such that $y \mid_K \Omega$. Therefore, $\leqslant_{vL}$ does not, in general, imply $\leqslant_K$ on the Δ_2^0 1-random reals.

14.5 Random K-degrees are countable

15
Triviality and lowness for K-

In this, and the next Chapters, we will explore some truly fascinating results showing how simple initial segment complexities can have dramatic effects on a sets computational complexity. The material on what we call K-trivial reals allows us to solve Post's problem "naturally" (well, reasonably naturally), without the use of a priority argument, or indeed without the use of requirements. We will also explore the related notion of lowness, which is when oracles don't help.

15.1 K-trivial and K-low reals

We begin by looking at the notions of triviality and lowness- anti-randomness properties- in the setting of prefix-free Kolmogorov complexity. The material will culminate in the powerful work on Nies showing that these classes turn out to be the same, and have remarkable properties in terms of their Turing degrees.

15.1.1 The basic triviality construction : the tailsum game

In this section, we will look at the basic methods of constructing both K-low and K-trivial reals. Recall that Chaitin's Theorem 6.4.2 shows that if, for all n, $C(\alpha \upharpoonright n) \leqslant C(n) + \mathcal{O}(1)$, then α is computable. Chaitin asked if the same result held for K in place of C. As we soon see in Theorem 15.1.3, Chaitin was able to show that if $K(\alpha \upharpoonright n) \leqslant K(n) + \mathcal{O}(1)$, then $\alpha \leqslant_T \emptyset'$.

Surprisingly, we cannot replace $\emptyset'$ by $\emptyset$ for K. That is, even though α looks identical to ω we cannot conclude that α is computable even for strongly c.e. reals α. This leads to the following definition.

Definition 15.1.1 (Downey, Hirschfeldt, Nies and Stephan [81])**.** We will call a real α *K-trivial* if $\alpha \leqslant_K \mathbb{N}$.

The reader should note that it is enough that the real be K-trivial on an infinite computable set. That is the following piece of folklore is true.

Proposition 15.1.2. *Suppose that we have a computable set $A = \{a_1, a_2, \dots\}$ in increasing order of magnitude, and for all i, $K(A \upharpoonright a_i) \leqslant K(a_i) + \mathcal{O}(1)$. Then A is K-trivial.*

Proof. Let $h(n) = a_n$, be computable. Then $K(n) = K(h(n)) + \mathcal{O}(1) = K(a_n) + \mathcal{O}(1)$. Notice that $K(n) \leqslant K(a_n) + \mathcal{O}(1)$, since to compute $A \upharpoonright n$, take the program for $A \upharpoonright h(n)$, then reconstruct n from $h(n)$ and truncate $A \upharpoonright h(n)$ to gte $A \upharpoonright n$. Then $K(A \upharpoonright n) \leqslant K(A \upharpoonright h(n)) + \mathcal{O}(1) \leqslant K(h(n)) + \mathcal{O}(1) \leqslant K(n) + \mathcal{O}(1)$. ☐

Using the same method as for Theorem 6.4.2, but with the underlying tree computed from $\emptyset'$, Chaitin proved the following (which we significantly improve later). This result says that if K-trivial reals exist then they are relatively simple in terms of their computational complexity.

Theorem 15.1.3 (Chaitin [43])**.** *If α is K-trivial, then $\alpha \leqslant_T \emptyset'$.*

Solovay was the first to construct a (Δ^0_2) K-trivial reals. This method was adapted by Zambella [329] and later Calude and Coles [34] to construct a left-c.e. K-trivial real. In [81], Downey, Hirschfeldt, Nies, and Stephan gave a new construction of a K-trivial real, and this time the real was strongly c.e. (Independently, Kummer had also constructed a K-trivial strongly c.e. real in an unpublished manuscript.) As we will later see this is a priority-free and later requirement free solution to Post's problem.

Theorem 15.1.4. (Downey, Hirschfeldt, Nies, and Stephan [81], Calude and Coles [34], after Solovay [284]) *There is a noncomputable c.e. set A such that $K(A \upharpoonright n) \leqslant K(n) + O(1)$.*

Remark While the proof below is easy, it is slightly hard to see why it works. So, by way of motivation, suppose that we were to asked to "prove" that the set $B = \{0^n : n \in \omega\}$ has the same complexity as $\omega = \{1^n : n \in \omega\}$. A complicated way to do this would be for us to build our own prefix-free machine M whose only job is to compute initial segments of B. The idea would be that if the universal machine U converges to 1^n on input σ then $M(\sigma)\downarrow = 0^n$. Notice that, in fact, using the Kraft-Chaitin Theorem it would be enough to build M *implicitly*, enumerating the length axiom $\langle |\sigma|, 0^n \rangle$. We are guaranteed that $\sum_{\tau \in \mathrm{dom}(M)} 2^{-|\tau|} \leqslant \sum_{\sigma \in \mathrm{dom}(U)} 2^{-|\sigma|} \leqslant 1$, and hence the Kraft-Chaitin Theorem applies. Note also that we could, for convenience and as we do in the main construction, use a string of length $|\sigma|+1$, in which case we would ensure that $\sum_{\tau\in\text{dom}(M)} 2^{-|\tau|} < 1/2$.

Proof of Theorem 15.1.4. The idea is the following. We will build a noncomputable c.e. set A in place of the B described in the remark and, as above, we will slavishly follow U on n in the sense that whenever U enumerates, at stage s, a shorter σ with $U(\sigma) = n$, then we will enumerate an axiom $\langle |\sigma|+1, A_s \upharpoonright n\rangle$ for our machine M. To make A noncomputable, we will also sometimes make $A_s \upharpoonright n \neq A_{s+1} \upharpoonright n$. Then for each j with $n \leqslant j \leqslant s$, for the currently shortest string σ_j computing j, we will also need to enumerate an axiom $\langle |\sigma_j|, A_{s+1} \upharpoonright j\rangle$ for M. This construction works by making this extra measure added to the domain of M small. *This extra measure is called the* tailsum*, and this is called the* tailsum method.

We are ready to define A:

$$A = \{\langle e,n\rangle : \exists s\,(W_{e,s} \cap A_s = \emptyset \wedge \langle e,n\rangle \in W_{e,s} \wedge \sum_{\langle e,n\rangle\leqslant j\leqslant s} 2^{-K(j)[s]} < 2^{-(e+2)})\},$$

where $W_{e,s}$ is the stage s approximation to the eth c.e. set and $K(j)[s]$ is the stage s approximation to the K-complexity of j.

Clearly A is c.e. Since $\sum_{j\geqslant m} 2^{-K(j)}$ goes to zero as m increases, if W_e is infinite then $A^{[e]} \cap W_e^{[e]} \neq \emptyset$. It is easy to see that this implies that A is noncomputable. Finally, the extra measure put into the domain of M, beyond one half of that which enters the domain of U, is bounded by $\sum_e 2^{-(e+2)}$ (corresponding to at most one initial segment change for each e), whence

$$\sum_{\sigma\in\text{dom}(M)} 2^{-|\sigma|} < \sum_{\sigma\in\text{dom}(U)} 2^{-(|\sigma|+1)} + \sum_e 2^{-(e+2)} \leqslant \frac{1}{2} + \frac{1}{2} = 1.$$

Thus M is a prefix-free machine, and hence $K(A \upharpoonright n) \leqslant K(n) + O(1)$. □

Before we turn to the related notion of lowness, we mention that clearly the proof also admits many variations. For instance, we can make A promptly simple, or below any nonzero computably enumerable degree. We cannot control the jump or make A Turing complete, since, as we will see, all K-trivials are nonhigh (and in fact, as shown by Nies [226], low).

15.1.2 The requirement-free version

As we see in Section 15.1.6, the construction above automatically yields a Turing incomplete c.e. set. It is thus an *injury-free* solution to Post's problem. It is not, however, *priority-free*, in that the construction depends on an ordering of the simplicity requirements, with stronger requirements allowed to use up more of the domain of the machine M. We can do methodologically better by giving a priority-free solution to Post's problem, in the sense that no explicit diagonalization (such as that of W_e above) occurs in the construction of the incomplete c.e. set, and therefore the construction of this set (as opposed to the verification that it is K-trivial) does not depend on an ordering of requirements. We now sketch this method, which is rather more like that of Solovay's original proof of the existence of a Δ_2^0 K-trivial real.

Let us reconsider the key idea in the proof of Theorem 15.1.4. At certain stages we wish to change an initial segment of A for the sake of diagonalization. Our method is to make sure that the total measure added to the domain of our machine M (which proves the K-triviality of A) due to such changes is bounded by 1. Suppose, on the other hand, we were *fortunate* in the sense that the universal machine itself "covered" the measure needed for these changes. That is, suppose we were lucky enough to be at a stage s where we desire to put n into $A_{s+1} - A_s$ *and* at that very stage $K_s(j)$ changes for all $j \in \{n, \dots, s\}$. That would mean that *in any case* we would need to enumerate new axioms describing $A_{s+1} \upharpoonright j$ for all $j \in \{n, \dots, s\}$, whether or not these initial segments change. Thus at that very stage, we could also change $A_s \upharpoonright j$ for all $j \in \{n, \dots, s\}$ at no extra cost. Notice that we would not need to copy the universal machine U at every stage. We could also enumerate a collection of stages $t_0, t_1 \dots$ and only update M at stages t_i. Thus for the lucky situation outlined above, we would only need the approximation to $K(j)$ to change for all $j \in \{n, \dots, t_s\}$ at some stage u with $t_s \leqslant u \leqslant t_{s+1}$. This observation would seem to allow a greater possibility for the lucky situation to occur, since many more stages can occur between t_s and t_{s+1}.

The key point in all of this is the following. Let $t_0, t_1, \dots$ be a computable collection of stages. *Suppose that we construct a set $A = \bigcup_s A_{t_s}$ so that for $n \leqslant t_s$, if $A_{t_{s+1}} \upharpoonright n \neq A_{t_s} \upharpoonright n$ then $K_{t_s}(j) < K_{t_{s+1}}(j)$ for all j with $n \leqslant j \leqslant t_s$. Then A is K-trivial.* We are now ready to define A in a priority-free way.

A requirement-free solution to Post's problem Let $t_0, t_1, \dots$ be a collection of stages such that t_i as a function of i dominates all primitive recursive functions. (Actually, as we will see, dominating the overhead in the Recursion Theorem is enough.) At each stage u, let $\{a_{i,u} : i \in \omega\}$ list $\overline{A}_u$. Define

$$A_{t_{s+1}} = A_{t_s} \cup \{a_{n,t_s}, \dots, t_s\},$$

where n is the least number $\leqslant t_s$ such that $K_{t_{s+1}}(j) < K_{t_s}(j)$ for all $j \in \{n, \dots t_s\}$. (Naturally, if no such n exists, $A_{t_{s+1}} = A_{t_s}$.) Requiring the complexity change for all $j \in \{n, \dots, t_s\}$, rather than just $j \in \{a_{n,t_s}, \dots, t_s\}$, ensures that A is coinfinite, since for each n there are only finitely many s such that $K_{t_{s+1}}(n) < K_{t_s}(n)$.

Note that there is no priority used in the definition of A. It is like the Dekker deficiency set or the so-called "dump set" (see [280], Theorem V.2.5).

It remains to prove that A is noncomputable. By the Recursion Theorem, we can build a prefix-free Turing machine M and know the coding constant c of M in U. That is, if we declare $M(\sigma) = j$ then we will have $U(\tau) = j$ for some τ such that $|\tau| \leqslant |\sigma| + c$. Note further that if we put σ into the domain of M at stage t_s, then τ will be in the domain of U by stage $t_{s+1} - 1$. (This is why we chose the stages to dominate the primitive recursive functions. This was the key insight in Solovay's original construction.)

Now the proof looks like that of Theorem 15.1.4. We will devote 2^{-e} of the domain of our machine M to making sure that A satisfies the e-th simplicity requirement. When we see a_{n,t_s} occur in W_{e,t_s}, where $\sum_{n \leqslant j \leqslant t_s} 2^{-K_{t_s}(j)} < 2^{-(e+c+1)}$, we change the M_{t_s} descriptions of all j with $n \leqslant j \leqslant t_s$ so that $K_{t_{s+1}}(j) < K_{t_s}(j)$ for all such j. The cost of this change is bounded by 2^{-e}, and a_{n,t_s} will enter $A_{t_{s+1}}$, as required.

15.1.3 There are few K-trivial reals

In this section we give a unified proof of some unpublished material of Zambella and of Chaitin's result that all K-trivials are Δ^0_2, while establishing some intermediate results of independent interest. The methods and proofs of this section are taken from Downey, Hirschfeldt, Nies and Stephan [81].

We recall from Chapter 5, the definition of Q_D in the Coding Theorem. That was, given a prefix-free machine D, we let $Q_D(\sigma) = \mu(D^{-1}(\sigma))$ so that $Q_D(\sigma)$ is the probability that D outputs σ. Recall that if D is the fixed universal machine we wrote $Q(\sigma)$ for $Q_D(\sigma)$. The Coding Theorem, Theorem 6.9.4, stated that

$$Q_D(\sigma) = O(2^{-K(\sigma)}).$$

Thus, for reals α and β, we have the following result.

Theorem 15.1.5. *$\alpha \leqslant_K \beta$ iff there is a constant c such that for all n,*

$$cQ(\beta \upharpoonright n) \geqslant Q(\alpha \upharpoonright n).$$

Proof. Suppose that $\alpha \leqslant_K \beta$. Then there is a constant d such that $K(\alpha \upharpoonright n) \leqslant K(\beta \upharpoonright n) + d$ for all n. This happens iff there is a constant d' such that for all n,

$$2^{-K(\alpha \upharpoonright n)} \geqslant d' 2^{-K(\beta \upharpoonright n)}.$$

This happens iff there is a c such that $Q(\alpha \upharpoonright n) \geqslant cQ(\beta \upharpoonright n)$ for all n. The other direction is similar. $\square$

Remark 15.1.6. For any σ, the real $Q(\sigma)$ is random.

Proof. To see that the remark is true we use the KC Theorem to build a machine M and show that $\Omega \leqslant_S Q(\sigma)$, where $\leqslant_S$ is Solovay reducibility (see [79] for a definition and discussion of Solovay reducibility). At stage s, if we see $U(\nu)\downarrow$, where U is the universal machine, we declare that $M(\nu) = \sigma$.

Then for some $c = c_M$, there is a ν' with $U(\nu') = \sigma$, and furthermore $|\nu| \leqslant |\nu'| + c$. Thus whenever we add $2^{-|\nu|}$ to Ω, we add $2^{-(|\nu|+c)}$ to $Q(\sigma)$, and hence $\Omega \leqslant_S Q(\sigma)$, which implies $Q(\sigma)$ is random. □

The Coding Theorem allows us to get an analog of the result of Chaitin [43], Theorem 6.7.4, on the number of descriptions of a string.

Corollary 15.1.7 (to the Coding Theorem)**.** *There is a constant d such that for all c and all σ,*

$$|\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}| \leqslant d2^c.$$

Proof. Trivially,

$$\mu(\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}) \geqslant 2^{-(K(\sigma)+c)} \cdot |\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}|.$$

But also, $\mu(\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}) \leqslant d \cdot 2^{-K(\sigma)}$, by Theorem 15.1.5. Thus,

$$d2^{-K(\sigma)} \geqslant 2^{-c}2^{-K(\sigma)}|\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}|.$$

Hence, $d2^c \geqslant |\{\nu : D(\nu) = \sigma \wedge |\nu| \leqslant K(\sigma) + c\}|$. □

We can now conclude that there are few K-trivials.

Theorem 15.1.8. *The set $S_d = \{\sigma : K(\sigma) < K(|\sigma|) + d\}$ has at most $O(2^d)$ many strings of length n.*

Proof. We build a prefix-free machine V. Suppose that $U_s(\nu) = \sigma \wedge |\nu| \leqslant K_s(|sigma|) + d$. Then define $V(\nu) = |\sigma|$. Then V is prefix-free as U is. Moreover, by Theorem 15.1.7,

$$|\{\nu : V(\nu) = |\sigma| \wedge |\nu| \leqslant K(|\sigma|) + d\}| \leqslant c2^d.$$

But then by construction, there is a constant p such that S_d has at most $pc2^d$ many members. □

We remark that it is possible to give another proof of this result based on the minimality of K. This time, $S_d = \{\sigma : K(\sigma) = K(|\sigma|) + d\}$. We use Kraft-Chaitin to build a machine M. If we see some τ such that $U(\tau) = \sigma$, σ has length n and $|\tau| = K(n) + d$ then we enumerate $\langle |\tau|, n\rangle$. Each such τ will to $Q_d(n)$ and hence the result will follow by the Coding Theorem, and minimality of K.

Corollary 15.1.9. (a) (Zambella [329]) *For a fixed d, there are at most $O(2^d)$ many reals α with*

$$K(\alpha \upharpoonright n) \leqslant K(n) + d$$

for all n.

(b) (Chaitin [43]) *If a real is K-trivial, then it is Δ^0_2.*

Proof. Consider the Δ_2^0 tree $T_d = \{\sigma : \forall \nu \subseteq \sigma(\nu \in S_d)\}$. This tree has width $O(2^d)$, and hence it has at most $O(2^d)$ many infinite paths. For each such path X, we can choose $\sigma \in T_d$ such that X is the only path above σ. Hence such X is Δ_2^0. $\square$

Recall from Definition 6.4.5, we had a notion of a string σ being extendably (C, c)-trivial and a real A to be (C, c) trivial on an infinite set I of places, and Merkle and Stephan [**?**] proved that this implied that $A \leqslant_T I$ using the method of Theorem 6.4.2. In the same way we can define the notion of being *extendably* (K, c) *trivial* on I, and essentially the same methods as above allows us to constructa tree of bounded width containig such reals. Thus we get the following.

Theorem 15.1.10 (Merkle and Stephan [**?**])**.** *Suppose that A is (K, c) trivial on I. Then $A \leqslant I \oplus \emptyset'$.*

Again we will actually show that the K-trivial reals are a subset of the superlow reals.

15.1.4 The G function

Zambella's Theorem, Theorem 15.1.9 (a), leads one to speculate as to *exactly* how many K-trivials there are, and how complicated it is to enumerate them. This question has been investigated in unpublished work of Downey, Miller and Yu.

Definition 15.1.11 (Downey, Miller, Yu [92])**.** Let

$$G(d) = |KT(d)|.$$

G seems a strangely complicated object. We calculate some arithmetical bounds on G. To do this we will need the following combinatorial result.

Theorem 15.1.12 (First Counting Theorem, [92])**.** *(i) $lim_c \frac{G(c)}{2^c} = 0$.*

(ii) Indeed,

$$\sum_{c \in \mathbb{N}} \frac{G(c)}{2^c} \text{ is finite.}$$

Proof. We define $G(c, n) = |KT(c, n)|$.

Lemma 15.1.13. $\sum_{c \in \mathbb{N}} \frac{G(c)}{2^c} \leqslant \liminf_n \sum_c \frac{G(c,n)}{2^c}$.

The proof of Lemma 15.1.13 is almost immediate. Any finite partial sum on the left represents a finite number of K-trivials. For sufficiently large n each of these reals is isolated, so that the sum on the right exceeds that of the left.

Lemma 15.1.14. *There is a finite q such that, for all n,*

$$\sum_{c\in\mathbb{N}} \frac{G(c,n)}{2^c} \leqslant q.$$

Proof. (Of Lemma 15.1.14) By definition of $G(\cdot,\cdot)$ we have that for some constant q, for all n,

$$\sum_{c\in\mathbb{N}} G(c,n)2^{-K(n)-c} \leqslant 2 \sum_{\sigma\in 2^n} 2^{-K(\sigma)}.$$

The first inequality follows from definition of $G(\cdot,\cdot)$. On the other hand, K is a minimal universal computably enumerable semi-measure (by the Coding Theorem), and hence there is a constant q such that for every n,

$$2 \sum_{\sigma\in 2^n} 2^{-K(\sigma)} \leqslant q2^{-K(n)},$$

since the quantity on the left is a computably enumerable semi-measure. $\square$

The proof is now finished by putting together Lemmas 15.1.13 and 15.1.14. $\square$

Before we turn to analyzing the Turing complexity of G, we point out that the result above is relatively sharp in in the following sense.

Theorem 15.1.15 (Second Counting Theorem, [92])**.**

$$G(b) = \Omega(\frac{2^b}{b^2}).$$

Proof. For any string σ, we know that

$$K(\sigma 0^r) \leqslant^+ K(\sigma) + K(0^r) \leqslant^+ K(\sigma) + K(0^{|\sigma|+r}).$$

Now if we choose σ of length below $b - 2\log b$, we know that $K(\sigma) \leqslant b$, and hence $K(\sigma 0^r) \leqslant K(|\sigma 0^r|) + b$. There are $\Omega(\frac{2^b}{b^2})$ such strings σ. $\square$

Theorem 15.1.16 (Downey, Miller, Yu [92])**.** $G \not\leqslant_T \emptyset'$.

Proof. Assume that G is Δ^0_2, and hence by the Limit Lemma, $G(n) = \lim_s G(n,s)$, where this time $G(n,s)$ denotes a computable approximation to $G(n)$. Also we assume that we know k such that that $0^\omega \in KT(k)$.

Using the KC Theorem, we build a machine M with coding constant d, which we know in advance. M looks for the first $c \geqslant k, 2^d$ such that

$$\frac{G(c)}{2^c} < 2^{-d}.$$

We can approximate the c in a Δ^0_2 manner, so that $c = \lim_s c_s$, where c_s is the stage s approximation of c.

The key idea behind the definition of M is that M wants to ensure that there are at least 2^{c-d} $KT(c)$ reals. If it does this, then we have a contradiction.

At stage s, M will pick (at most) 2^{c_s-d} strings of length s extending those of length $s-1$ already defined, and promise to give them each M-descriptions of length $K(s)+c_s-d$ (hence they are $KT(c_s)$ *strings*). It clearly has enough room to do this, since

$$2^{c_s-d}2^{-(K(s)+c_s-d)} = 2^{-K(s)}.$$

At stage s, if c_s has a new value then all of M's old work is abandoned and M starts building 2^{c_s-d} $KT(c_s)$ reals which branch off of 0^ω starting at length s. If c_s has the same old value, then M continues building the 2^{c_s-d} $KT(c_s)$ reals. Eventually c_s stabilizes and M gets to build his 2^{c-d} $KT(c)$ reals, contradicting the definition of c. $\square$

We remark that a crude upper bound on G is that it is computable in $\mathbf{0}'''$. It is unknown if $G \leqslant_T \emptyset''$, or even if this question is machine dependent.

15.1.5 Lowness

In the setting of classical computability theory, recall that a set was called low if its jump was the same Turing degree as $\emptyset'$. The idea is that it is no more helpful as an oracle than $\emptyset$. We look at the same notion for randomness. Let R be the collection of random sets for some randomness concept. Then R^X is the class obtained for the same concept using X as an oracle.

Definition 15.1.17. We say that a set X is R-*low* if $R = R^X$. We say that X is low for R-tests, R_0-low if, for every R^X-test $\{U_n^X : n \in \mathbb{N}\}$, there is a R-test $\{V_n : n \in \mathbb{N}\}$ such that $\cap_n V_n \supseteq \cap_n U_n$.

For instance, a set X is Martin-Löf low if the collection of reals Martin-Löf random relative to X is the same as the collection of Martin-Löf random reals. Hence X is no help in making reals non-random. Notice also that if X is R_0-low then it is automatically R-low, but the converse is not clear. However, since there is a universal Martin-Löf test, we have the following.

Observation 15.1.18. *The collections Martin-Löf low and Martin-Löf low for tests coincide.*

Martin-Löf low sets were first studied by Kučera and Terwijn, answering a question of van Lambalgen [314].

Theorem 15.1.19 (Kučera and Terwijn [160])**.** *There is a c.e. set A that is Martin-Löf low.*

Proof. We give an alternative proof to that in [160], taken from Downey [71]. It is clear that there is a primitive recursive function f, so that $U_{f(n)}^A$ is the universal Martin-Löf test relative to A. Let I_n^A denote the corresponding

Solovay test. Then X is A-random iff X is in at most finitely many I_n^A. We show how to build a $\{J_n : n \in \omega\}$, a Solovay test, so that for each $(p,q) \in I_n^A$ is also in J_n. This is done by simple copying: if $(p,q) < s$ is in $\cup_{j \leqslant s} I_j^{A_s}$ is not in $J_i : i \in s$, add it. Clearly this "test" has the desired property of covering I_n^A. We need to make A so that the "mistakes" are not too big. This is done in the *same* way as the construction of a K-trivial.

The crucial concept comes from Kučera and Terwijn: Let $M_s(y)$ denote the collection of intervals $\{I_n^{A_s} : n \leqslant s\}$ which have $A_s(y) = 0$ in their use function. Then we put $y > 2e$ into $A_{s+1} - A_s$ provided that e is least with $A_s \cap W_{e,s} = \emptyset$, and

$$\mu(M_s(y)) < 2^{-e}.$$

It is easy to see that this can happen at most once for e and hence the measure of the total mistakes is bounded by $\Sigma 2^{-n}$ and hence the resulting test is a Solovay test. The only thing we need to prove is that A is noncomputable. This follows since, with priority e, whenever we see the some y with $\mu(M_s(y)) \geqslant 2^{-e}$, such y will *not* be added and hence this amount of the A-Solovay test will be protected. But since the total measure is bounded by 1, this cannot happen forever. □

We remark that the same method can construct an apparently more unhelpful set.

Definition 15.1.20. We call a set X *strongly* K-low if there is a constant d such that for all σ, $K^X(\sigma) \geqslant K(\sigma) - d$.

This notion is due to An A Muchnik who, in unpublished work, constructed such a real. It is evident that the same method of proof (keeping the measure of the injury of the uses down) will show.

Theorem 15.1.21 (Muchnik, unpubl.)**.** *There is a c.e. set X that is strongly K-low.*

The reader cannot miss the similarities in the proofs of the existence of K-trivials and K-lows and even strongly K-low reals. These are all notions of K-antirandomness, and should somehow be related.

Later we will see in some deep work of Nies that these classes all *coincide*!

15.1.6 K-trivials solve Post's problem

The basic method introduced in this section the *quanta pushing* or more colorfully, the *decanter* method, is the basis for almost all results on the K-antirandom reals. In this section we will look only at the basic method to aid the reader with the more difficult nonuniform applications in subsequent sections. It is not difficult to show that K-trivial reals are *wtt*-incomplete. The following result proves that they are a more-or-less natural solution to Post's problem.

Theorem 15.1.22 (Downey, Hirschfeldt, Nies, Stephan [81]). *If a real α is K-trivial then α is Turing incomplete.*

15.1.7 The Decanter Method

In this section we will motivate a very important technique for dealing with K-trivial reals now called the decanter technique, and its later incarnation, the golden run machinery. It evolved from attempted proofs that there were Turing complete K-trivial reals, the blockage being turned around into a proof that no K-trivial real was Turing complete in the time-honoured symmetry of computability theory. Subsequently, the removal of artifacts of the original proof, and the use using a treelike structure by Nies and we have now what appears a generic technique for dealing with this area. The account below models the one from Downey, Hirschfeldt, Nies and Terwijn [82].

The following result shows that K-trivials solve Post's Problem. This will be greatly improved in the later sections.

Theorem 15.1.23 (Downey, Hirschfeldt, Nes Stephan [81]). *Suppose that A is K-trivial. Then A is Turing incomplete.*

The proof below runs the same way whether A is Δ^0_2 or computably enumerable. We only need the relevant approximation being $A = \cup_s A_s$ or $A = \lim_s A_s$.

15.1.8 The first approximation, *wtt*-incompleteness

The fundamental tool used in all of these proofs is what can be described as amplification. Suppose that A is K-trivial with constant of triviality b, and we are building a machine M whose coding constant within the universal machine U is known to be d.

Now the import of these constants is that if we describe n by some KC-axiom $\langle p, n\rangle$ meaning that we describe n by something of length p, then in U we describe n by something of length $p+d$ and hence the opponent at some stage s must eventually give a description of $A_s \upharpoonright n$ of length $p+b+d$. The reader should think of this as meaning that the opponent has to play *less quanta* than we do for the same effect.

What has this to do with us? Suppose that we are trying to claim that A is *not* K-trivial. Then we want to *force* U to issue too many descriptions of A, by using up all of its quanta.

The first idea is to make the opponent play many *times on the same length and hence amount of quanta.*

The easiest illustration of this method is to show that now K-trivial is *wtt*-complete.

Proposition 15.1.24. *Suppose that A is K-trivial then A is wtt-incomplete.*

Proof. We assume that we are given $A = \lim_s A_s$, a computable approximation to A. Using the Recursion Theorem, we build a c.e. set B, and a prefix free machine M. We suppose that $\Gamma^A = B$ is a weak truth table reduction with computable use γ. Again by the Recursion Theorem, we can know Γ, γ and we can suppose that the coding constant is d and the constant of triviality is b as above.

Now, we pick $k = 2^{b+d+1}$ many followers $m_k < dots < m_1$ targeted for B and wait for a stage where $\ell(s) > m_1$, $\ell(s)$ denoting the length of agreement of $\Gamma^A = B[s]$.

At this stage we will *load* an M-description of some *fresh, unseen* $n > \gamma(m_1)$ (and hence bigger than $\gamma(m_i)$ for all i) of size 1, enumerating an axiom $\langle 1, n \rangle$. The translation of course is that at some stage s_0 we must get a description of $A_{s_0} \upharpoonright n$ in U of length $c + d$ or less. That is at least $2^{-(b+d)}$ must enter the domain of U devoted to describing this part of A.

At the first such stage s_0, we can put m_1 into $B_{s_0+1} - B_{s_0}$ causing a change in $A \upharpoonright n - A_{s_0} \upharpoonright n$. (We remark that in this case we could use any of the m_i's but later it will be important that the m_i's enter in reverse order.) Then there must be at some stage $s_1 > s_0$, with $\ell(s_1) > m_1$, a new $A_{s_1} \upharpoonright n \neq A_{s_0} \upharpoonright n$ *also* described by something of length $b+d$. Thus U must have at least $2^{-(b+d)}$ more in its domain. If we repeat this process one time for each m_i then eventually we U runs out of quanta since $2^{-(b+d)}k > 1$. $\square$

15.1.9 The second approximation: impossible constants

The argument above is fine for weak truth table reducibility, but there are clearly problems in the case that Γ is a *Turing* reduction, for example.

That is, suppose that our new goal is to show that no K-trivial is Turing complete. The problem with the above construction is the following.

When we play the M-description of n, we have used all of our quanta available for M to describe a single number. *Now it is in the opponents power to move the use $\gamma(m_1, s)$ (or $\gamma(m_k, s)$ even) to some value bigger than n* before *even it decides to match our description of n.* Thus it costs him very little to match our M-description of n:

He moves $\gamma(m_k, s)$ then describes $A_s \upharpoonright n$ and we can no longer cause *any* changes of A below n, as all the gamma-uses are too big.

It is at this point that we realize it is pretty dumb of us to try to describe n in one. All that really matters is that we load lots of quanta beyond some point were it is measured many times. For instance, in the *wtt* case, we certainly could have used many n's beyond $\gamma(m_1)$ loading each with, say, 2^{-e} for some small e, and only attacking once we have amassed the requisite amount beyond $\gamma(m_1)$.

The is the idea behind our second step.

Impossible assumption: We will assume that we are given a Turing reduction $\Gamma^A = B$ and the overheads of the coding and Recursion Theorem result in a constant of 0 for the coding, and the constant of triviality is 0.

Hence we have $\Gamma^A = B[s]$ in some stage by stage manner, and moreover when we enumerate $\langle q, n\rangle$ into M then the opponent will eventually enumerate something of length q into U describing $A_s \upharpoonright n$. Notice that with these assumptions, in the wtt case we'd only need one follower m. Namely in the wtt case, we could load (e.g.) $\frac{7}{8}$ onto n, beyond $\gamma(m)$ and then put m into B causing the domain of U to need $\frac{7}{4}$ since we count $A_s \upharpoonright n$ for two different $A_s \upharpoonright n$-configurations.

In the case that γ is a Turing reduction, the key thing to note is that we still have the problem outlined above. Namely if we use the dumb strategy, then he will change $A_s \upharpoonright \gamma(m, s)$ moving some γ-use *before* he describes $A_s \upharpoonright n$. Thus he only needs to describe $\mathcal{A}_s \upharpoonright n$ *once*.

Here is where we use the drip feed strategy for loading. What is happening is that we really have a called procedure $P(\frac{7}{8})$ asking us to load $\frac{7}{8}$ beyond $\gamma(m)$ and then use m to count it twice. It might be that whilst we are trying to load some quanta, the change of the problem might happen, a certain amount of "trash", that is, axioms enumerated into M that do not cause the appropriate number of short descriptions to appear in U. We will need to show that this trash is small enough that it will not cause us problems.

Specifically, we would use a procedure $P(\frac{7}{8}, \frac{1}{8})$ asking for twice counted quanta (we call this a 2-*set*) of size $\frac{7}{8}$ but only having trash bounded by $\frac{1}{8}$.

Now, $\frac{1}{8} = \sum_j 2^{-(j+4)}$. Initially we might try loading quanta beyond the current use $\gamma(m, s_0)$ in lots of 2^{-4}. If we are successful in reaching our target of $\frac{7}{8}$ before A changes, then we are in the *wtt*-case and can simply change B to get the quanta counted twice.

Now suppose we load the quanta 2^{-4} on some $n_0 > \gamma(m, s_0)$. The opponent might at this very stage move $\gamma(m, s)$ to some new $\gamma(m, s_1) > n_0$, at essentially no cost to him. We would have played 2^{-4} for no gain, and would throw the 2^{-4} into the trash. Now we would begin to try to load anew $\frac{7}{8}$ beyond $\gamma(m, s_1)$ *but this time we would use chunks of size* 2^{-5}. Again if he moved immediately, then we would trash that quanta and next time use 2^{-6}. Notice that if we assume that $\Gamma^A = B$ this movement can't happen forever, lest $\gamma(m, s) \to \infty$.

On the other hand, in the first instance, perhaps we loaded 2^{-4} beyond $\gamma(m, s_0)$ and he did not move $\gamma(m, s_0)$ at that stage, but simply described $A \upharpoonright n_0$ by some description of size 4. At the next step, we would pick another n beyond $\gamma_(m, s_0) = \gamma(m, s_1)$ and try again to load 2^{-4}. If the opponent *now* changes, then we lose the *second* 2^{-4} but he *must* count the *first* one (on n_0) twice. That is, whenever he actually does not move $\gamma(m, s)$ then he must match our description of the current n, and this *will* later be counted *twice* since either *he* moves $\gamma(m,s)$ over it (causing it to me counted twice) *or* we put m into B making $\gamma(m,s)$ change.

Thus, for this simplified construction, each time we try to load, he either matches us(in which case the amount will contribute to the 2-set, and we can return $2^{-\text{current } \beta}$ where β is the current number being used for the loading to the target, *or* we lose β, but gain in that $\gamma(m,s)$ moves again, and we put β in the trash, but make the next $\beta = \frac{\beta}{2}$.

If $\Gamma^A = B$ then at some stage $\gamma(m,s)$ must stop moving and we will succeed in loading our target $\alpha = \frac{7}{8}$ into the 2-set. Our cost will be bounded above by $\frac{7}{8} + \frac{1}{8} = 1$.

15.1.10 The less impossible case

Now we will remove the simplifying assumptions. The key idea from the *wwt*-case where the use is fixed but the coding constants are nontrivial, is that we must make the changes beyond $\gamma(m_k)$ a $k-$set. Our idea is to combine the two methods to achieve this goal. For simplicity, suppose we pretend that the constant of triviality is 0, but now the coding constant is 1. Thus when we play 2^{-q} to describe some n , the opponent will only use $2^{-(q+1)}$.

Emulating the wtt-case, we would be working with $k = 2^{1+1} = 4$ and would try to construct a 4-set of changes. What we will do is break the task into the construction of a 2-set of a certain weight, a 3-set and a 4-set of a related weight in a coherent way.

We view these as procedures P_j for $2 \leqslant j \leqslant 4$ which are called in in reverse order in the following manner.

Our overall goal begins with, say $P_4(\frac{7}{8}, \frac{1}{8})$ asking us to load $\frac{7}{8}$ beyond $\gamma(m_4, s_0)$ initially in chunks of $\frac{1}{8}$, this being a 4-set.

To do this we will invoke the lower procedures. The procedure P_j $(2 \leqslant j \leqslant 4)$ enumerates a j-set C_j. The construction begins by calling P_4, which calls P_3 several times, and so on down to P_2, which enumerates the 2-set C_2 and KC set L of axioms $\langle q, n \rangle$.

Each procedure P_j has rational parameters $q, \beta \in [0,1]$. The *goal* q is the weight it wants C_j to reach, and the *garbage quota* β is how much it is allowed to waste.

In the simplified construction, where there was only one m, the goal was $\frac{7}{8}$ and the β evolved with time. The same thing happens here. P_4's goal *never* changes, and hence can never be met lest U use too much quanta. Thus A cannot compute B.

The main idea is that procedures P_j will ask that procedures P_i for $i < j$ do the work for them, with eventually P_2 "really" doing the work, but the the goals of the P_i are determined inductively by the garbage quotas of the P_j above. Then if the procedures are canceled before completing their tasks then the amount of quanta wasted is acceptably small.

We begin the construction by starting $P_4(\frac{7}{8}, \frac{1}{8}$. Its action will be to

1. Choose m_4 large.

2. Wait until $\Gamma^A(m_4)\downarrow$.

When this happens, P_4 will call $P_3(2^{-4}, 2^{-5})$. Note that here the idea is that P_4 is asking P_3 to enumerate the 2^{-4}'s which are the current quanta bits that P_4 would like to load beyond m_4's current Γ-use. (The actual numbers being used here are immaterial except that we need to make them converge, so that the total garbage will be bounded above.

Now if, while we are waiting, the Γ-use of m_4 changes, then we will go back to the beginning. But let's consider what happens on the assumption that this has not yet occurred.

What will happen is that $P_3(2^{-4}, 2^{-5})$ will pick some m_3 large, wait for $\Gamma(m_3)$ convergence, and then it will now invoke $P_2(2^{-5}, 2^{-6})$, say. This will pick its own number m_2 again large, wait for $\Gamma^A(m_2)\downarrow$ and finally now we will get to enumerate something into L. Thus, at this very stage we would try to load 2^{-5} beyond $\gamma(m_2, s)$ in lots of 2^{-6}.

Now whilst we are doing this, many things can happen. The simplest case is that nothing happens to the uses, and hence, as with the wtt case, we would successfully load this amount beyond $\gamma(m_2, s)$. Should we do this then we can enumerate m_2 into B and hence cause this amount to be a 2-set C_2 of weight 2^{-5} and we have reached our target.

This would return to P_3 which would realize that it now has 2^{-5} loaded beyond $\gamma(m_3, s)$, and it would like another such 2^{-5}. Thus it would again invoke $P_2(2^{-5}, 2^{-6})$. If it did this successfully, then we would have seen a 2-set of size 2^{-5} loaded beyond $\gamma(m_3, s)$ (which is unchanged) and hence if we enumerate m_3 into B we could make this a 3-set of size 2^{-5}, which would help P_4 towards its goals.

Then of course P_4 would need to invoke P_3 again and then down to P_2. The reader should think of this as "wheels within wheels within wheels" spinning ever faster.

Of course, the problems all come about because uses can change. The impossible case gave us a technique to deal with that. For example, if only the outer layer P_2 has its use $\gamma(m_2, s)$ change, then as we have seen, the amount already matched would still be a 2-set, but the latest attempt would be wasted. We would reset its garbage quota to be half of what it was, and then repeat. Then we could rely on the fact that (assuming that all the other procedures have m_i's with stable uses) $\lim_s \gamma(m_2, s) = \gamma(m_2)$ exists, eventually we get to build the 2-set of the desired target with acceptable garbage, build ever more slowly with ever lower quanta.

In general, the inductive procedures work the same way. Whilst waiting, if uses change, then we will initialize the lower procedures, reset their garbages to be ever smaller, but not throw away any work that has been successfully completed. Then in the end we can argue by induction that all tasks are completed.

Proof of Theorem 15.1.22. We are now ready to describe the construction. Let $k = 2^{b+d+1}$ and $c = b + d$. The method below is basically the same for all the constructions with one difference as we later see.

As in the wtt case, our construction will build a k-set C_k of weight $> 1/2$ to reach a contradiction.

The procedure P_j ($2 \leqslant j \leqslant k$) enumerates a j-set C_j. The construction begins by calling P_k, which calls P_{k-1} several times, and so on down to P_2, which enumerates L (and C_2).

Each procedure P_j has rational parameters $q, \beta \in [0, 1]$. The *goal* q is the weight it wants C_j to reach, and the *garbage quota* β is how much it is allowed to waste.

We now describe the procedure $P_j(q, \beta)$, where $1 < j \leqslant k$, and the parameters $q = 2^{-x}$ and $\beta = 2^{-y}$ are such that $x \leqslant y$.

1. Choose m large.
2. Wait until $\Gamma^A(m) \downarrow$.
3. Let $v \geqslant 1$ be the number of times P_j has gone through step 2.

   $j = 2$: Pick a large number n. Put $\langle r_n, n\rangle$ into L, where $2^{-r_n} = 2^{-v}\beta$. Wait for a *stage* t such that $K_t(n) \leqslant r_n + d$, and put n into C_1. (If M_d is a prefix-free machine corresponding to L, then t exists.)

   $j > 2$: Call $P_{j-1}(2^{-v}\beta, \beta')$, where $\beta' = \beta 2^{j-k-w-1})$ and w is the number of P_{j-1} procedures started so far.

   In any case, if weight$(C_{j-1}) < q$ then repeat step 3, and otherwise return.
4. Put m into B. This forces A to change below $\gamma(m) < \min(C_{j-1})$, and hence makes C_{j-1} a j-set (if we assume inductively that C_{j-1} is a $(j-1)$-set). So put C_{j-1} into C_j, and declare $C_{j-1} = \emptyset$.

If $\gamma^A(m)$ changes during the execution of the loop at step 3, then cancel the run of all subprocedures, and go to step 2. Despite the cancellations, C_{j-1} is now a j-set because of this very change. (This is an important point, as it ensures that the measure associated with numbers already in C_{j-1} is not wasted.) So put C_{j-1} into C_j, and declare $C_{j-1} = \emptyset$.

This completes the description of the procedures. The construction consists of calling $P_k(\frac{7}{8}, \frac{1}{8})$ (say). It is easy to argue that since quotas are inductively halved each time they are injured by a use change, they are bounded by, say $\frac{1}{4}$. Thus L is a KC set. Furthermore C_k is a k-set, and this is a contradiction since then the total quanta put into U exceeds 1. $\square$

The following elegant description of Nies' is taken from [82]:

We can visualize this construction by thinking of a machine similar to Lerman's pinball machine (see [280, Chapter VIII.5]). However, since we enumerate rational quantities instead of single objects, we replace the balls in Lerman's machine by amounts of a precious liquid, say 1955 Biondi-Santi Brunello wine. Our machine consists of decanters $C_k, C_{k-1}, \ldots, C_0$. At any stage C_j is a j-set. We put C_{j-1} above C_j so that C_{j-1} can be emptied into C_j. The height of a decanter is changeable. The procedure $P_j(q, \beta)$ wants to add weight q to C_j, by filling C_{j-1} up to q and then emptying it into C_j. The emptying corresponds to adding one more A-change.

The emptying device is a hook (the $\gamma^A(m)$-marker), which besides being used on purpose may go off finitely often by itself. When C_{j-1} is emptied into C_j then $C_{j-2}, \ldots, C_0$ are spilled on the floor, since the new hooks emptying $C_{j-1}, \ldots, C_0$ may be much longer (the $\gamma^A(m)$-marker may move to a much bigger position), and so we cannot use them any more to empty those decanters in their old positions.

We first pour wine into the highest decanter C_0, representing the left domain of L, in portions corresponding to the weight of requests entering L. We want to ensure that at least half the wine we put into C_0 reaches C_k. Recall that the parameter β is the amount of garbage $P_j(q, \beta)$ allows. If v is the number of times the emptying device has gone off by itself, then P_j lets P_{j-1} fill C_{j-1} in portions of size $2^{-v}\beta$. Then when C_{j-1} is emptied into C_j, at most $2^{-v}\beta$ much liquid can be lost because of being in higher decanters $C_{j-2}, \ldots, C_0$. The procedure $P_2(q, \beta)$ is special but limits the garbage in the same way: it puts requests $\langle r_n, n\rangle$ into L where $2^{-r_n} = 2^{-v}\beta$. Once it sees the corresponding $A \upharpoonright n$ description, it empties C_0 into C_1 (but C_0 may be spilled on the floor before that because of a lower decanter being emptied).

15.1.11 K-trivials form is a robust class

It turns out that the K-trivials are a remarkable robust class, and coincide with a host of reals defined in other ways. This also has significant degree-theoretical implications. For example, as we see, not only are the K-trivials Turing incomplete, but are closed downwards under Turing reducibility and form a natural Σ^0_3 ideal in the Turing degrees.

What we need is an improved version of the decanter method. In the previous chapter it was shown that K-trivials solve Post's Problem. Suppose however, we actually applied the method above to a K-trivial and a partial functional Γ. Then what would happen would be that for some i the procedure P_i *would not return.* This idea forms the basis for most applications of the decanter method, and the run that does not return would be called a *golden run.*

For instance, suppose that we wanted to show Nies' result that all K-trivials are superlow.

Let A be K-trivial. Our task is to build a functional $\Gamma^K(e)$ computing whether $\Phi_e^A(e)\downarrow$. For ease of notation, let us denote $J^A(e)$ to be the partial function that computes $\Phi_e^A(e)$. Now the obvious approach to this task is to monitor $J^A(e)[s]$. Surely if the never halts then we will never believe that $J^A(e)\downarrow$. However, we are in a more dangerous situation when we see some stage s where $J^A(e)[s]\downarrow$. If we define $\Gamma^K(e) = J(e)[s]$ then we risk the possibility that this situation could repeat itself many times since it is in the opponent's power the changes $A_s \upharpoonright \varphi_e(e,s)$.

Now if *we* were building A then we would know what to do. We should *restrain* A in a familiar way and hence with finite injury A is low. However, the *opponent* is building A, and all we know is that A is K-trivial.

The main idea is to load up quanta beyond the use of the e-computation, before we change the value of $\Gamma^A(e)[s]$, that is changing our belief from divergence to conergence. Then, if A were to change on that use after we had successfully loaded, negating our belief, and causing us to reset $\Gamma^K(e)$ it would cost the opponent.

As with the proof above, this cannot happen too many times for any particular argument, and in the construction to be described, there will be a golden run which does not return. The interpretation of this non-returning is that the $\Gamma^K(e)$ that the run R builds will actually work. Thus, the construction of the lowness index is *non-uniform.*

Thus, the new idea idea would be to have *tree* of possibilities. The height of the tree is $k = 2^{b+d+1}$, and the tree is ω branching. A node $\sigma\widehat{\ }i$ denotes the action to be performed for $J^A(i)$ more or less assuming that noe σ is the highest priority node that does not return.

At the top level we will be working at a procedure $P_k(\frac{7}{8},\frac{1}{8})$ yet again, and we know in advance that this won't return with a k-set of that size.

What we do is distribute the tasks out to the successors of λ, the empty node. Thus outcome e would be devoted to solving $J^A(e)$ via a k-set. It will be given quanta, say, $2^{-(e+1)}\alpha_k$ where $\alpha_k = \frac{7}{8}$. (Here this choice is arbitary, save that it is suitably convergent. For instance, we could ude the series $\sum_{n\in\mathbb{N}}\frac{1}{n^2}$ which would sharpen the norms of the tt-reductions to n^2.) To achieve its goals, when it sees some apparent $J^(A)(e)\downarrow[s]$, It will invoke $P_k(2^{-(e+1)}\alpha_k, 2^{-e}\beta_k)$ where $\beta_k = \frac{1}{8}$. We denote this procedure by $P_k(e, 2^{-(e+1)}\alpha_k, 2^{-(e+1)}\beta_k)$. These will look for a k-set of the appropriate size, which, when it achieves its goal, the version of Γ at the empty string says it believes.

Again, notice that if this P_k returns (and this is the idea below) 2^{e+2} many times then $P_k(\alpha_k,\beta_k)$ would return, which is impossible.

As with the case of Theorem 15.1.22, to achieve its goals, *before it believes that* $J^A(e)\downarrow[s]$, it needs to get its quanta by invoking the team so nodes below e. These are, of course of the form $e\widehat{\ }j$ for $j\in\omega$. They will be asked to try to achieve a $k-1$ set and $P_k(e, 2^{-(e+1)}\alpha_k, 2^{-(e+1)}\beta_k)$ by using $P_{k-1}(e\widehat{\ }j, 2^(e+1)2^{-(j+1)}\beta_k, 2^{-(j+1)}2^{-(e+1)}\beta_{k-1})$, with $\beta_{k-1} << \beta_k$ and j

chosen appropriately. Namely we will choose those j, say, with $j > \varphi_e^A(e)[s]$. That is, thses j will, by convention, have their uses beyond $\varphi_e^A(e)[s]$ and hence will be working similarly to the "next" m_i "down" in the method of the incompleteness proof of Theorem 15.1.22. Of course such procedures would await $\varphi_e^A(j)[s]$ and try to load quanta in the form of a $k-1$ set beyond the $\varphi_j^A(j,s)$ $(> \varphi_e^A(e))$, the relevant j-use.

The argument procedures working in parallel work their way down the tree. As above when procedure $\sigma\widehat{\ }i$ is injured because the $J^A(t)[s]$ is unchanged for all the uses of $t \in \sigma$, yet $J^A(i)[s]$ changes before the procedure returns then we reset all the garbage quotas in a systematic way, so as to make the garbage quota be bounded.

Now the argument is the same. There is some m least σ of length m, and some final α, β for which $P_m(e, \alpha, \beta)$ is invoked and never returns. Then the procedure built at σ will be correct on all $j > \varphi_e^A(e)$. Moreover, we can always calculate how many times it would be that some called procedure would be invoked to fulfil $P_m(\alpha, \beta)$. Thus the can bound the number of times that we would change our mind on $\Gamma^K_{P_m(\alpha,\beta)}(i)$ for any argument i. That is, A is superlow. In fact, as Nies pointed out, this gives a little more. Recall that an *order* is a computable nondecreasing function with infinite limit.

Definition 15.1.25 (Nies [225, 226])**.** We say that a set B is jump traceable iff there is a computable order h and a weak array $i\{W_{g(j)} : j \in \mathbb{N}\}$, such that $|W_{g(j)}| < h(j)$, and $J^B(e) \in W_{g(e)}$.

Theorem 15.1.26 (Nies [225, 226])**.** *Suppose that A is K-trivial. Then A is jump traceable.*

The proof is to observe that we are actually constructing a trace. We remark that Nies [225], and Figueira, Nies and Stephan, [105] investigated the notion of jump traceability in its own right. It is not too hard to prove that the notion of jump traceability and superlowness *coincide* for c.e. sets, but differ on Δ_2^0 sets.

A mild variation of the proof above also shows the following.

Theorem 15.1.27 (Nies [225])**.** *Suppose that A is K-trivial. Then there exists a K-trivial computably enumerable B with $A \leqslant_{tt} B$.*

Proof. (sketch) Again the golden run proof is more or less the same, our task being to build B. This is farmed out to outcomes e in the ω-branching tree, where we try to build $\Gamma^B(e) = A(e)$. Again at level j, the size of the use will be determined by the number of times the module can act before it returns enough quanta to give the node above the necessary $j-1$ set. This is a computable calculation. Now when the opponent seeks to load qanta beyond $\gamma^j(e,s)$ before we believe this, we will load matching quanta beyond e for A. The details are then more or less the same. □

Other similar arguments show that K-triviality is basically a *computably enumerable* phenomenom. That is the following is true.

Theorem 15.1.28 (Nies [225])**.** *The following are equivalent*

(i) A is K-trivial.

(ii) A has a Δ_2^0 approximation $A = \lim_s A_s$ which reflects the cost function construction. That is,

$$\{ \sum_{x \leqslant y \leqslant s} \frac{1}{2} c(y,s) : x \text{ minimal } A_s(x) \neq A_{s-1}(x) \} < \frac{1}{2}.$$

15.1.12 More characterizations of the K-trivials

We have seen that the K-trivials are all jump traceable. In this ection, we sketch the proofs that the class is characterized by other "antirandomness" properties.

Theorem 15.1.29 (Nies and Hirschfeldt [227])**.** *Suppose that A is K-trivial. Then A is low for K.*

Corollary 15.1.30. *The following are equivalent:*

(i) A is K-trivial.

(ii) A is low for Martin-Löf randomness.

(iii) A is low for K.

Proof. The corollary is immediate by the implication (iii)→(ii)→(i). We prove Theorem 15.1.29. Again this is another golden run construction. This proof proceeds in a similar way to that showing that K-trivials are low, except that $P_{j,\tau}$ calls procedures $P_{j-1,\sigma}$ based on computations $U^A(\sigma) = y[s]$ (since we now want to enumerate requests $\langle |\sigma| + d, y \rangle$), and the marker $\gamma(m,s)$ is replaced by the use of this computation. That is, we wish to believe a computation, $U^A(\sigma) = y[s]$ and to do so we want to load quanta beyond the use $u(\sigma, s)$. This is done more or less exactly the same way, beginning at P_k and descending down the nodes of the tree. Each node ν will this time build a machine $\widehat{U}_\nu$, which will copy ν-believed computations; namely those for which we have successfully loaded the requisite $|\nu|-$set. We need to argue that for the golden ν, the machine is real. The garbage is bounded by, say, $\frac{1}{8}$ by the way we reset it. The machine otherwise is bounded by U itself. □

Corollary 15.1.31 (Nies [226])**.** *The K-trivials are closed downward under $\leqslant_T$.*

We finish this section by proving a rather useful characterization of Martin-Löf lowness.

Theorem 15.1.32 (Nies and Stephan [**?**]). *A is low for Martin-Löf randomness iff*

$$\exists R \text{ c.e. open } (\mu(R) < 1 \wedge \forall z \in 2^{<\omega}[K^A(z) \leqslant |z|-1 \Rightarrow [z] \subseteq R]). \quad (15.1)$$

Proof. (Nies and Stephan [**?**]) Define $U_n^X = \{[z] : \exists m \leqslant |z| : K(z \upharpoonright m) \leqslant m - n\}$, being a X-universal Martin-Löf test. Then X is Martin-Löf low if there is an unrelativized test $\{V_n : n \in \mathbb{N}\}$ with $\cap_n U_n^X \subseteq \cap_n V_n$. By Theorem 11.9.1, this is equivalent to $\cap_n U_n^X \subseteq R$ for some Σ_1^0 class R, with $\mu(R) < 1$, as required.

To prove the other direction, suppose that A is low for Martin-Löf randomness. Take a Σ_1^0 class V of measure below 1, with V containing all the nonrandom reals. The claim is that for some σ with $[\sigma] \notin V$, and some k,

$$\forall \tau \supseteq \sigma (K^A(\tau) \leqslant |\tau| - k \to [\tau] \in V).$$

If not the we can define a real $z = \lim_s z_s$ in stages as follows. Let $z_0 = \lambda$. Having defined z_k define z_{k+1} to be a preper extension of z_k with $[z_{k+1}] \notin V$. Then by construction, and the relativized version of Schnorr's Theorem, z is not A-random, yet z is 1-random as $z \notin V$. Thus we can fix σ, k with $\forall \tau \supseteq \sigma (K^A(\tau) \leqslant |\tau| - k \to [\tau] \in V)$. Now define $R = \{\tau : [\sigma\tau] \in V\}$. Then $[\overline{R}]$ is a Π_1^0 class contained in the random sets and hence cannot have measure 0. Thus $\mu R < 1$. Finally, $K^A(\sigma\tau) \leqslant K^A(\sigma) + K^A(\tau) + \mathcal{O}(1)$, it follows that for all τ,

$$K^A(\tau) \leqslant |\tau| - b \to K^A(\sigma\tau) \leqslant |\sigma\tau| - m \to [\sigma\tau] \in V \to [\sigma] \in R.$$

□

15.1.13 Bases of cones of 1-randomness

It turns out that the fact that K-trivials solve Post's problem is also a corollary of Kučera's priority-free solution to Post's problem. (Theorem **??**) This fact was first realized by Hirschfeldt, Nies and Stephan.
DENIS

15.1.14 Lowness for Kurtz 2-randomness

Recently there have been some further characterizations of the K-trivials. One that is fairly counter-intuitive is the following.

Definition 15.1.33 (Downey, Nies, Weber, Yu [**?**]). We say that A is low for weak 2-randomness tests iff for all Σ_2^A generalized Martin-Löf tests (i.e. open sets $\{U_n : n \in \mathbb{N}\}$ with $\mu(U_n) \to 0$, there exists a generalized Martin-Löf test $\{\widehat{U}_n : n \in \mathbb{N}\}$ with

$$\cap_{n \in \mathbb{N}} \widehat{U}_n \supseteq \cap_{n \in \mathbb{N}} U_n.$$

Downey, Nies, Weber and Yu showed that there were c.e. sets that were low for weak 2 randomness tests, and hence low for Kurtz 2-randomness since they would not make any Kurtz 2-random real not Kurtz 2-A-random. They also proved the following result.

Theorem 15.1.34 (Downey, Nies, Weber, Yu [?])**.** *Suppose that A is low for low for weak 2 randomness tests. Then A is K-trivial.*

The folwoing proof uses Nies' idea of lowness for multiple classes.

Definition 15.1.35 (Nies)**.** A set A is in $\mathrm{Low}(\mathcal{C}, \mathcal{D})$ if $\mathcal{C} \subseteq \mathcal{D}^A$.

The point here is that relativizing $\mathcal{D}$ usually makes it smaller. Thus, we would expect that in general $\mathcal{C} \not\subseteq \mathcal{D}^A$ even if $\mathcal{C} \subseteq \mathcal{D}$. The lowness class of Definition 15.1.35 consists of the sets A for which the inclusion still holds.

Let $\mathsf{MLRand}, \mathsf{W2Rand}$ denote the classes of ML-random and weakly 2-random sets, respectively.

Theorem 15.1.36. $\mathrm{Low}(\mathsf{W2Rand}, \mathsf{MLRand}) = \mathrm{Low}(\mathsf{MLRand})$*. In other words, if each weakly 2 random is Martin-Löf random relative to A, then A is in fact low for Martin-Löf randomness.*

Since every Martin-Löf test is a generalized Martin-Löf test, $\mathsf{W2Rand}^A \subseteq \mathsf{MLRand}^A$ for any A. Thus having $A \in \mathrm{Low}(\mathsf{W2Rand}) - \mathrm{Low}(\mathsf{MLRand})$ would contradict Theorem 15.1.36 and a corollary to the theorem is that every real which is low for weak 2 randomness is low for Martin-Löf randomness. (Note that here we mean the broader notion of low for random, rather than the (possibly) more restrictive low for tests.)

We recall Merkle's characterization of Martin-Löf randomness, Theorem 9.3.12.

Theorem 15.1.37 (Merkle)**.** *If $Z = z_0 z_1 z_2 \dots$ where $K(z_i) \leqslant |z_i| - 1$ for each i, then Z is not Martin-Löf random.*

We recall from Theorem 15.1.32 that a set A is low for Martin-Löf randomness iff

$$\exists R \text{ c.e. open } (\mu(R) < 1 \wedge \forall z \in 2^{<\omega}[K^A(z) \leqslant |z| - 1 \Rightarrow [z] \subseteq R]). \quad (15.2)$$

We will use a consequence of the failure of (15.2). For an open set V and a string w, the conditional measure $\mu(V \mid w)$ is $2^{|w|}\mu(V \cap [w])$.

Claim 15.1.38. *Suppose (15.2) fails for A. Let β, γ be rationals such that $\beta < \gamma < 1$. For each c.e. open set V and each string w, if $\mu(V \mid w) \leqslant \beta$, then there is z such that $K^A(z) \leqslant |z| - 1$ and $\mu(V \mid wz) \leqslant \gamma$.*

Proof. Suppose that no such z exists, and consider the c.e. set of strings

$$G = \{z : \mu(V \mid wz) > \gamma\}.$$

Whenever $K^A(z) \leqslant |z| - 1$ then $z \in G$. Let R be the c.e. open set generated by G. Note that $z0, z1 \in G \Rightarrow z \in G$. So if $(z_i)_{i<N}$ is a listing of the minimal strings in G ($N \leqslant \infty$), then $R = \bigcup_{i<N}[z_i]$.

Now

$$\beta \geqslant \mu(V \mid w) \geqslant \sum_{i<N} 2^{-|z_i|}\mu(V \mid wz_i) \geqslant \mu(R) \cdot \gamma.$$

Thus $1 > \beta/\gamma \geqslant \mu R$ and (15.2) holds, contradiction. □

Proof of Theorem 15.1.36. Suppose that A is not low for ML-random. Thus the hypothesis of Claim 15.1.38 is satisfied. We show that $\mathsf{W2Rand} \subseteq \mathsf{MLRand}^A$ fails, by building a set $Z \in \mathsf{W2Rand}$ that is not ML-random relative to A. We define (noneffectively) a sequence of strings $z_0, z_1, \ldots$ such that $K^A(z_i) \leqslant |z_i| - 1$ and let $Z = z_0 z_1 z_2 \ldots$, so that Z is not ML-random relative to A by Theorem 9.3.12 relativized to A.

Let $\{U_{e,n}\}_{e,n\in\omega}$ be an enumeration of all potential generalized Martin-Löf tests. For $Z \in \mathsf{W2Rand}$, for each actual GML test $\{U_{e,n}\}$ we define a number n_e and ensure $Z \notin U_{e,n_e}$. At the beginning of Step e, $z_0, \ldots, z_{e-1}$ have been defined, and we let

$$V_e = \bigcup_{\substack{i<e \\ n_i \text{ defined}}} U_{i,n_i},$$

and $w_e = z_0 \ldots z_{e-1}$. We ensure inductively that

$$\mu(V_e \mid w_e) \leqslant \gamma_e := 1 - 2^{-e}. \tag{15.3}$$

In particular, since $\mu(V_e|w_e) < 1$, $[w_e] \nsubseteq V_e$ for each e. Since the V_e are open and nested, $V_e \subseteq V_{e+1}$ for all e, this is sufficient to give $Z \notin U_{e,n_e}$ whenever $\{U_{e,n}\}$ is a test, as required. To see this, note that $Z \in U_{e,n_e}$ requires some initial segment $w_m \subset Z$ be such that $[w_m] \subseteq U_{e,n_e}$ (This holds without loss of generality, and in fact necessarily $m > n_e$). However, our guarantee of $[w_{m+1}] \nsubseteq V_{m+1}$, $w_{m+1} \supset w_m$, contradicts $[w_m] \subseteq U_{e,n_e} \subseteq V_{m+1}$, so $Z \notin U_{e,n_e}$.

Note that w_0 is the empty string and $V_0 = \emptyset$, so that (15.3) holds for $e = 0$.

Step $e \geqslant 0$. If $\{U_{e,n}\}_{n\in\omega}$ is not a test (i.e., $\lim_n \mu(U_{e,n}) \neq 0$), then leave n_e undefined. Otherwise, choose n_e so large that

$$\mu(U_{e,n_e}) \leqslant 2^{-|w_e|-e-2}.$$

In particular, $\mu(U_{e,n_e}|w_e) \leqslant 2^{-(e+2)}$.

Then letting $V_{e+1} = V_e \cup U_{e,n_e}$, we get

$$\mu(V_{e+1} \mid w_e) \leqslant \gamma_e + 2^{-(e+2)} = 1 - 2^{-e} + 2^{-(e+2)} < 1.$$

Applying Claim 15.1.38 to $V = V_{e+1}$, $w = w_e$, $\beta = \gamma_e + 2^{-(e+2)}$, and $\gamma = \gamma_{e+1} > \beta$, there is $z = z_e$ such that $K^A(z) \leqslant |z| - 1$ and $\mu(V_{e+1} \mid w_e z) \leqslant \gamma_{e+1}$. Thus (15.3) holds for $e + 1$. □

It had been strongly suspected that the class of sets low for weak 2 randomness was a proper subclass of the K-trivials. This hypothesis was surprisingly disproved by Miller, and independently Nies..

Theorem 15.1.39 (Nies, Miller)**.** *Suppose that A is K-trivial. Then A is low for weak 2 randomness tests. Therefore K-triviality is equivalent to low for Kurtz 2-randomness.*

The original proofs of Theorem 15.1.39 used the decanter method, but there is a more interesting and informative proof using Kjos-Hanssen's result about the reducibility $\leqslant_{\mathcal{L}R}$ of Chapter 14. Recall from definition 14.4.2, that $A \leqslant_{\mathcal{L}R} B$ means that each B-random real is A-random. We recall the result of Kjos-Hanssen (Theorem **??**).

Theorem 15.1.40 (Kjos-Hanssen [**?**])**.** *$A \leqslant_{\mathcal{L}R} B$ iff every Π^A_1 class of positive measure has a Π^B_1 subclass of positive measure.*

Recall $A \leqslant_{LK} B$ means that for all σ, $K^B(\sigma) \leqslant K^A(\sigma) + \mathcal{O}(1)$.

Theorem 15.1.41 (Binns, Kjos-Hanssen, Miller, Solomon [**?**])**.** *If $A \leqslant_{\mathcal{L}R} B$ then $A \leqslant_{LK} B$.*

Notice that this result gives one of Nies' implications above, by taking $B = \emptyset$.

Corollary 15.1.42 (Nies [225])**.** *A is low for Martin-Löf randomness iff A is low for K.*

We need the following easy lemma from basic analysis.

Lemma 15.1.43. *Suppose that $(a_i)_{i\in\omega}$ is a sequence of real numbers with $0 \leqslant a_i \leqslant 1$. Then $\Pi_{i\in\omega}(1 - a_i) > 0$ iff $\sum_{i\in\omega} a_i$ converges.*

Proof. (of Theorem 15.1.41) For each (n, τ) we will define a finite set $V_{n,\tau}$ as follows. Let $s = \langle n, \tau\rangle$ and suppose that V_t is already defined for all $t < s$. Let m denote the length of the longest string in $\cup_{t<s} V_t$. Now we define $V_s = \{\sigma 0^m : \sigma \in 2^m\}$. Then for $I \subseteq \omega \times 2^{<\omega}$,

$$\mu(\cap_{s\in I}\overline{[V_s]}) = \Pi_{(n,\tau)\in I}(1 - 2^{-n}).$$

In particular each V_s is independent for the others.

Let $I = \{(|\sigma|, \tau) : U^A(\sigma) = \tau\}$. Then I is A-c.e. and hence $P = \cap_{s\in I}\overline{[V_s]}$ is a Π^A_1 class. Also $\sum_{(n,\tau)\in I} 2^{-n} \leqslant \sum_{\sigma\in\mathrm{dom}(U^A)} 2^{-|\sigma|} \leqslant 1$ by Kraft's inequality. Thus, as $(0, \tau) \notin I$ for any τ and hence $\mu(P) = \Pi_{(n,\tau)\in I}(1 - 2^{-n}) > 0$, by Lemma 15.1.43. Therefore, by Kjos-Hanssen's result, Theorem **??**, there is a Π^B_1 class $Q \subseteq P$ with $\mu(Q) > 0$.

Let $J = \{(n, \tau) : [V_{(n,\tau)} \cap Q = \emptyset\}$. Then J is a B-c.e. set. Also $\Pi_{(n,\tau)\in J}(1 - 2^{-n}) = \mu(\cap_{s\in J}\overline{[V_s]}) \geqslant \mu(Q) > 0$. Thus by Lemma 15.1.43, $\sum_{(n,\tau)\in J} 2^{-n}$ converges, say to below 2^c $c \in \mathbb{N}$. Then we now have a B-KC set $\widehat{J} = \{(n + c, \tau) : \tau \in J\}$. Now $(n, \tau) \in J$ iff $(n + c, \tau) \in \widehat{J}$ which implies $K^B(\tau) n + c + \mathcal{O}(1) \leqslant n + \mathcal{O}(1)$. But then, $Q \subseteq P$ implies $J \subseteq I$, and hence $(K^A(\tau), \tau) \in J$ for each $\tau \in 2^{<\omega}$. Consequently, $K^B(\tau) \leqslant K^A(\tau) + \mathcal{O}(1)$, meaning $A \leqslant_{LK} B$. $\square$

Using similar techniques, we can also establish the following lemma.

Lemma 15.1.44 (Binns, Kjos-Hanssen, Miller, and Solomon [**?**])**.** *Suppose that $A \leqslant_T B'$ and $A \leqslant_{LR} B$. Then every Π_1^A class has a Σ_2^B subclass of the same measure.*

Proof. Again we define V_t inductively. Let $s = \langle \sigma, \tau \rangle \in 2^{<\omega} \times 2^{<\omega}$ and let k denote the longstest string in $\cup_{t<s} V_t$. Define $V_s = \{\nu 0^{|\tau|} : \nu \in \underline{2^k}\}$. As in the proof of Theorem 15.1.41, if $I \subseteq 2^{<\omega} \times 2^{<\omega}$, then $\mu(\cap_{s \in I} \overline{[V_s]}) = \Pi_{(\sigma,\tau)\in I}(1 - 2^{-|\tau|})$.

Now suppose that $X \neq \emptyset$ is a Π_1^A class. Let S^A be a A-c.e. prefix free set of strings with $X = 2^\omega - [S^A]$, with $\lambda \notin S^A$. Let $I = \{(\sigma, \tau) : \tau \in S^A$ with use $\sigma\}$. Let P be the Π_1^A class $P = \cap_{s\in I} \overline{[V_s]}$. Again we have $\mu(P) \leqslant 1$ be the Karft inequality since $\sum_{(\sigma,\tau)\in I} 2^{-|\tau|} = \sum_{\tau \in S^A} 2^{-|\tau|}$. Again we invoke Lemma 15.1.43 to see that $\mu(P) = \Pi_{(\sigma,\tau)\in I}(1 - 2^{-|\tau|}) > 0$. Thus by Kjos-Hanssen's theorem **??**, we have that there is a Π_1^B class $Q \subseteq P$ with $\mu(Q) > 0$.

This time we define the B-c.e. set $J = \{(\sigma, \tau) : [V_{\sigma,\tau}] \cap Q = \emptyset\}$. We note again that $\Pi_{(\sigma,\tau)\in J}(1 - 2^{-|\tau|} = \mu(\cap_{s\in J} \overline{[V_s]}) > 0$. Thus by Lemma 15.1.43, $\sum_{(\sigma,\tau)\in J} 2^{-|\tau|}$ converges. Let $A = \lim_s A_s$ be a B-computable sequence ahose limit is A given by the Limit Lemma. Now we define

$$T_s = \{(\sigma, \tau) \in J : \exists t \geqslant s(\tau \in S_t^{A_t} \text{ with use } \sigma\}.$$

Now let

$$U_s = \{\tau : \exists \sigma[(\sigma, \tau) \in T_s\}.$$

Notice that $i\{T_s : s \in \mathbb{N}\}$ and $\{U_s : s \in \mathbb{N}\}$ are computable sequnces of B-c.e. sets. The claim is that $Y = \cup_{s\in\mathbb{N}} \overline{[U_s]}$ is the desired Σ_2^B class.

To see this, first note that $S^A \subseteq U_s$ for all s and hence $Y \subseteq X$. Then, for each $(\sigma, \tau) \in T_0 - I$, there is a final stage t such that σ is a prefix of A_t lest $(\sigma, \tau) \in I$. Then for any stage $s > t$, $(\sigma, \tau) \notin T_s$. Now we fix $\varepsilon > 0$ and choose n sufficiently large so that $\sum_{(\sigma,\tau)\in J, \langle(\sigma,\tau)>n} 2^{-|\tau|} < \varepsilon$. Choose s large enough that $(\sigma, \tau) \in T_0 - I$ and $(\sigma, \tau) < n$ implies $(\sigma, \tau) \notin T_s$. Then we would have

$$\mu(X - \overline{[U_s]}) \leqslant \sum_{\tau \in U_s - S^A} 2^{-|\tau|} \leqslant \sum_{(\sigma,\tau)\in T_s - I} 2^{-|\tau|} \leqslant \sum_{(\sigma,\tau)\in J, \langle(\sigma,\tau)>n} 2^{-|\tau|} < \varepsilon.$$

Since $\varepsilon > 0$ is chosen arbitarily, $\mu(X) = \mu(Y)$. $\square$

Theorem 15.1.45 (Binns, Kjos-Hanssen, Miller, and Solomon [**?**])**.** *The following are equivalent.*

(i) $A \leqslant_T B'$ *and* $A \leqslant_{LR} B$.

(ii) Every Π_1^A *class has a* Σ_2^A *class of the same measure.*

(iii) Every Σ_2^A *class has a* Σ_2^B *subclass of the same measure.*

Proof. The fact that (i) implies (ii) is Lemma **??**. To see that (ii) implies (iii), let W be a Σ_2^A class and let $W = \cup_{i \in \mathbb{N}} X_i$ for Π_1^A classes X_i. Now consider the Π_1^A class $X = \{0^i 1\alpha : i \in \mathbb{N}$ and $\alpha \in X_i\}$. Now apply (ii). There is a Σ_2^B subclass Y of X of the same measure. For each i, let $Y_i = \{\alpha : 0^i 1\alpha \in Y\}$. Then Y_i is a Σ_2^B class with $Y_i \subseteq X_i$ for all i. Also $\mu(Y_i) \leqslant \mu(X_i)$. Notice that if $\mu(Y_i) < \mu(X_i)$ for some i, then

$$\mu(Y) = \sum_{i \in \omega} 2^{i+1} \mu(Y_i) < \sum_{i \in \omega} 2^{i+1} \mu(X_i) = \mu(X),$$

which is a contradiction. Thus, $\mu(X_i) = \mu(Y_i)$ for all i. Now define $Z = \cup_{i \in \omega} Y_i$. Thus Z is a Σ_2^B class and $Z \subseteq W$. Finally, $\mu(W - Z) \leqslant \sum_{i \in \omega} \mu(X_i - Y_i) = 0$, and hence $\mu(Z) = \mu(W)$. □

We remark that it is shown in Binns, Kjos-Hanssen, Miller and Solomon [**?**] that $A \leqslant_{LR} B$ does *not* imply $A \leqslant_T B'$.

Now we can prove Theorem **??** as a corollary. Suppose that A is K-trivial. Then $A \leqslant_{\emptyset}'$ and $A \leqslant_{LR} \emptyset$. Hence by Theorem 15.1.45, (iii), every Σ_2^A class has a Σ_2^0 subclass of the same measure.

DENIS: domination here also?

15.1.15 K-trivials are closed under $+$

In this section we prove the following.

Theorem 15.1.46 (Downey, Hirschfeldt, Nies and Stephan [81])**.** *If α and β are K-trivial then so is $\alpha + \beta$.*

Proof. Assume that α, β are two K-trivial reals. Then there is a constant c such that $K(\alpha \upharpoonright n)$ and $K(\beta \upharpoonright n)$ are both below $K(n) + c$ for every n. By Theorem 15.1.8 there is a constant d such that for each n there are at most d strings $\tau \in \{0,1\}^n$ satisfying $K(\tau) \leqslant K(n) + c$. Let e be the shortest program for n. One can assign to $\alpha \upharpoonright n$ and $\beta \upharpoonright n$ numbers $i, j \leqslant d$ such that they are the i-th and the j-th string of length n enumerated by a program of length up to $|e| + c$.

Let U be a universal prefix-free machine. We build a prefix-free machine V witnessing the K-triviality of $\alpha + \beta$. Representing i, j by strings of the fixed length d and taking $b \in \{0,1\}$, $V(eijb)$ is defined by first simulating $U(e)$ until an output n is produced and then continuing the simulation in order to find the i-th and j-th string α and β of length n such that both are generated by a program of size up to $n + c$. Then one can compute $2^{-n}(\alpha + \beta + b)$ and derive from this string the first n binary digits of the real $\alpha + \beta$. These digits are correct provided that e, i, j are correct and b is the carry bit from bit $n + 1$ to bit n when adding α and β – this bit is well-defined unless $\alpha + \beta = z \cdot 2^{-m}$ for some integers m, z, but in that case $\alpha + \beta$ is computable and one can get the first n bits of $\alpha + \beta$ directly without having to do the more involved construction given here. □

From this result Downey, Hirschfeldt, Nies and Stephan [81] proved that the *wtt*-degrees of K-trivial reals forms an ideal. However, in Corollary 15.1.31, it is whown that the K-trivials are closed downward under $\leqslant_T$. This gives us the following theorem.

Theorem 15.1.47 (Nies [226]). *The K-low degrees form a Σ_3^0 ideal in the computably enumerable Turing degrees.*

15.2 Listing the K-trivials

We next prove a result about the presentation of the class $\mathcal{K}$ of K-trivial reals. First consider the computably enumerable case. As is true for every class that contains the finite sets and has a Σ_3^0 index set, there is a uniformly computably enumerable listing (A_e) of the computably enumerable sets in $\mathcal{K}$. Here we show there is a listing that includes the witnesses of the Σ_3^0 statement, namely the constants via which the A_e are K-trivial. This is true even in the Δ_2^0 case.

We say that α is K-trivial via the constant c if $K(\alpha \upharpoonright n) \leqslant K(n)+c$ for all n. A Δ_2^0-approximation is a computable $\{0,1\}$-valued function $\lambda x, s\ B_s(x)$ such that $B_s(x) = 0$ for $x \geqslant s$ and $B(x) = \lim_s B_s(x)$ exists for each x.

Theorem 15.2.1 (Downey, Hirschfeldt, Nies and Stephan [81]). *There is an effective list $((B_{e,s}(x))_{s\in\mathbb{N}}, d_e)$ of Δ_2^0-approximations and constants such that each K-trivial real occurs as a real $B_e = \lim_s B_{e,s}$, and each B_e is K-trivial via the constant d_e. Moreover, $B_{e,s}(x)$ changes at most $O(x^2)$ times as s increases, with effectively given constant.*

Proof. We define, uniformly in e, Δ_2^0-approximations $B_{e,s}$ and Kraft-Chaitin sequences V_e such that, for effectively given constants g_e and for each stage u,

$$\forall w \leqslant u\, \exists r \leqslant K_s(n) + g_e + 3\, (\langle r, B_{e,s} \upharpoonright n\rangle \in V_{e,s}). \tag{15.4}$$

Then we obtain d_e by adding to $g_e + 2$ the coding constant of a prefix-free machine uniformly obtained from V_e.

We need a lemma whose proof will be obtained by analyzing the proof of DENIS Theorem 5.8 in [226]. For those familiar with that paper, we include a proof of this lemma below. The lemma says that there is a uniformly computable set Q_e of "good stages" such that B_e changes only at a good stage, and the cost of these changes, namely the weight of short descriptions of the new initial segments $B_{e,s} \upharpoonright m$, is bounded by an effectively given constant 2^{g_e}.

Lemma 15.2.2. *There is an effective list $((B_{e,s}(x))_{s\in\mathbb{N}}, g_e, Q_e)$ of Δ_2^0-approximations, constants, and (indices for) computable sets of stages, with the following properties.*

1. *$B_{e,u}(x) \neq B_{e,u-1}(x) \Rightarrow u \in Q_e$.*
2. *Let $Q_e = \{q_e(0) < q_e(1) < \cdots\}$ (Q_e may be finite). If $q_e(r+1)$ is defined, then let $\widehat{c}(z,r) = \sum_{z<y\leqslant q_e(r)} 2^{-K_{q_e(r+1)}(y)}$ and let*

$$\widehat{S}_e = \sum \{\widehat{c}(x,r) : u = q_e(r+2) \text{ defined} \wedge \\ x \text{ is minimal such that } B_{e,u}(x) \neq B_{e,u-1}(x)\}.$$

Then $\widehat{S}_e < 2^{g_e}$.

Moreover, $B_{e,s}(x)$ changes at most $O(x^2)$ times as s increases, with effectively given constant.

We first complete the proof of the theorem assuming the lemma. We obtain V_e by emulating the construction of an K-trivial real in Theorem 15.1.4 (see also [226], Proposition 3.3). At stage u, for each $w \leqslant u$, put $\langle K_u(w) + g_e + 3, B_u \upharpoonright w\rangle$ into V_u in case

(a) $u = w$, or

(b) $u > w \ \wedge \ K_u(w) < K_{u-1}(w)$, or

(c) $B_{u-1} \upharpoonright w \neq B_u \upharpoonright w$.

Clearly, each V_e satisfies (15.4). It remains to show that V_e is a Kraft-Chaitin sequence. We drop the subscript e in what follows. The weight contributed by axioms added for reasons (a) and (b) is at most $2^{-g-2} \leqslant 1/4$. Now consider the axioms added for reason (c). Since B only changes at stages in Q, for each w there are at most two enumerations at a stage $u = q(r+2)$ such that $w > q(r)$. The weight contributed by all w at such stages is at most $\Omega/4$. Now assume $w \leqslant q(r)$, and let $u = q(r+2)$.

Case 1. $K_{q(r+1)}(w) > K_u(w)$. This happens at most once for each value $K_u(w)$, $u \in Q$. Since each value corresponds to a new description of w, the overall contribution is at most $\Omega/8$.

Case 2. $K_{q(r+1)}(w) = K_u(w)$. Since $B(x)$ changes for some minimal $x < w$ at u, the term $2^{-K_u(w)}$ occurs in the sum $\widehat{c}(x,r)$. Since $\widehat{S} \leqslant 2^g$, the overall contribution is at most $1/8$. $\square$

Proof of Lemma 15.2.2. By [226], Theorem 6.2, let $(\Gamma_m)_{m\in\mathbb{N}}$ be a list of (total) tt-reductions such that the class of K-trivial reals equals $\{\Gamma_m(\emptyset') : m \in \mathbb{N}\}$. Let $A_m = \Gamma_m(\emptyset')$, with the Δ^0_2-approximation $A_{m,s} = \Gamma_m(\emptyset'_s)$. DENIS NEED TO FIX We refer to the proof of [226], Theorem 5.8, and adopt its notation.

Let e be a computable code for a tuple consisting of the following: m, a constant b (we hope A_m is K-trivial via b), numbers i (a level in the tree of runs of procedures) and n (we consider the n-th run of a procedure $P_i(p,\alpha)$, hoping it will be golden), and a constant g_e which we hope will be such that $2^{g_e} = p/\alpha$. (We assume that g_e is at least the constant via which the empty set is K-trivial.) Given e, we define a set Q_e. If e meets our expectations then Q_e will be equal to A_m and will be K-trivial via g_e. Otherwise, Q_e will be finite, but g_e will still be a correct constant via which Q_e is K-trivial.

As in the main construction, we obtain a coding constant d for a prefix-free machine by applying the Recursion Theorem with parameters to m, b, let $k = 2^{b+d}$, and only consider those $i \leqslant k$.

Given e, we run the construction as in [226], Theorem 5.8, in order to define Q_e. For each u, we can effectively determine if u is a *stage* in the sense of that construction. Moreover we can determine if by stage u we started the n-th run $P_i(p, \alpha)$ of a procedure P_i. We leave Q_e empty unless $g_e = p/\alpha$. In that case we check if $u = q(r)$ in the sense of [226], Theorem 5.8. If so we declare $u \in Q_e$.

Finally we let $B_{e,u}(x) = A_{m,\max(Q_e \cap \{0,\ldots,u\})}$. Thus if Q_e is finite we are stuck with $A_{m,\max Q_e}$. The property $\widehat{S} \leqslant 2^{g_e}$ is verified in the proof of [226], Theorem 5.8. The $O(x^2)$ bound on the number of changes follows as in [226], Fact 3.6. □

Note that we can replace the list (Γ_m) in the above proof by a listing of a subclass of the K-trivials containing the finite sets. Thus there are also effective listings with constants for the K-trivial computably enumerable sets and for the K-trivial computably enumerable reals.

Let C be a set of computably enumerable indices closed under equality of computably enumerable sets. We say that C is *uniformly* Σ^0_3 if there is a Π^0_2 relation P such that $e \in C \leftrightarrow \exists n\, (P(e, n))$ and there is an effective sequence (e_n, b_n) such that $P(e_n, b_n)$ and $\forall e \in C\, \exists n\, (W_e = W_{e_n})$. We have proved that $\mathcal{K}$ is uniformly Σ^0_3. It would be interesting to see which other properly Σ^0_3 index sets have that property, for instance the class of computable sets.

Recall that A is strongly K-trivial via a constant c if $\forall \sigma\, (K(\sigma) \leqslant K^A(\sigma) + c)$, where K^A is K-complexity relativized to A. In [226] it is proved that each K-trivial real is strongly K-trivial. However, in the proof the constant of strong K-triviality is not obtained in a uniform way. The following corollary shows that this non-uniformity is necessary.

Corollary 15.2.3 (Downey, Hirschfeldt, Nies and Stephan [81])**.** *There is no effective way to obtain from a pair (A, b), where A is a computably enumerable set that is K-trivial via b, a constant c such that A is strongly K-trivial via c.*

Proof. Otherwise, by Theorem 15.2.1 above we would obtain a listing (A_e, c_e) of all the strongly K-trivials with appropriate DENIS constants. Nies [**?**], Theorem 5.9, showed that such a listing does not exist. □

15.3 Martin-Löf cupping, and other variations

15.4 Nies Low_2 Top Theorem

Theorem 15.4.1 (Nies unpubl.). *Suppose that I is a nontrivial Σ_3^0 ideal in the c.e. Turing degrees. Then there is a low_2 c.e. set A such that*

$$W_e \in I \textit{ implies } W_e \leqslant_T A$$

Corollary 15.4.2. *There is a low_2 c.e. set A such that for all K-trivial sets W, $W \leqslant_T A$.*

Proof. Our method of proof is a low_2 construction with a coding one akin to the Downey-Terwijn proof that each Σ_3^0 ideal in the c.e. wtt-degrees can be realized as the degrees of presentations of a left-c.e. real, Theorem 8.4.9. Thus we will recall from there the following Lemma of Yates.

Lemma 15.4.3 (Yates [324]). *Let $I \in \Sigma_3^0$ and let $\mathcal{C} = \{W_i : i \in I\}$ be a collection of c.e. sets containing all the finite sets. Then there is a uniformly c.e. collection $\{V_e : e \in \mathbb{N}\}$ such that $\mathcal{C} = \{V_e : e \in \mathbb{N}\}$.*

By this lemma, we can suppose that that the Σ_3^0-ideal is given to us by a uniform collection of c.e. sets $U_0, U_1, U_2, \ldots$. Indeed, by replacing U_{e+1} by $\oplus_{j \leqslant e+1} U_j$, we can even consider that $U_i \leqslant U_{i+1}$ uniformly.

This will then give rise to the following coding requirements.

$$C_e : U_e \leqslant_T A.$$

To meet C_e we will build a reduction $\Gamma_e^A = U_e$. (Of course, this is not quite correct since will will build a number of reductions, the one which works will be generated by the true path of the priority tree.)

The fundamental idea is that we will define $\Gamma^A = U_e$ in stages, and, as usual, if $x \notin U_{e,s}$ and x enters $U_e - U_{e,s}$ then we will need to change $A_s \upharpoonright \gamma(x, e, s)$.The first approximation to this will be to simply enumerate $\gamma(x, e, s)$ into A_{s+1}. The first approximation is that $U_e \leqslant_m A$, since then $x \in U_e$ iff $\gamma(x, e, s) \in A$, where $\gamma(x, e, s) = \gamma(x)$ is the first use appointed for $\Gamma_i^A(x)$.

However, as we will see, there are other possible reasons for enumerating $\gamma(x, e, s)$ into $A_t - A_s$. We will use the usual conventions. If we move $\gamma(x, e, s)$ then we will also monotonically move $\gamma(x', e, s)$ for all $x' > x$ currently defined. Thus if for some reason $\gamma(x, e, s) \to \infty$ then this is also true *a fortiori* for all $\gamma(x', e, s)$ and $x' > x$. This method resembles Sacks' coding in the Density Theorem.

Nevertheless, if we can show that for a fixed Γ we have $\lim_s \gamma(x, e, s)$ exists, and whenever x enters U_e we will change $A_s \upharpoonright \gamma(x, e, s)$ then we will have $U_e \leqslant_T A$.

Turning to the low_2 requirements we have the following.

$$N_i : \text{limsup}_{s \to \infty} \ell(\tau, s) \to \infty \to \Phi_i^A \text{ is total]).}$$

Here $\ell(\tau, s)$ denotes the length of convergence

$$\max\{x : \forall y \leqslant x(\Phi_i^A(y) \downarrow [s])\}$$

measured at the node τ on the true path devoted to N_i. Note that this will make A low$_2$ since, as usual for an infinite injury argument, the true path, TP, is recursive in $0''$ and hence we can answer the question "Is Φ_i^A total?" recursively in $0''$.

As a first approximation, the priority tree will have 3 types of nodes :

- β nodes for the sake of C_e with a single outcome ∞.
- τ nodes for the sake of N_i with outcomes $\infty <_L f$. (Later this will be modified, as we see, when we incorporate "τ-correctness")
- α nodes living below $\tau\widehat{\ }\infty$ also devoted to N_i via subrequirements $N_{i,j}$. Such an α will be trying to preserve a computation of the form $\Phi_i^A(j)$. These nodes have outcomes $\cdots <_L 3 <_L 2 <_L 1 <_L 0$. The unique τ node associated with α will be denoted by $\tau(\alpha)$. We will refer to α as a worker node and τ as α's *mother*. For the i associated with τ, on the true path, an outcome n will demonstrate that $\Phi_i^A(j) \downarrow$ with use n.

The action of a β node is as described above. When we visit β, if the ∞ outcome looks correct, we will define more of $\Gamma_\beta^A = U_{e(\beta)}$ by defining more $\gamma(x, \beta, s)$ and enumerating these, if defined, into $A - A_s$ should x have entered $U_{\beta,s}$ since the last β-stage. approximation to Tot via Γ. In the construction to follow, $\gamma(x, \beta, s)$ can also be changed for the sake of the $N_{j,k}$ of *lower* priority (which are defined precisely below). However, this action will be controlled by $\tau(\alpha)$of *higher* priority than β, and we will certainly ensure that $\lim_s \gamma(x, \beta, s)$ exists, *should β be on the true path.*

Below the infinite outcome of a τ node, that is where the length of convergence looks infinite infinitely often, there will be a tree of α (worker) nodes each devoted to some k, that is some subrequirement $N_{i,k}$ of N_i. These nodes will be devoted to requirements of the form

$$\Phi_i^A(k) \downarrow .$$

These will be able to tolerate some kind of finite noise from above, but essentially act like lowness requirements.

The basic idea at τ is that as the length of convergence rises, and hence $\tau\widehat{\ }\infty$ looks correct, we want to preserve more and more of A to try to force Φ_i^A to be total if infinitely often it looks total. As suggested above, this preservation is implemented by the tree of worker nodes below the mother τ.

Now there are two quite different things that τ must deal with. One thing it must deal with if the fact that above τ, there are coding nodes β whose action is more or less out of control in so far as β is concerned. Thus, in some sense, we might see some apparently τ-expansionary stage, s_0 and hence τ might wish to preserve $\Phi_i^A(k)\downarrow$ for $k \leqslant k'$ to try to force Φ^A to be total. Now, as we will see, τ has a way of dealing with β-nodes of *lower* priority (i.e. below $\tau\widehat{\ }\infty$), *but* if we have $\beta\widehat{\ }\infty \preccurlyeq \tau$, then surely τ cannot stop β from encoding U_β into A.

The reader should think of this as being like the Thickness Lemma. Here, there are encodings above that a negative requirement cannot control. *But* there is a notion of a node τ's computations being "correct". Correctness entails the notion that from above no injury can occur to a given computation.

Andre Nies came up with a nice idea here which allows an analogue of this notion of correctness. The injury from above will turn out to be *uniformly low*$_2$, and hence this notion of correctness can be incorporated into the construction at τ.

We need the following important lemma.

Lemma 15.4.4 (Nies, unpubl.)**.** *Uniformly in the index of a computably enumerable set U, we can determine a uniformly computably enumerable sequence $\{V_e : e \in \mathbb{N}\}$, such that*

$$e \in \mathit{Tot}^A \to V_e \leqslant_T U,$$

$$e \notin \mathit{Tot}^A \to V_e =^* K,$$

where K denotes the halting problem.

Proof. We will consider Φ_e to be replaced by Ψ_e such that $\Phi_e^Z(k)\downarrow[s]$ iff for all $k' \leqslant k$, $\Phi_e^Z(k')\downarrow$, and then we will also use the hat convention. Now at stage s, if $n \in K_s$ and $\max\{k : \widehat{\Phi}_e^U(k)\downarrow[s]\} < n$, enumerate n into V_e. Then if Φ_e^U is not total, then by the hat convention, we have that for infinitely many s, $\max\{k : \widehat{\Phi}_e^U(k)\downarrow[s]\}$ drops down to a fixed liminf. Thus $K =^* V_e$. Otherwise, $\max\{k : \widehat{\Phi}_e^U(k)\downarrow[s]\} \to \infty$. Then $V_e \leqslant_T U$ since to decide if $n \in V_e$ compute a stage where $\max\{k : \widehat{\Phi}_e^U(k)\downarrow[s]\} > n$ via U-correct computations. Then $n \in V_e$ iff $n \in V_{e,s}$.

To complete the proof, not that the sequence V_e was obtained uniformly in $U = \cup_s U_s$. $\square$

Now we apply this Lemma to the sequence $\{U_j : j \in \mathbb{N}\}$ representing our Σ_3^0 ideal I. We have a computable function f such that for each computably enumerable set W_k, we have $V_e = W_{f(e,k)}$. Then for each $U_i = W_{g(i)} \in I$, we have $e \in \mathrm{Tot}^{U_i}$ iff $W_{f(e,g(i))} \leqslant_T U_i$ iff $W_{f(e,g(i))} \in I$. That is we have the following:

Corollary 15.4.5 (Nies, unpubl.)**.** *The sequence $\{U_e : e \in \mathbb{N}\}$ representing I is uniformly low*$_2$*.*

The broad idea now is that above τ will be a finite number of coding nodes β_i, say. These are coding sets U_i for a finite number of i, and we have thus a uniformly low$_2$ amount of "noise" in the τ-computations. Before we say exactly how we will deal with this, we will discuss the impact of coding nodes of lower priority than τ.

A typical situation is the following. We have a node τ devoted to N_i. In a manner to be later described, it will be able to guess at the behavior of higher priority C_f nodes, and will only use correspondingly τ-correct computations. However, if $\limsup_{s\to\infty}\ell(\tau,s)\to\infty$ we need to ensure that Φ^A is total. Thus we will need to deal with various β-nodes between $\tau\widehat{\ }\infty$ and α as well as β nodes below α. (Recall α is some τ-worker node trying to preserve $\Phi_i^A(k)\downarrow$.) The point is that β such nodes may be trying to put infinitely many elements into A whereas α is trying to preserve computations. So suppose that we have

$$\tau\widehat{\ }\infty \subset \beta_1\widehat{\ }\infty \subset \beta_2\widehat{\ }f \subset \beta_3\widehat{\ }\infty \subset \alpha.$$

The way that α deals with these β-nodes between it and τ is the following. We reach τ (i. e. s is a τ stage) and it is expansionary with $\ell(\tau,s)>k$. *We also assume that $\ell(\tau,s)>m(\alpha,s)+1$ via τ-correct computations, yet to be defined.* What α would now like to do is to preserve its computation from A,

$$\Phi_i^A(k)\downarrow[s].$$

But it cannot really stop the β_i from putting their numbers (which may well be below the use $\varphi_i(k)[s]$) into A. What α tries to do is to lift the relevant β_i-uses above $\varphi_i(k)[s]$ for almost all such uses. If α succeeds, then the computation $\Phi_i^A(k)\downarrow[s]$ becomes not only τ-correct, but α-correct, and hence α can preserve the computation forever. On the other hand, if α fails, we must arrange things so that we do *not* play the ∞ outcome of τ, and that in fact we do not allow $\limsup_{s\to\infty}\ell(\tau,s)\to\infty$. The trick is to capriciously kill the τ computations we are trying to preserve before we actually believe them.

In more detail, as above, suppose we we reach τ and for any of the β_i with $\tau\widehat{\ }\infty \subseteq \beta_i\widehat{\ }\infty \subseteq \alpha$ (in our case $i=1,3$) we have that $\gamma^A(e(\beta_i),x(\beta_i)) < \varphi_i^A(k)[s]$ (for $x>k$), for some least $x(\beta_i)$. What α (or at least τ blaming α) does is

- request that a number $z\leqslant\gamma(e(\beta_i),x(\beta_i))[s]$ enter $A[s]$, and
- lift $\gamma(e(\beta_i),x')$ above s for all x' greater than $x(\beta_i)$.

The entry of this number z entering $A_{s+1}-A_s$ is that it will kill the τ computation α desired to preserve, and hence it is now reasonable that we do not believe the τ-stage is expansionary after all. The relevance of this comes at the next potential $\tau\widehat{\ }\infty$ stage s'. The number z has entered A below the use of the

$$\Phi_i^A(k)\downarrow[s].$$

computation, and we have reset the $\gamma(e(\beta_i),x(\beta_i))[s]$. At this very stage s', we would see if the conditions $\gamma^A(e(\beta_i),x(\beta_i)) > \varphi_i^A(k)[s']$ we desire are yet satisfied of if there remain in the same bad state as they were at stage s. If they are still offensive, (in particular, because $\varphi\varphi_i^A(k)[s']$ has also increased since, after all, we changed $A_{s'} - A_s$ on $\varphi_i^A(k)[s]$), then we would repeat the actions again, enumerating the new $\gamma(e(\beta_i), x(\beta_i))[s']$ into $A_{s'+1}$ capriciously killing the τ computations we'd like to preserve, but again playing the finite outcome of τ.

Now, if this cycle repeats itself infinitely often, then there are only finitely many $\tau^\frown\infty$ stages and, indeed, Φ_i^A is *not total*.

Now suppose that we hit τ at some τ-correct stage t and we have that all of the offending γ-uses above α have cleared the $\Phi_i^A(k)\downarrow[t]$. use. At such a stage, we would allow α to impose restraint, initializing lower priority requirements.

Modulo the notion of a τ-correct computation, notice that α can now only be injured by $\gamma^A(e(\beta_i), x(\beta_i))$ associated with the offending β_i between τ and α for x below $k = k(\alpha)$. This is finite injury. Also modulo the notion of a τ-correct computation, this all means that *if* $\tau^\frown\infty$ is on the true path, then Φ_i^A will indeed be total. Once α asserts control after some finite injury noise, at some τ-correct stage, α's restraint will be successful and preserve the $\Phi_i^A(k)\downarrow[t]$ computation forever.

τ-correctness Now we turn to the mysterious notion of τ-correctness. We have seen that various β-nodes *above* τ can *also* affect things, but it is not within τ's power to clear γ-markers associated with such β of higher priority. So what is τ to do?

Now we return to the fact that the U_e are uniformly low_2. Thus we are processing numbers as above, and we reach a stage where it appears that we have a τ-expansionary stage t because we have aligned the γ markers. It could be, however, that for some such k, the $\Phi_i^A(k)\downarrow[t]$-computation might actually be A-incorrect because some p enters $U_\beta - U_{\beta,s}$ and we would need to enumerate $\gamma(\beta,p)[t]$ into A and maybe $\gamma(\beta,p)[t] < \varphi_i^A(k)[t]$. Thus, we should not have played $\tau^\frown\infty$ after all.

To avoid this we will build our own new procedure $\Xi^{U_\beta} = \Xi_\tau$ at τ. Of course we will know its index, and hence its low_2-ness index by the Recursion Theorem. When we are ready to believe that the stage is τ-expansionary, we will enumerate (essentially) $\Xi_\tau^{U_\beta}(n)\downarrow[t]$ with some (many) *new* n and having use $\varphi'(k)[t]$, where $\varphi'(k)[t]$ denotes the maximum of the p with $\gamma(\beta,p)[t] < \varphi(k)[t]$. (These are the only β-markers which could kill the A-computation at τ we would like to preserve. Then since U_β is uniformly low_2, we would be able to test this computation, akin to the Robinson technique, to see if Ξ^{U_β} is total. The idea here is that for any small j with $\Xi_\tau^{U_\beta}(j)\uparrow[t]$ we would also re-define them with the new use $\varphi'(k)[t]$.)

Now we mark time in the construction, continuing to define $\Xi_\tau^{U_\beta}(n')\downarrow[t]$ this for $n' > n$, (for the same k-use) (of course really at substages of stage t) not leaving τ until the situation resolves itself.

As with the Robinson trick, resolution will occur when

- *Either* we are shown that one of the $\Xi^{U_\beta}(m)$ computations for some m are U_β-incorrect. This will be some m corresponding to $k' \leqslant k$.
- *Or* we get a $\mathbf{0}''$-certification that we should believe the outcome ∞.

The point here is that U_β and hence the the coded version of U_β in A is uniformly low$_2$, and hence $\mathbf{0}''$ can figure out if some procedure with U_β as an oracle is total or not. We have begun a process that will make it total if the original computations are U_β-correct, and hence *either* the original ones were *not* correct, or we would get a $\mathbf{0}''$-certification uniformly that the procedure is total, which is a weak endorsement for the computation.

In the first case, we would play the f outcome since the computation is wrong anyway. In the second, we would we would like to play the ∞ outcome.

However, this needs to be implemented on the tree in a Δ_3^0 refinement of τ. After all, all we have is a uniform Δ_3^0 way of deciding this. Being Δ_3^0 means that we will need to replace the two outcomes of τ by an infinite number of outcomes of τ:

$$(1,\infty) <_L (1,f) <_L (2,\infty) <_L \dots .$$

These outcomes represent the Σ_3^0 outcomes for totality, the ones of the form (i,∞) and the Σ_3^0 outcomes for non-totality, (i,f). As usual, we can assume that exactly one of these is correct infinitely often.

We turn now to the formal details.

15.4.1 The Priority Tree

Define the priority tree as follows. If ν is on the priority tree and $|\nu| = 3e$, ν is devoted to C_e. Put $\nu\widehat{\ }\infty$ on the priority tree. ν is a β-node and $e(\nu) = e$.

Otherwise, we use two lists L_1 and L_2 to assign requirements to nodes. As usual the lists $L_1(\lambda) = L_2(\lambda) = \omega$. We use the convention that we do not change lists as we pass to the outcomes of a node unless specifically so instructed.

If $|\nu| \equiv 1 \bmod 3$ assign N_i to ν where $\langle e,i\rangle$ is the least member of $L_1(\nu)$. Put $\nu\widehat{\ }(j,\infty)$ and $\nu\widehat{\ }(j,f)$ (for $j \in \mathbb{N}$) on the priority tree as an $2\omega^*$ sequence, with $\nu\widehat{\ }(1,\infty) <_L \nu\widehat{\ }(1,f) <_L \nu\widehat{\ }(2,\infty) <_L \dots$. Let $L_1(\nu\widehat{\ }(j,\infty) = L_1(\nu\widehat{\ }(j,f)) = L_1(\nu) - \{i\}$. Let $L_2(\nu\widehat{\ }(j,f)) = L_2 - \{\langle i,k\rangle : k \in \omega\}$. ν is a τ node, and $i(\nu) = i$.

Finally, if $|\nu| \equiv 2 \bmod 3$, find the least $\langle i,k\rangle$ in $L_2(\nu)$ such that $i \notin L_1(\nu)$. Assign $N_{i,k}$ to α. Put $\nu\widehat{\ }s$ and on the priority tree for all $s \in \mathbb{N}$ with $\nu\widehat{\ }\langle s+1\rangle <_L \nu\widehat{\ }s$. Let $L_2(\nu\widehat{\ }s) = L_2(\nu) - \{\langle i,k\rangle\}$. ν is an α-node, $i(\nu) = i$ and $k(\nu) = k$.

15.4.2 The Construction

Step 1.

At each stage of the construction, we put at most one number into A. We determine this number by approximating TP by TP_s as follows. We begin at λ and say that s is a λ-stage. Suppose that s is a ν-stage. And suppose that t is the maximum of 0 and the last ν-stage. There are 3 cases.

Case 1. ν is a β-node.

For all $n \leqslant s$, if not currently defined, and $n \notin U_{\beta,s}$, define a marker $\gamma(n, \beta, s)$. Choose these large and fresh. If $\gamma(n, \beta, s)$ is defined, $\gamma(n, \beta, s) \notin A_s$ and $n \in U_{\beta,s}$ enumerate $\gamma(n, \beta, s)$ into A_{s+1}. Initialize any α-nodes extending β with α's mother above β, *and* $k(\alpha) \geqslant n$.

Case 2. ν is a τ-node.

Let $i = i(\nu)$. Determine whether it is reasonable to believe that the stage is ν-expansionary and why.

First if the Δ^0_3 approximation to $\Xi_\tau^{U_\beta}$ has played outcome (j, f) for some j with $\nu\widehat{\ }(j, f)$ *left* of the outcome of ν played at stage t, then declare that s is a $\nu\widehat{\ }(j, f)$ stage, and define $m\ell(\nu, s) = m\ell(\nu, t)$.

Otherwise, we compute a first approximation to $\ell(\nu, s)$. Let $\ell(\nu, s) = \max\{x : \forall y \leqslant x \Phi_i^A(y) \downarrow [s])\}$.

Now for any β extending $\nu\widehat{\ }(j, \infty)$ and any $j \in \mathbb{N}$, if $\gamma(n, \beta, s) \leqslant \varphi^A(n)[s]$ enumerate $\gamma(n, \beta, s)$ into A_s (making $\ell(\nu, s) < n$, now). We cancel $\gamma(n', \beta, s)$ for $n' \geqslant n$. (*We remark that at this stage we are* not *re-defining* $\gamma(n, \beta, s+1)$. *This action will be taken the next time we visit* β.)

We can then generate a ν-length of agreement function $\ell(\nu, s)$ replacing the above, but now *correct from below.* If $\ell(\nu, s) \leqslant m\ell(\nu, t)$, then compute the Δ^0_3 approximation to

$$\Xi_\tau^{U_\beta},$$

is total or not. Now we will definitely not believe it is total. It must therefore output some (j, f) with $j < s$. We choose the least such $j \leqslant s$ if the Δ^0_3 approximation has assumed (j, f) at some stage s' with $t \leqslant s' \leqslant s$. We will declare that s is a $\nu\widehat{\ }(j, f)$-stage. Otherwise, if the Δ^0_3 approximation is saying $\Xi_\tau^{U_\beta}$ is total, define $m\ell(\nu, s) = m\ell(\nu, t)$.

If $\ell(\nu, s) \geqslant m\ell(\nu, t)$, then we see if this is confirmed by running the Δ^0_3 approximation to $\Xi_\tau^{U_\beta}$. That is, we wait running the enumeration of U_β for $\beta\widehat{\ }\infty \preccurlyeq \nu$, defining more and more of $\Xi_\tau^{U_\beta}(n)[v] \downarrow$ for more and more n giving all of them $\max\{p : \gamma(p, \beta, s) \leqslant \varphi^A(t+1)[s] \wedge \beta$ longest with $\beta\widehat{\ }\infty \preccurlyeq \nu\} =_{def} \xi(n)[s]$ as their use. We do this until the situation resolves itself. Either we see that the A-computation is injured because U_β enumerates one of it markers into A, changing $U_\beta \upharpoonright \xi(t+1)$, or the Δ^0_3 procedure outputs (j, ∞) for some (least) j, at some stage s'' for $t' < s'' \leqslant s$, where t' is the the minimum of t and the last $\nu\widehat{\ }(k, \infty)$ stage.

In the latter case, we declare that s is a $\nu\widehat{\ }(j,\infty)$-stage, and define $m\ell(\nu,s) = m\ell(\nu,t)+1$, and we declare this to be the ν-correct length of agreement *from above and below.*

Case 3. ν is an α-node.

Let $i = i(\alpha), k = k(\alpha)$, and $\tau = \tau(\alpha)$. If $\ell(\tau,s) \leqslant m(\nu,s)$, set $TP_s = \nu\widehat{\ }\infty$. If this is the first α stage either that has occurred, or has occurred since α was last initialized, then initialize all β requirements with $\alpha\widehat{\ }\infty \preccurlyeq \beta$, and initialize all requirements with mothers $\widehat{\tau}$ and $\alpha\widehat{\ }\infty \preccurlyeq \widehat{\tau}$.

Step 2.

Determined TP_s, the true path at stage s, which is the unique string ν of length s such that s is a ν stage. Initialize all γ with $\gamma \not\leqslant_L TP_s$.

End of Construction.

15.4.3 The Verification

We verify the following by simultaneous induction on $\nu \subset TP$:

(i) If ν is a β-node, then, for $e = e(\nu)$, $\lim_x \Gamma^A(e,x) = \Gamma^A(e)$ exists for all x, and moreover $\Gamma^A_\nu = U_\nu$. In particular, $\lim_s \gamma(x,\nu,s)$ exists.

(ii) If ν is a τ-node, then $\nu\widehat{\ }(j,\infty) \subset TP$ iff there are infinitely many ν-expansionary stages which are ν correct from above and from below.

(iii) ν is a τ-node, and $\nu\widehat{\ }(j,\infty) \subseteq TP$ iff for all k and almost all $\nu\widehat{\ }(j,\infty)$-stages s, $\Phi^A_{i(\nu)}(k)\downarrow$, and for almost all $\nu\widehat{\ }(j,\infty)$-stages s, ν takes no action for k.

(iv) If ν is a τ-node, and $\nu\widehat{\ }(j,f) \prec TP$, then ν has only finite effect on the nodes extending $\nu\widehat{\ }(j,f)$.

(v) If ν is an α node then ν initializes $\widehat{\nu}$ extending ν only finitely often, and ν is initialized only finitely often.

We assume (i)-(v) for all $\sigma \subset \nu$. Let s_0 be a stage at which the hypotheses apply to all such σ and we are never again to the left of ν. There are 3 cases to consider.

Case 1. ν is a β-node. Then after stage s_0 ν is never again initialized. Thus axioms enumerated for ν after stage s_0 cannot be changed except through the agency of enumeration into $A - A_{s_0}$. Moreover, at every β-stage we issue new axioms, and hence the only relevant number (i) will hold for ν, provided that we can argue that $\lim_s \gamma(x,\nu,s)$ exists, for all x. The only way that $\gamma(x,\nu,s+1) \neq \gamma(x,\nu,s)$ after stage s_0 *without* x entering U_ν, is that some $\tau\widehat{\ }(j,\infty) \preccurlyeq \nu$ acts to clear $\gamma(x,\nu,s)$ from some use. Since $\tau\widehat{\ }(j,\infty) \preccurlyeq \nu$, and $\nu \prec TP$, we know that $\tau\widehat{\ }(j,\infty) \preccurlyeq TP$. Hence, there are only finitely many τ stages where action is taken for numbers $\leqslant x$ by τ. Thereafter, τ has *no* effect upon ν at argument x, and hence $\lim_s \gamma(x,\nu,s)$, since this is true for all such τ-nodes above ν. Of course Nodes $\tau\widehat{\ }(j,f) \preccurlyeq \nu$ will only have finite effect upon ν. This concludes the argument for this case.

Case 2. ν is an τ node. This is the case the whole deal was set up for. If $\nu\widehat{\ }(j,f) \preccurlyeq TP$, Then, we are only finitely often left of $\nu\widehat{\ }(j,f)$, and no action at ν has any effect on nodes extending $\nu\widehat{\ }(j, f)$ unless we play some outcome left of (j, f). This gives (iv).

Thus we need to consider the case that $\nu\widehat{\ }(j, \infty) \prec TP$. Without loss of generality we will suppose that s_0 has the additional property that we are never again left of $\nu\widehat{\ }(j, \infty)$ after stage s_0. By Corollary 15.4.5, we know that Ξ^{U_β} is total, where U_β is the longest β node above ν. (Without loss of generality, there is some such, for otherwise there is no injury from above.) We claim that this implies that Φ_ν^A is total too, via (ii), (iii).

Now we argue that (ii) and (iii) hold for all k, and do so by induction. Suppose that argument holds for all $k' < k$, and additionally s_1 is a $\nu\widehat{\ }(j, \infty)$ stage where all $\gamma(n, \widehat{\beta}, s_1)$ exceed $\varphi^A(k-1)[s_1]$, for $n \geqslant k-1$, $\nu\widehat{\ }(j\infty) \leqslant_L \widehat{\beta}$, and the stage is correct from below and above, and moreover, the computations are U_β correct. Now any new axioms appointed to nodes $\widehat{\beta}$ as above after stage s_1 will be bigger than s_1 and hence cannot affect these computations.

Now we argue for k. We know that $\nu\widehat{\ }(j\infty) \prec TP$ and hence there are infinitely many $\nu\widehat{\ }(j\infty)$ stages. Consider the next ν-stage, say s_2. At such a stage, we will enumerate any $\gamma(k, \widehat{\beta}, s_1)$ less than $\varphi^A[k][s_2]$, and will do so at each ν-stage until we see one where it appears to be correct from below. This is the only time we can play a $\nu\widehat{\ }(j, \infty)$-stage, and we will only the do when we are told that the computations are U_β-correct via the outcome (j, ∞) being played again. Since the outcome (j, ∞) is the correct one, this means that infinitely often the computations are really U_β correct, if s_3 is then next $\nu\widehat{\ }(j, \infty)$ stage where the k-computations are truly U_β-correct, then $\varphi^A[k][s_2]$ is truly correct, giving the result.

Case 3. ν is a α-node. Straightforward. □

We have seen that there is no low c.e. degree above all the K-trivials, and there is a low$_2$ c.e. degree. What happens if c.e. is removed? The following is a corollary to a more general theorem of Kučera and Slaman about ideals in the Δ_2^0 degrees.

Theorem 15.4.6 (Kučera and Slaman [162]). *Suppose that I is any nontrivial Σ_3^0 ideal ideal in the c.e. degrees, and in particular the K-trivials. Then there is a low Δ_2^0 degree $\mathbf{a}$ such that $\mathbf{b} < \mathbf{a}$ for all $\mathbf{b} \in I$.*

The method of proof is a finite injury forcing one akin to the low basis theorem with coding steps interleaved. We refer the reader to [162] for details. The remaining question here is whether such a $\mathbf{a}$ can be *random*.

15.5 Strong jump traceability

15.5.1 Basics

In this section we study the notion of strong jump traceability introduced by Nies [225], and studied by Figueira, Nies and Stephan [**?**] and later Cholak, Downey and Greenberg [**?**]. indexGreenberg, N. They form a fascinating class which is a proper subclass of the K-trivials defined by a cost function construction as we will see. We have seen that all K-trivials are low. That is we have seen that if A is K-trivial then A is jump traceable. In the computably enumerable case this is the same as being *superlow*: if B is computably enumerable then B is superlow iff there is an order g with B', g-traceable.

Now it turns out that K-trivials are, in fact very low indeed, as we now see. We remind the reader that J^A denotes the actual value (if any) of the jump of A on argument n.

Definition 15.5.1 (Figueira, Nies and Stephan [**?**])**.** We say that A is *strongly jump traceable* iff for all orders h there is a trace $W_{g(n)}$ with $J^A(n) \in W_{g(n)}$ for all n.

It is not altogether clear that such reals should exist.

Theorem 15.5.2 (Figueira, Nies and Stephan [**?**])**.** *There exist c.e. sets A which are promptly simple and strongly jump traceable.*

Proof. (sketch) The following sketch is from [**?**]. To begin the proof, suppose that h is a slow-growing order, and that we wish to construct a non-computable c.e. set A which is jump-traceable with respect to h. Let P_e be the e^{th} non-computability requirement (which says that $A \neq \bar{W}_e$) and let N_e be the requirement which is responsible for enumerating that part T_e of the trace we build which is suppose to trace $J^A(e)$.

We order the requirements thus:

$$\underbrace{N_0\, N_1\, N_2\, \cdots\, N_e\, \cdots}_{h(e)=1}\; P_0\; \underbrace{\cdots\, N_e\, \cdots}_{h(e)=2}\; P_1\; \underbrace{\cdots\, N_e\, \cdots}_{h(e)=3}\; P_2\; \cdots \tag{15.5}$$

The construction is now straightforward: each P_e is appointed a follower x. If at stage s, P_e is not yet satisfied, and x appears in W_e, then it is enumerated into A, and P_e becomes satisfied. If a new computation $J^A(s)$ appears at stage s, then N_e traces its value in T_e and initialises all weaker positive requirements, which will need to be appointed new, large followers.

The key to the success of this construction is that each requirement P_e acts at most once, and does not need to act again even if it is initialised. It may be instructive to think of the priority ordering as dynamic; when P_e acts, then it is removed from the list of requirements and is never troubled (nor does it influence other requirements) again.

To make A jump-traceable via all orders h, a further dynamic element is introduced to the priority ordering. The property of a partial computable function being an order function is Π^0_2, and we approximate it in this fashion. Say that a stage s is *e-expansionary* if at this stage we have further evidence that the e^{th} partial computable function φ_e is an order function. If the stage is indeed e-expansionary then the positive requirements are pushed down the ordering so that for every x such that $\varphi_e(x)\downarrow\ [s]$, there are at most $\varphi_e(x)$ many positive requirements stronger than $N_{e,x}$, the requirement that traces $J^A(e)$ with at most $\varphi_e(x)$ many values. To protect the positive requirements from being moved down infinitely often, we insist that a positive requirement $P_{e'}$ cannot be moved by φ_e if $e' < e$; these positive requirement are ignored when we count the number of positive requirements which appear before some $N_{e,x}$. If $P_{e'}$ acts then we initialise every $N_{e,x}$ and start a new trace. $\square$

The usual variations can apply: Cholak, Downey and Greenberg show that for all low computably enumerable sets B, there is a computably enumerable sjt real $A \not\leqslant_T B$.

Since $A \leqslant_T B$ implies that J^A is uniformly coded into J^B, we have the following.

Theorem 15.5.3 (Figueira, Nies and Stephan [?])**.** *The strongly jump traceable reals are closed downward under Turing reducibility.*

Figueira, Nies and Stephan also studied the relationship between being sjt and other approximations of the jump.

Definition 15.5.4 (Figueira, Nies and Stephan [?])**.** We say that a set D is *well-approximable* iff for each order function h, there is a computable function $g(\cdot,\cdot)$ such that, for all x, $|\{s : g(x, s+1) \neq g(x,s)\}| \leqslant h(x)$ and $D(x) = \lim_s g(x,s)$.

Figueira, Nies and Stephan studied the case where A' is well-approximable. Again it is easy to see that this is closed downwards under Turing reducibility. Notice that we can also study the J^AQ version. Downey and Greenberg define A to be *strongly jump well approximable*, iff J^A is well approximable. It is not known if well-approximable, strongly well-approximable, and strong jump traceability are distinct. For the c.e. case we can easily see the following.

Theorem 15.5.5 (esentially Figueira, Nies and Stephan [?])**.** *Suppose that A is a c.e. set. Then the following are equivalent:*

(i) A is strongly jump traceable.

(ii) A' is well approximable.

(iii) A is strongly jump well approximable.

The reader may wonder what all of this has to do with lowness. Figueira, Nies and Stephan showed that being sjt makes a real have quite low intial segment complexity.

Theorem 15.5.6 (Figueira, Nies and Stephan [**?**]). *(i) If A' is well approximable, then for every order h and almost all x,*

$$C(x) \leqslant C^A(x) + h(C^A(x)).$$

(ii) A is sjt iff for every order h, for almost all x,

$$C(x) \leqslant C^A(x) + h(C^A(x)).$$

(iii) Consequently, every A which has A' well-approximable is strongly jump traceable.

Proof. (i) For any strings x and y we know that $C(x) \leqslant^+ |y| + 2C(x|y)$ In particular if $y = x^*_A$, the minimal A-program for x, we have $C(x) \leqslant^+ |C^A(x)| + 2C(x|C^A(x))$. Let h be an order. It suffices to show that $2C(x|C^A(x)) \leqslant h(C^A(x))$ for almost all x. Now given x^*_A and $C(x|C^A(x))$, we can compute a program p_x with $U(p_x, x^*_A) = x$, and hence x can be computed from $C(x|C^A(x)), x^*_A$, and p_x. The trick is to keep the approximation to the bits of p_x small, as we see below. Define $\Psi^A(m,n,q)$ as follows.

(i) We have $\Psi^A(m,n,q)\uparrow$ until $x = U^A(q)$.

(ii) Find the first program p with $|p| = n$ and $U(p,q) = x$. Unless such a p exists $\Psi^A(m,n,q)\uparrow$.

(iii) If $m \notin [1,n]$, let $\Psi^A(m,n,q)\uparrow$. Finally, define $\Psi^A(m,n,q)\downarrow$ iff the m-bit of p is 1.

Now Ψ^A is part of the A-jump, so we have a canonical index $\alpha(m,n,p)$ so that $J^A(\alpha(m,n,p)) = \Psi^A(m,n,q)$. Now we choose a slow growing order b such that $b(\alpha(m,n,p)) << nh(|q|)$ for all n,q. Then take some approximation for A' with at most $b(x)$ mind changes for all x.

Now let $n_x = C(x|x^*_A)$. Then $\Psi^A(m,n_x,x^*_A)\downarrow$ iff the m-th bit of p_x is 1, where $p_x = U^*(x|x^*_A)$. We know that $p_x = A'(\alpha(1,n_x,x^*_A))\dots A'(\alpha(n_x,n_x,x^*_A))$. This has an approximation that changes at most $n_x \max\{b(\alpha(m,n_x,x^*_A)) : 1 \leqslant m \leqslant n_x\} \leqslant n_x^2 h(|x^*_A|)$ many times, as b is nondecreasing. Therefore since $U(p_x,x^*_A) = x$ and we can describe p_x with n_x, x^*_A and the number of changes in $A'(\alpha(1,n_x,x^*_A))\dots A'(\alpha(n_x,n_x,x^*_A))$, we have the following. $n_x = C(x|x^*_A) \leqslant^+ 2|n_x| + |n_x^2 h(|x^*_A|)| \leqslant^+ 4|n_x| + |h(|x^*_A|)|$.

The result will then foloow by showing that for almost all x, $n_x \leqslant^+ 2|h(|x^*_A|)|$. Since we know that $C(x) \leqslant^+ |C^A(x)| + 2C(x|C^A(x))$, this upper bound implies that

$$C(x) \leqslant^+ |x^*_A| + h(|x^*_A|) = C^A(x) + h(C^A(x)).$$

There is a constant N such that for all $n \geqslant N$, $8|n| \leqslant n$. For almost all x, $|h(|x_A^*|)| \geqslant N$. For such x, we know that either $n_x \leqslant |h(|x_A^*|)|$, or $4|n_x| \leqslant fracn_x 2$. In the second case, $n_x - 4|n_x| \geqslant \frac{n_x}{2}$. Since $n_x \leqslant n_x^2 h(|x_A^*|)$, this would mean that $\frac{n_x}{2} \leqslant^+ |h(|x_A^*|)|$. In both cases we would have $n_x \leqslant^+ 2|h(|x_A^*|)|$.

(ii) First suppose that A is sjt. Let h_0 be any order. It suffices to prove that $C(x) \leqslant^+ \widehat{h}(C^A(x))$ for almost all x, where $h = \lfloor \frac{h_0}{2} \rfloor$, and for any function k, $\widehat{k}n = k(n) + n$. Let α be a coding with $J^A(\alpha(x)) = U^A(z_x)$ where z_x denotes the x-th string in the llex ordering of $2^{<\omega}$. Let T be a trace for J^A with bound g such that $g(n) \leqslant h(|z_n|)$. Take m with $U^A(z_m) = y$ and $|z_m| = C^A(y)$. We know that $y \in T^{(\alpha(m))}$, we can code y with m and a number $\leqslant g(\alpha(m))$, in the enumeration of $T^{(\alpha(m))}$. This uses at most $|z_m| + g(\alpha(m)) \leqslant C^A(y) + h(C^A(y))$ many bits. Thus for all y, $C(y) \leqslant C^A(y) + h(C^A(y))$.

Conversely, suppose that for every order h, for almost all x, $C(x) \leqslant C^A(x) + h(C^A(x))$. By Kolmogorov's Theorem, for all n there is an n with $n \leqslant C(x)$. Let c be a constant so that

$$\forall x[J^A(|x|)\downarrow \rightarrow C^A(x, J^A(x)) \leqslant |x| + c].$$

(The last inequality holds since given x we can compute $J^A(|x|)$ relative to A.) Suppose that h is an order. Define the order g so that for all e, $3^{g(e+c)} \leqslant h(e)$. For almost all x, if $J^A(|x|)\downarrow$, then

$$C(x, J^A(x)) \leqslant \widehat{g}(C^A(x, J^A(|x|)) \leqslant |x| + g(|x| + c) + c.$$

Now we define the trace

$$T^{(e)} = \{y : \forall x(|x| = e \rightarrow C(x, y) \leqslant e + g(e + c) + c\}.$$

Then for almost all e, if $J^A(e)\downarrow$, then $J^A(e) \in T^{(e)}$, sicne for any x with $|x| = e$, $C(x, J^A(e)) \leqslant e + g(e + c) + c$. We show that for almost all x, $|T^{(e)}| \leqslant h(e)$, suppose that $y \in T^{(e)}$. Take x with $|x| = e$ and $C(x) \geqslant e$. Then

$$C(x, y) \leqslant e + g(e + c) + c \leqslant C(x) + g(e + c) + c.$$

By Lemma 6.4.7, for almost all e, there are at most $3^{g(e+c)} \leqslant h(e)$ many y's in $T^{(e)}$. $\square$

15.5.2 The c.e. case: distinctness

The relationship between strong jump traceability and K-triviality was clarified by Cholak, Downey and Greenberg [**?**] for in the computably enumerable case. The first result is that the sjt c.e. reals form a *proper* subclass of the K-trivials.

Theorem 15.5.7 (Cholak, Downey and Greenberg [**?**])**.** *There exists a K-trivail real A which is not h-jump traceable for some order h.*

Proof. The construction of a K-trivial set which is not strongly jump-traceable came out of a construction of a K-trivial set which is not n-c.e. for any n. Since the collection of K-trivial sets is closed downwards under Turing reduction, and so it must contain sets that are not n-c.e. for any n. But how would a direct construction of such a set go?

We know that any construction will essentially be a cost-function construction, such as the by now classic construction of a promptly simple, c.e. K-trivial set mentioned in the introduction. That construction can be redescribed as follows. The e^{th} requirement R_e wishes to show that the set A we construct is not co-c.e. via the e^{th} co-c.e. approximation, namely $\bar{W}_e$. The requirement is given the sum of 2^{-e} which is the capital it is allowed to spend. It appoints a follower x_0, and waits for its realisation, that is, for $x_0 \notin \bar{W}_e$. If, upon realisation, the cost of changing $A(x_0)$ is greater than 2^{-e}, the follower is abandoned, a new one x_1 is picked, and the process repeats itself.

Suppose now that we want to ensure that the constructed set A is not 2-c.e. The e^{th} requirement wants to ensure that A is not 2-c.e. via the e^{th} 2-c.e. approximation $X_e = Y_e \setminus Z_e$ (where both Y_e and Z_e are c.e.) Again the requirement is provided with 2^{-e} much capital to spend. It may appoint a follower x_0 and wait for first realisation, namely x enters X_e (and not Y_e for now.) Provided the price is not too high, the requirement would then extract x_0 from A (we start with $A = \omega$) and wait for second realisation, i.e. x entering Y_e. It would then wish to re-enumerate x_0 into A and thus confirm a win on the requirement. The point here is that the follower needs two "permissions" from the cost-function, and the danger is that we spend some capital on the first action (the extraction), but the second action would be too expensive and the follower would have to be abandoned. The amount we spent on extraction is non-refundable, though, and so this strategy would soon run into trouble.

A better strategy is the following. From the initial sum 2^{-e}, set aside a part (say $2^{-(e+1)}$) which is kept for the second re-enumeration of a follower and will not be used otherwise (for extraction). Of the remaining $2^{-(e+1)}$, we apportion some (say $2^{-(e+2)}$) for the sake of extraction of the first follower x_0. If the cost of extraction of x_0 is higher, then we abandon x_0 (at no cost to us) and allot the same amount $2^{-(e+2)}$ for the extraction of the next follower x_1. Suppose, for example, that we did indeed extract x_1, but when it is realised again and we are ready to re-enumerate it into A, its cost has risen beyond the sum $2^{-(e+1)}$ which we set aside for this task. We have to abandon x_1, appoint a new follower x_2, and start from the beginning. We did lose an uncompensated $2^{-(e+2)}$; so we reduce the sum that we may spend on extracting x_2 to $2^{-(e+3)}$, and keep going.

Between extractions, the sum we may spend on the next extraction is kept constant, and so the usual argument shows that some future follower will get extracted (all this assuming that all followers are realised, of course.) On top of this, abandoning followers upon re-enumeration may happen only finitely many times, because each such abandoned follower x carries a cost of $2^{-(e+1)}$ which comes from descriptions of numbers below the stage at which that follower is abandoned. The next follower x' is appointed only after the previous one is cancelled, and is chosen to be large; the cost associated with x will not be counted toward changing $A(x')$, and so if x' is abandoned upon re-enumeration, this is due to a completely different part of the universal machine which has weight of at least $2^{-(e+1)}$. We can thus see that the process cannot happen more than 2^{e+1} many times.

Figure 15.1. The cost of enumerating x into A at stage s is $\sum_{x<n<s} 2^{-K_s(n)}$ and the cost of enumerating x' into A at stage s' is $\sum_{x'<n<s'} 2^{-K_{s'}(n)}$; the weights are "disjoint".

In fact, we note that the same reasoning may be applied to the extraction steps; new followers are chosen large after we abandon a previous follower upon extraction, and since between extractions the acceptable price is fixed at some 2^{-m}, this kind of abandonment will not happen (between extractions) more than 2^m times. Inductively, we can determine in advance a bound on the number of possible failures, and if we wish, we can distribute the permissible costs evenly, as we do below.

Finally, for $n > 2$, we apply this strategy with n layers of apportioning pieces of capital to various attempts at changing $A(x)$ on some follower x, n many times. To make A not be strongly jump-traceable rather than not n-c.e., what we need to do is to change $J^A(x)$ on some input x more than $h(x)$ many times, where h is some order we will specify in advance (and x is a "slot" in the jump that we control.) To change $J^A(x)$ we need to put the use of this computation into A; keeping A c.e., this means that we change the use, but the principle that the same x receives attention $h(x)$ many times remains and so the same strategy works.

15.5.3 The formal construction and proof

We enumerate a set A and a function p, partial computable in A. The requirement R_e is that $\langle W_e^{[x]}\rangle_{x<\omega}$ is not a trace for p which obeys an order function h, which we soon define. By Lemma **??**, this will suffice to show that A is not strongly jump-traceable.

For $e < \omega$, let T_e consist of all sequences $\langle k_0, k_1, \dots, k_i\rangle$ where $i < e$ and for each $j \leqslant i$ we have $k_j < 2^{e2^j}$. A node $\sigma \in T_e$ is a leaf of T_e iff it has length e. If $\sigma \in T_e$ is not a leaf, then we let $\varepsilon_\sigma = \varepsilon_\sigma^e = 2^{-e2^{|\sigma|}}$.

The idea here is that each leaf on the tree corresponds to a particular attempt at meeting R_e, and if $\sigma \in T_e$ is a leaf, and $i < e$, then $\varepsilon_{\sigma \upharpoonright i}$ is the amount that we would be willing to spend on the $(e-i)^{\text{th}}$ attack with the follower corresponding to σ. The tree T_e and the rationals ε_σ were chosen so that:

1. $\varepsilon_{\langle\rangle} = 2^{-e}$;
2. if $\sigma \in T_e$ is not a leaf, then it has exactly $1/e_\sigma$ many immediate successors on T_e; and further,
3. if $|\sigma| < e-1$ then the sum of ε_τ, as τ ranges over immediate successors of σ on T_e, is ε_σ.

These facts let us, by reverse induction on $|\sigma|$, to show that for $\sigma \in T_e$ which is not a leaf, the sum of ε_τ, as τ ranges over all extensions of σ on T_e which are not leaves, is $(e-|\sigma|)\varepsilon_\sigma$. Thus the sum of ε_τ, as τ ranges over all nodes on T_e which are non-leaves, is $e2^{-e}$. This will be the total amount we let R_e spend; and so the construction will obey the cost-function, as $\sum_{e<\omega} e2^{-e}$ is finite.

We can now define h. Partition ω into intervals $\langle I_e \rangle$ (so $\max I_e + 1 = \min I_{e+1}$), letting the size of I_e be the number of leaves of T_e; we index the elements of I_e as x_σ for leaves σ of T_e. We define $h(x) = e-1$ for all $x \in I_e$.

The requirements R_e act independently. If not yet satisfied at stage s, the requirement R_e will have a pointer $\sigma = \sigma_e[s]$ pointing at some leaf of T_e; the requirement will be conducting an attack with x_σ at some level $i < e$ (the level will be decreasing with time, until the attack is abandoned, or fully succeeds when we get to the root.)

In the beginning, we let $\sigma[0] = 0^e$, the leftmost leaf of T_e (we order the nodes of T_e lexicographically); and we begin an attack with $x_{\sigma[0]}$ on level $e-1$.

The following are the instructions for an attack on level $i < e$ (at a stage s). Let $\sigma = \sigma[s]$. Recall that the cost of enumerating a number x into A at stage s is

$$c(x)[s] = \sum_{n \in (x,s)} 2^{-K(n)[s]}.$$

1. Define $p^A(x_\sigma) = s$ with use $s+1$. Wait for $s \in W_e^{[x_\sigma]}$. [While waiting, if some other requirement puts a number $y \leqslant s$ into A and so makes $p(x_\sigma)$ undefined, redefine $p(x_\sigma)$, again with value s and use s.]
2. At stage $t > s$, s enters $W_e^{[x_\sigma]}$. Compare the cost $c(s)[t]$ of putting s into A at this stage with the permissible waste $\varepsilon_{\sigma\upharpoonright i}$.
   - If $c(s)[t] \leqslant \varepsilon_{\sigma\upharpoonright i}$, then enumerate s into A (making $p(x_\sigma)$ undefined.) Leave σ unchanged and attack with it on level $i-1$. If already $i = 0$ then declare victory and cease all action.

- If $c(s)[t] > \varepsilon_{\sigma \upharpoonright i}$ then we abandon x_σ. Move one step to the right of $\sigma \upharpoonright i+1$. That is, if $\sigma = (k_0, \ldots, k_{e-1})$ then let
$$\sigma[t+1] = (k_0, \ldots, k_{i-1}, k_i + 1, 0, \ldots, 0).$$
Attack with the new σ on level $e-1$.

Justification

We must argue that the above algorithm is consistent: in this case, that if at some stage t we want to abandon an attack with x_σ on level $i < e$ and redefine $\sigma[t+1]$, then the string we defined above is actually on T_e, which will hold iff $k_i + 1 < 1/\varepsilon_{\sigma \upharpoonright i}$.

Fix such an i and σ. Let $\sigma^* = \sigma \upharpoonright i$ and let $m = \sigma(i)$. We know that for all $k \leqslant m$, some attack was made with some string extending $\sigma^{*\frown} k$ (for example with $\sigma^{*\frown} k^\frown (0, \ldots, 0)$;) let τ_k be the rightmost string extending $\sigma^{*\frown} k$ which was ever used for an attack (so $\tau_m = \sigma$); so we know that we attacked with τ_k on level i and that this attack is abandoned. Let s_k be the stage at which the attack with τ_k on level i began, and let $t_k > s_k$ be the stage at which this attack was abandoned (so $t_m = t$).

The key point, as discussed above, is that $t_{k-1} \leqslant s_k$, so the intervals (s_k, t_k) are disjoint. At stage t_k, the attack with τ_k is abandoned because $c(s_k)[t_k] > \varepsilon_{\sigma^*}$. Now

$$1 > \mu\left(\mathrm{dom} U\right) > \sum_{k \leqslant m} \sum_{n \in (s_k, t_k)} 2^{-K(n)} \geqslant \sum_{k \leqslant m} \sum_{n \in s_k, t_k} 2^{-K(n)[t_k]} =$$
$$= \sum_{k \leqslant m} c(s_k)[t_k] > (m+1)\varepsilon_{\sigma^*}.$$

It follows that $m + 1 < 1/\varepsilon_{\sigma^*}$ as required.

Verification

First, note that by the instructions given, for each $e < \omega$, for each $\tau \in T_e$ which is not a leaf, there is at most one $s < \omega$ which is enumerated into A because of a successful attack with some $\sigma \supset \tau$ on level $|\sigma|$. Thus R_e did not spend more than $e2^{-e}$ and so the construction obeys the cost function, making A K-trivial.

Fix $e < \omega$. There are two possible outcomes for R_e.

1. There is some stage s at which we begin an attack with $x_{\sigma[s]}$ at some level, but s never turns up in $W_e^{[x_\sigma]}$. The attack is never concluded. But in this case, no further modifications are made for $p(x_\sigma)$ and it has a final value s, which is not traced.

2. Some attack with some x_σ on level 0 succeeds. This means that
$$\left| W_e^{[x_\sigma]} \right| \geqslant e > h(x_\sigma)$$

and so the trace does not obey the order h.

In either case, we see that R_e is met, and so A is not strongly jump-traceable.

15.5.4 An ideal

We know that the strongly jump-traceable sets are downward closed under Turing reduction. In this section we show that the join of two c.e., strongly jump-traceable sets is also strongly jump-traceable, and so in the c.e. degrees, the strongly jump-traceable degrees form an ideal. In fact, we show that for every order function h there is another order function g such that if sets A_0 and A_1 are c.e. and jump-traceable via g, then $A \oplus B$ is jump-traceable via h.

This result should be contrasted with the following theorem, a proof of which appears in Nies [225], and in essence in Downey, Jockusch, and Stob [**?**]. The original proof was in an unpublished manuscript.

Theorem 15.5.8 (Bickford and Mills [**?**])**.** *Ther exist superlow c.e. sets A and B such that $A \oplus B \equiv_T \emptyset'$.*

Proof. (sketch) The proof is not too difficult. We code $\emptyset'$ into $A \oplus B$ using coding markers $\gamma(n, s)$ so as to build the reduction $\Gamma^{A \oplus B} = \emptyset'$. If n enters $\emptyset'[s]$ then we would put $\gamma(n, s)$ into whichever set does the least overall damage. We will move markers for other reasons, namely to try build a trace $\{V_e : e \in \omega\}$, $|V_e| \leqslant 3^{e+1}$, for example, to satisfy the requirements below for $C \in \{A, B\}$.

$$N_e^C : \Phi_e^C(e) \in V_e.$$

The idea is that if we see a computation $\Phi_e^A(e)[s]\downarrow$, say, we can put this value into $V_{e,s+1}$ and move the markers $\gamma(n, s)$ above s for all $n \geqslant e$, by putting them into the other set $B_{s+1} - B_s$. (Similarly for $C = B$). Then the $\Phi_e^A(e)[s]$ computation can only be injured by N_j^B computations of higher priority than N_e^A and by coding $\gamma(q, s)$ into A for $q \leqslant e$. $\square$

We remark that by work of Bickford and Mills [**?**], and Downey, Jockusch and Stob [**?**], we cannot replace $\equiv_T$ by $\equiv_{wtt}$ in Theorem 15.5.8. In fact Downey, Jockusch and Stob [**?**] proved that the weak truth table degrees of c.e. array computable sets form an ideal in the c.e. wtt degrees, and we have seen that all superlow sets are array computable.

The construction for the jion theorem below is the simplest known example of the *box amplification* (or *promotion*) method, and so we wish to describe the motivation for its discovery. For this, we need to examine the construction of a non-computable, strongly jump-traceable real. The following theorem improves an earlier one of Keng Meng Ng who showed that there was a class of c.e. sets $\mathcal{C}$ such that for all $A \in \mathcal{C}$ and all strongly jump traceable B, $A \oplus B$ was strongly jump traceable.

Theorem 15.5.9 (Join Theorem-Cholak, Downey and Greenberg [**?**])**.** *Suppose that A and B are c.e. reals jump traceable for an order h. then there is an order $\widehat{h}$ such that $A \oplus B$ are jump traceable at order $\widehat{h}$. In particular, if A and B are sjt c.e. reals then so if $A \oplus B$.*

Proof. Consider the proof of the existence of an sjt c.e. real in Theorem 15.5.2. Suppose that we wanted to prove the theorem wrong, that is, to construct c.e. sets A_0 and A_1 which are strongly jump-traceable but such that $A_0 \oplus A_1$ is not. We would presumably attempt to use the strategy of section 15.5.2 and try to diagonalise against possible traces for $\Phi^{A_0 \oplus A_1}$ by changing its values sufficiently many times, this time by enumerating the current use into either A_0 or A_1. In the priority ordering of the requirements we place both these diagonalisation requirements, and the requirements which try to trace J^{A_0} and J^{A_1} as in the construction of a strongly jump-traceable c.e. set.

Again recall that in this construction, after some requirement P_e acts, it gets removed from the list, and the blocks of N_x requirements to its left and to its right are merged; in a sense, this increases the priority of those to the right, because they suffered an injury – which means that the number of times they can be injured has just decreased by one. They have been *promoted*.

In our false construction, suppose we start with the same ordering (except that there are two kinds of negative requirements, one for A_0 and one for A_1.) Each time a positive requirement P_e acts, and say enumerates a number into A_0, it needs to be demoted down the list and placed after all the negative requirements it has just injured; since these requirements may later impose new restraint, a new follower for P_e may be needed each time one such requirement decides to impose restraint. Since some of the negative requirements are also promoted by positive requirements weaker than P_e, we cannot put any computable bound, in advance, on the last place of P_e on the list, and hence, on the number of followers it will need. Thus we cannot state the computable bound which we mean to beat, and the construction fails.

This failure is turned around into our proof. Now we are given two c.e., strongly jump-traceable sets A_0 and A_1, and an order function h, and we wish to trace $J^{A_0 \oplus A_1}$, obeying h. Fix an input e (the requirements that trace $J^{A_0 \oplus A_1}$ act completely independently.) When at some stage of the construction we discover that $J^{A_0 \oplus A_1}(e)$ converges, before we trace the value, we want to receive some confirmation that this value is genuine. Say that the computation has use $\sigma_0 \oplus \sigma_1$, where $\sigma_i \subset A_i[s]$. What we do is define functionals Φ_0 and Φ_1, and define $\Phi_i^{\sigma_i}(x) = \sigma_i$. If indeed $\sigma_i \subset A_i$ then σ_i would appear as a value in a trace T_x^i for $\Phi_i^{A_i}$ which we receive (using the universality of J^{A_i} and the recursion theorem.) Thus we can wait until both strings σ_i appear in the relevant "box" T_x^i, and only then believe the computation $J^{A_0 \oplus A_1}(e)[s]$. Of course, it is possible that both

σ_i appear in T^i_x but that neither σ_i is really an initial segment of A_i; in which case we will have traced the wrong value. In this case, however, both boxes T^i_x have been promoted, in the sense that they contain an element (σ_i) which we know is not the real value of $\Phi_i^{A_i}(x)$, and $\Phi_i^{A_i}(x)$ becomes undefined (when we notice that A_i moved to the right of σ_i) and is therefore useful for us for testing another potential value of $J^{A_0 \oplus A_1}(e)$ which may appear later. If the bound on the size of T^i_x (which we prescribe in advance, but has to eventually increase with x) is k, then we originally think of T^i_x as a "k-box", a box which may contain up to k values; after σ_i appears in T^i_x and is shown to be wrong, we can think of the promoted box as a $k-1$-box. Eventually, if T^i_x is promoted $k-1$ many times, then we have a 1-box; if a string σ_i appears in a 1-box then we know it must be a true initial segment of A_i. In this way we can limit the number of false $J^{A_0 \oplus A_1}(e)$ computations that we trace. Since all requirements act independently, this allows us to trace $J^{A_0 \oplus A_1}$ to any computable degree of precision we may like.

That is the main idea of all "box-promotion" constructions. Each construction is infused with combinatorial aspects which counter difficulties that arise during the construction (difficulties which we think of as possible plays of an opponent, out to foil us.) The combinatorics determine how slowly we want the size of the given trace to grow, and which boxes should be used in every test we make. In this construction, the difficulty is the following: in the previous scenario, it is possible, say, that σ_0 is indeed a true initial segment of A_0, but σ_1 is not an initial segment of A_1. And to make matters worse, the latter fact is discovered even before σ_1 turns up in T^1_x. However, we already defined $\Phi_0^{A_0}(x) = \sigma_0$ with A_0-correct use, which means that the input x will not be available later for a new definition. The box T^0_x has to be discarded, and further, we got no compensation – no other box has been promoted. As detailed below, the mechanics of the construction instruct us which boxes to pick so that this problem can in fact be countered. The main idea (which again appears in all box-promotion constructions) is to use clusters of boxes (or "meta-boxes") rather than individual boxes. Instead of testing σ_i on a single T^i_x, we bunch together a finite collection M_i of inputs x, and define $\Phi_i^{\sigma_i}(x) = \sigma_i$ for all $x \in M_i$. We only believe the computation $J^{A_0 \oplus A_1}(e)$ if σ_i has appeared in T^i_x for all $x \in M_i$. If this is believed and then later discovered to be false, then all of the boxes included in M_i have been promoted; we can then break M_i up into smaller meta-boxes and use each separately; thus we magnify the promotion, to compensate for any losses we may occur on the other side.

15.5.5 The formal construction and proof

In what follows, we fix a number e and show how to trace $J^{A_0 \oplus A_1}(e)$ limiting the errors to a prescribed number m. To do this, given the number m, the requirement will ask for an infinite collection of boxes, and describe precisely how many k-boxes, for each k, it requires for its use (for A_0 and

A_1). As m grows, the least k for which k-boxes are required will grow as well (we denote that number by $k^*(m)$.) For m and $k \geqslant k^*(m)$, let $r(k,m)$ be the number of k-boxes which is required to limit the size of the trace for $J^{A_0 \oplus A_1}(e)$ by m. (In fact, if $k \geqslant k^*(m), k^*(m')$ then we'll actually have $r(k,m) = r(k,m')$, but this is not important.) Again, this means that the requirement will define functionals $\Phi_{e,i}$ (for $i < 2$) and expect to get traces $\langle T_x^{e,i} \rangle_{x<\omega}$ for $\Phi_{e,i}^{A_i}$ which obey a bound h_e, such that for all $k \geqslant k^*(m)$, the collection of x such that $h_e(x) = k$ has size at least $r(k,m)$.

Then, given an order function g, we define an order function $f = f_g$, such that if c.e. sets A_0 and A_1 are jump-traceable via f, then $A_0 \oplus A_1$ are jump-traceable via g. This is done in the following way. For each $c < \omega$, we partition ω into intervals $\langle I_k^c \rangle_{k \geqslant c}$ (so $\min I_{k+1}^c = \max I_k^c + 1$), such that

$$|I_k^c| = \sum_{\{e \,:\, k^*(e) \leqslant k\}} r(k, g(e))$$

and define a function f^c by letting $f^c(x) = k$ if $x \in I_k^c$. Note that since $\lim_e g(e) = \infty$, for any k, for large enough e we have $k^*(g(e)) > k$ and so the prescribed size of I_k^c is indeed finite. It is easy to see that f^c is an order function.

We also note that $f^c(0) = c$. By Lemma **??**, there is an order function f such that for all x and c, $f(\alpha_c(x)) \leqslant f^c(x)$. This is the required function.

Now given A_0 and A_1 which are jump-traceable via f, we get traces S^0, S^1 for J^{A_0}, J^{A_1} which obey f. This allows us, uniformly in c, to get traces $S^{c,0}, S^{c_1}$ for $\Psi_c^{A_0}, \Psi_c^{A_1}$, which obey f^c.

For each c and $k \geqslant c$, let

$$\langle N_{k,e}^c \rangle_{\{e \,:\, k^*(g(e)) \leqslant k\}}$$

be a partition of I_k^c, such that $|N_{k,e}^c| = r(k, g(e))$. For each $c < \omega$, we run the construction for all the e such that $k^*(g(e)) \geqslant c$ simultaneously, with the e^{th} requirement defining $\Phi_{e,0}$ and $\Phi_{e,1}$ with domain contained in $\bigcup_{k \geqslant k^*(g(e))} N_{k,e}^c$ and using $S^{c,0}$ and $S^{c,1}$ as traces. Using Posner's trick, we can effectively get an index c' such that for both $i = 0,1$, $\Phi_i^{A_i} = \bigcup_{\{e \,:\, k^*(g(e)) \geqslant c\}} \Phi_{e,i}^{A_i} = \Psi_{c'}^{A_i}$. By the recursion theorem, there is some c such that $\Psi_c = \Psi_{c'}$ and so indeed $T^i = S^{c,i}$ is a trace for $\Phi_i^{A_i}$, and so for large enough e (those e such that $k^*(g(e)) \geqslant c$) we can get a trace $T^{e,i}$ for $\Phi_{e,i}^{A_i}$ which obeys h_e. For large enough e, this construction will trace $J^{A_0 \oplus A_1}(e)$ with bound $g(e)$. Here end the global considerations; what is left to do is to fix e and m, define $k^*(m)$ and $r(k,m)$ (and so h_e), and describe how, given traces for both $\Phi_{e,i}^{A_i}$ which we define, we can trace $J^{A_0 \oplus A_1}(e)$ with fewer than m mistakes.

The local strategy

So indeed, fix an e and an m. We define functionals $\Phi_{e,i}$ and get traces $\langle T_x^{e,i} \rangle$ for them, as described above, with bound h_e (which we soon define.)

Let $k^*(e) = \llcorner m/2 \lrcorner$. For any n, define a *meta*$_0^n$*-box* to be any singleton $\{x\}$ and define a *meta*$_{k+1}^n$*-box* to be a collection of $n+2$ many meta$_k^n$-boxes. We often ignore the distinction between a meta$_k^n$-box M and $\cup^{(k)} M$, that is, the collection of numbers (inputs) which appear in meta$_0^n$-sub-boxes of M. In this sense, the size of a meta$_k^n$-box is $(n+2)^k$. At the beginning, a meta-box M is an l-box (for either A_0 or A_1) if for all $x \in M$, $h_e(x) \leqslant l$. At a later stage s, a meta-box M is an l-box for A_i if for all $x \in M$, we have $h_e(x) - |T_x^{e,i}[s]| \leqslant l$.

At the beginning, for all $k \geqslant k^*(e)$ we wish to have two meta$_{k+1}^k$-boxes which are k-boxes. We thus let $r(k,m) = 2(k+2)^{k+1}$. Denote these two meta-boxes by N_k and N'_k. From now we drop all e subscripts, so $\Phi_i = \Phi_{e,i}$, $T_x^i = T_x^{e,i}$, $h = h_e$, etc.

At the beginning of a stage s, we have two numbers $k_0^*[s]$ and $k_1^*[s]$ (we start with $k_i^*[0] = k^*(e)$). For $i < 2$, every $k \in [k_i^*[s], s)$ has some *priority* $p_i(k)[s] \in \frac{1}{2}\mathbb{N}$. For such k we have finitely many meta$_k^{\llcorner p_i(k)[s] \lrcorner}$-boxes $M_1^i(k), \ldots, M_{d_i(k)}^i(k)$ $[s]$, each of which is *free* in the sense that for all x in any of these boxes, we have $\Phi_i^{A_i}(x)\uparrow$ $[s]$.

First at stage $s \geqslant k^*(e)$, for both $i = 0,1$ we let $M_1^i(s)[s], \ldots, M_{s+2}^i(s)[s]$ be the meta$_s^s$-sub-boxes of N_s (recall that these are all s-boxes.) We let the priority $p_i(s) = s$.

Suppose now that we are given a computation $J^{A_0 \oplus A_1}(e)[s]$ with use $\sigma_0 \oplus \sigma_1$, which we want to test. The test is done in steps, in increasing priority. We start with step s.

Instructions for testing $\sigma_0 \oplus \sigma_1$ at step $n \in \frac{1}{2}\mathbb{N}$

For $i = 0,1$, if there is some k such that $p_i(k)[s] = n$ (there will be at most one such k for each i), then we take the last meta-box $M = M_{d_i(k)}^i(k)$ $[s]$, and test σ_i on M by defining $\Phi_i^{\sigma_i}(x) = \sigma_i$ for all $x \in M$. We then run the enumeration of the trace T^i and of A_i until one of the following happens:

- For all $x \in M$, σ_i appears in T_x^i (we say that the test *returns*.)
- σ_i is not an initial segment of A_i anymore (we say that the test *fails*.)

One of the two has to occur since T^i is indeed a trace for $\Phi_i^{A_i}$.

If all tests that were started (either none, one test for one σ_i, or two tests for both σ_i) have returned, then we move to test at step $n - 1/2$; but if $n = 1$ then all tests at all levels have returned, and so we believe the computation $J^{A_0 \oplus A_1}(e)[s]$ and trace it. In the latter case, from now we monitor this belief; we just keep defining $p_i(s')$ and $M_j^i(s')$ at later stages s'. If at a later stage t we discover that one of the σ_i was not in fact an initial segment of A_i, we update priorities as follows and go back to following the instructions above.

Also, if some test at step n fails, then we stop the testing at stage s and update priorities.

Updating priorities

Suppose that at some stage s, a test of σ_i at step n returns, but at a stage $t \geqslant s$ we discover that $\sigma_i \not\subset A_i$. Let k be the level such that $p_i(k)[s] = n$. We do the following:

1. If $k = k_i^*[s]$ then let $k_i^*[t+1] = k - 1$.
2. Redefine $p_i(k-1)[t+1] = n$ and $d_i(k-1)[t+1] = \llcorner n \lrcorner + 2$, and let $M_1^i(k-1)[t+1], \dots, M_{\llcorner n \lrcorner+2}^i(k-1)[t+1]$ be the collection of meta$_{k-1}^{\llcorner n \lrcorner}$-sub-boxes of $M_{d_i(k)[s]}^i$ (which was the meta$_k^{\llcorner n \lrcorner}$-box used for the testing of σ_i at step n of stage s.)
3. If $k = s$ then redefine $p_i(k)[t+1] = s+1/2$, redefine $d_i(s)[t+1] = s+2$ and let $M_1^i(s)[t+1], \dots, M_{s+2}^i(s)[t+1]$ be the meta$_s^s$-sub-boxes of N_s' (note that these were untouched so far.)

On the other side, if at stage t we still have $\sigma_{1-i} \subset A_{1-i}[t]$, and a test of σ_{1-i} at stage s at step n has started (and so returned), then we need to discard the meta-box $M_{d_{1-i}(k)}^{1-i}(k)[s]$ (where again $p_{1-i}(k)[s] = n$) and redefine $d_{1-i}(k)[t+1] = d_{1-i}(k)[s] - 1$. We do this also if $t = s$ and the first test at step s has returned, but we immediately found out that $\sigma_i \not\subset A_i$, and the test on the σ_i side did not even return once.

Justification and verification

Let $i < 2$, $s \geqslant k^*(e)$, $k \in [k_i^*[s], s)$, and $j \in \{1, \dots, d_i(k)[s]\}$. Let $n = p_i(k)[s]$.

Lemma 15.5.10. *The meta$_k^{\llcorner n \lrcorner}$-box $M_j^i(k)[s]$ is a k-box; indeed, for all $x \in M_j^i(k)[s]$, there are at least $\llcorner n \lrcorner - k$ many strings in $T_x^i[s]$ which lie to the left of $A_i[s]$ (and $h(x) = \llcorner n \lrcorner$).*

Proof. Let s be the least such that we define, for some level k, $p_i(k)[s] = n$. Then $k = \llcorner n \lrcorner$ and there are two possibilities:

- If $n \in \mathbb{N}$, then $s = n$, the definition is made at the beginning of stage s, and we define $M_1^i(s), \dots, M_{s+2}^i(s)$ $[s]$ to be sub-boxes of N_s, which is an s-box.
- If $n \notin \mathbb{N}$ then at stage $s-1$, a test that began at stage $\llcorner n \lrcorner \leqslant s-1$ (and returned on the σ_i side) is resolved by finding that $\sigma_i \not\subset A_i[s]$. We

then define $p_i(\llcorner n \lrcorner)[s] = n$ and define $M^i_1(\llcorner n \lrcorner), \dots, M^i_{\llcorner n \lrcorner+2}(\llcorner n \lrcorner)$ $[s]$ to be sub-boxes of $N'_{\llcorner n \lrcorner}$, which is an $\llcorner n \lrcorner$-box.

In either case, the $M^i_j(\llcorner n \lrcorner)[s]$ are $\llcorner n \lrcorner$-boxes, so indeed for each x in these meta-boxes, $h(x) = \llcorner n \lrcorner$, and T^i_x indeed contains at least $\llcorner n \lrcorner - \llcorner n \lrcorner$ many strings.

By induction, if $n = p_i(k)[t]$ at a later stage t, then for all j, $M^i_j(k)[t]$ is a sub-box of some $M^i_{j'}(\llcorner n \lrcorner)[s]$, and so for all $x \in M^i_j(k)[t]$ we have $h(x) = \llcorner n \lrcorner$.

Suppose that at stage t we redefine $p_i(k-1)[t+1] = n$ and redefine $M^i_j(k-1)[t+1]$. Then at some stage $r \leqslant t$ we defined, for all $x \in M^i_{d_i(k)}[s]$, $\Phi^{\sigma_i}_i(x) = \sigma_i$ where $\sigma_i \subset A_i[r]$ but $\sigma_i \not\subset A_i[t+1]$. By induction, at stage r there are at least $\llcorner n \lrcorner - k$ many strings in $T^i_x[r]$ that lie to the left of $A_i[r]$; they all must be distinct from σ_i. The test at stage r returned, which means that $\sigma_i \in T^i_x[r+1]$; thus $T^i_x[t+1]$ contains at least $\llcorner n \lrcorner - (k-1)$ many strings that lie to the left of $A_i[t+1]$. □

Lemma 15.5.11. *The sequence $k^*_i[s]$ is non-increasing with s; for all s we have $k^*_i[s] > 0$.*

Proof. By Lemma 15.5.10, for all $j \leqslant d_i(k^*_i)[s]$ and $x \in M^i_j(k^*_i)[s]$ we have $|T^i_x| \geqslant h(x) - k^*_i[s]$; as $|T^i_x| < h(x)$ we must have $k^*_i[s] > 0$. □

Lemma 15.5.12. *The sequence $p_i(k)[s]$ is strictly increasing with k.*

Proof. Assume this at the beginning of stage t. We first define $p_i(t)[t] = t$; all numbers used prior to this stage were below t.

Now suppose that at stage t we update priorities because of a test which returns at some stage $s \leqslant t$ is found to be incorrect. The induction hypothesis for s, and the instructions for testing, ensure that the collection of levels k for which a σ_i-test has returned at stage s is an interval $[k_0, s]$. Priorities then shift one step downward to the interval $[k_0 - 1, s-1]$; the sequence of priorities is still increasing. Finally, a new priority $s + 1/2$ is given to level s; it is greater than the priorities for levels $k < s$ (which get priority at most s) but smaller than the priority k which is given to all levels $k \in (s, t]$. □

Also note that we always have $p_i(k) \geqslant k$ because we start with $p_i(k)[k] = k$, then perhaps later change it to $k + 1/2$, and from then on it never decreases.

The following key calculation ensures that we never run out of boxes at any level, on either side, so the construction can go on and never get stuck. It ties losses of boxes on one side to gains on the other. For any $k \in [k^*_i[s], s]$, let $l_i(k)[s]$ be the least level l such that $p_{1-i}(l) \geqslant p_i(k)$ $[s]$. Such a level must exist because at the beginning of the stage we let $p_{1-i}(s) = s$, which is greater or equal to $p_i(k)[s]$ for any $k \leqslant s$. Thus $l_i(k)[s] \geqslant 1$.

Lemma 15.5.13. *At stage s, for $i < 2$ and $k \in [k_i^*[s], s]$, the number $d_i(k)[s]$ of meta-k-boxes is at least:*

- $l_i(k)[s]$, *if* $p_{1-i}(l_i(k)) > p_i(k)\,[s]$;
- $l_i(k)[s] + 1$, *if* $p_{1-i}(l_i(k)) = p_i(k)\,[s]$.

Proof. This goes by induction on the stage. Suppose this is true at the end of stage $t-1$; we consider what changes we may have at stage t.

First at stage t, we define $p_i(t) = t = p_{1-i}(t)$. We thus have $l_i(t) = t$ and $p_{1-i}(l_i(t)) = p_i(t)$ and so we are required to have $t+1$ many t-boxes; we actually have $d_i(t)[t] = t+2$ many.

Suppose that a test which began at stage $s \leqslant t$ is resolved at stage t, and priorities are updated.

There are two sides. Suppose first that $\sigma_i \not\subset A_i[t+1]$, and that $d_i(k)[t+1] \neq d_i(k)[t]$. If $k < s$, then a test at level $k+1$ returned at stage s. We then redefine $d_i(k)[t+1] = \llcorner n \lrcorner + 2$ where $n = p_i(k)[t+1]$ $(= p_i(k+1)[s])$. As mentioned, we always have $p_{1-i}(\ulcorner n \urcorner) \geqslant \ulcorner n \urcorner$ and so $l_i(k)[t+1] \leqslant \llcorner n \lrcorner + 1$, so we're in the clear. If, however, $k = s$, then we redefine $d_i(s)[t+1] = s+2$ and $p_i(s)[t+1] = s+1/2$; again, $p_{1-i}(s+1)[t+1] \geqslant s+1$ and so $l_i(k)[t+1] \leqslant s+1$, so $d_i(s) \geqslant l_i(k) + 1$ $[t+1]$ as required.

Now take the losing side: suppose that $\sigma_i \subset A_i[t+1]$. We may have lost some meta-boxes on this side; but changing priorities on the other side give us compensation. Let $k \in [k_i^*[t], t]$; before anything else, we note that if $k > s$ then $d_i(k)[t+1] = k+2$, $l_i(k)[t+1] = k$ and $p_{1-i}(k)[t+1] = k$, so there are sufficiently many k-boxes. We assume then that $k \leqslant s$.

We also examine the case that $k = s$. In this case, $d_i(k)[t+1] = d_i(k)[s] - 1 = s+1$. We have $p_i(k)[t+1] = s$ and $l_i(k)[t+1] \leqslant s$ and so $d_i(k) \geqslant l_i(k) + 1$ $[t+1]$. We assume from now that $k < s$.

Let $n = p_i(k)[t+1] = p_i(k)[s]$. We note that if there is no k' such that $p_{1-i}(k')[s] = n$, then there is no k' such that $p_{1-i}(k')[t+1] = n$. This is because the only priority we may add at stage t (after the initial part of the stage) is $s + 1/2$, and $n < s$. Thus, if $n' = p_{1-i}(l_i(k))[s] > n$ then $p_{1-i}(l_i(k))[s] \geqslant n' > n$, because there are three possibilities for the behaviour of $l_i(k)$ and $p_{1-i}(l_i(k))$. Let $k' = l_i(k)[s]$, and note that $k' < s$.

1. A test for σ_{1-i} at step n' of stage s returns. In this case, $l_i(k)[t+1] = k' - 1$ and $p_{1-i}(l_i(k))[t+1] = n'$.
2. A test for σ_{1-i} at level k' (at stage s) does not return, but a test for σ_{1-i} at level $k'+1$ does return. In this case the priority n' is removed on side $1-i$ at stage t; we redefine $p_{1-i}(k')[t+1] = p_{1-i}(k'+1)[s] > n'$. However, we still have $l_i(k)[t+1] = k'$ because (if $k' > k_{1-i}^*[s]$) we still have $p_{1-i}(k'-1)[t+1] = p_{1-i}(k'-1)[s] < n$.

3. A test for σ_{1-i} at level $k'+1$ is not started or does not return. In this case there is no change at level k' and $k'-1$; we have $l_i(k)[t+1] = k'$ and $p_{1-i}(k')[t+1] = n'$.

In any case, we see that we cannot have a case at which $l_i(k)$ increases from stage s to stage $t+1$, or that $p_{1-i}(l_i(k))[s] > n$ but $p_{1-i}(l_i(k))[t+1] = n$. Thus the required number of k-meta-boxes does not increase from stage s to stage $t+1$. Thus we need only to check what happens if $d_i(k)[t+1] = d_i(k)[s] - 1$. Assume this is the case; we check each of the three scenarios above.

In case (1), the number of required boxes has decreased by one; this exactly compensates the loss. Case (3) is not possible if a k-box is lost; this is because a test at step n is started only after a test for σ_{1-i} at step $p_{1-i}(k'+1)[s]$ has returned.

The same argument shows that if case (2) holds and we lost a k-box, then necessarily $n' = n$. But then $d_i(k)[s] \geqslant k'+1$, but the fact that now $p_{1-i}(k')[t+1] > n$ implies that the number of required boxes has just decreased by one, to k'; again the loss is compensated. □

We are now ready to finish. We note that if indeed $J^{A_0 \oplus A_1}(e)$ converges, then at some point the correct computation appears and is tested. Of course all tests must return, and so the correct value will be traced.

If, on the other hand, a value $J^{A_0 \oplus A_1}(e)[s]$ is traced at stage s because all tests return, but at a later stage t we discover that this computation is incorrect, say $\sigma_i \not\subset A_i[t+1]$, then $k_i^*[t+1] < k_i^*[t]$. As we always have $k_i^*[r] \geqslant 1$, this must happen fewer than $2k^*(e) \leqslant m$ many times. It follows that the total number of values traced is at most m, as required. □

□

15.5.6 Strongly jump-traceable c.e. sets are K-trivial

Theorem 15.5.14 (Cholak, Downey and Greenberg [**?**])**.** *There is an order h such that ‘ if A is c.e. and jump traceable at order h, then A is K-trivial.*

Proof. Let A be strongly jump-traceable; we prove that it is low for K, and hence K-trivial. We need to cover U^A by an oracle-free machine, obtained via the Kraft-Chaitin theorem. We enumerate A and thus approximate U^A. When a string σ enters the domain of U^A we need to decide whether we believe the A-computation that put σ in domU^A; again the idea is to test this by testing the use $\rho \subset A[s]$ which enumerated σ into dom$U^A[s]$; again the naive idea is to pick some input x and define a functional $\Psi^\rho(x) = \rho$. Then Ψ^A is traced by a trace $\langle T_x \rangle$; only if ρ is traced do we believe it is indeed an initial segment of A and so believe that $U^A(\sigma)$ is a correct computation. We can then enumerate $(|\sigma|, U^A(\sigma))$ into a Kraft-Chaitin set we build and so ensure that $K(U^A(\sigma)) \leqslant^+ |\sigma|$.

The combinatorics of the construction aim to ensure that we indeed build a Kraft-Chaitin set; that is, the amount of mass that we believe is finite. This would of course be ensured if we only believed correct computations, as $\mu(\mathrm{dom}U^A)$ is finite. However, the size of most T_x is greater than 1, and so an incorrect ρ may be believed. We need to limit the mass of the errors.

To handle this calculation, rather than treat each string σ individually, we batch strings up in pieces of mass. When we have a collection of strings in $\mathrm{dom}U^A$ whose total mass is 2^{-k} we verify A up to a use that puts them all in $\mathrm{dom}U^A$. The greater 2^{-k} is, the more stringent the test will be (ideally, in the sense that the size of T_x is smaller). We will put a limit m_k on the amount of times that a piece of size 2^{-k} can be believed and yet be incorrect. The argument will succeed if

$$\sum_{k<\omega} m_k 2^{-k}$$

is finite.

Once we use an input x to verify an A-correct piece, it cannot be used again for any testing, as $\Psi^A(x)$ becomes defined permanently. Following the naive strategy, we would need at least 2^k many inputs for testing pieces of size 2^{-k}. Even a single error on each x (and there will be more, as the size of T_x has to go to infinity) means that $m_k \geqslant 2^k$ is too large. Again, the rest of the construction is a combinatorial strategy: which inputs are assigned to which pieces in such a way as to ensure that the number of possible errors m_k is sufficiently small. The strategy has two ingredients.

First, we note that two pieces of size 2^{-k} can be combined into a single piece of size $2^{-(k-1)}$. So if we are testing one such piece, and another piece, with comparable use, appears, then we can let the testing machinery for $2^{-(k-1)}$ take over. Thus, even though we need several testing locations for 2^{-k} (for example if a third comparable piece appears), at any stage, the testing at 2^{-k} is really responsible for at most one such piece.

The naive reader would imagine that it is now sufficient to let the size of T_x (for x testing 2^{-k}-pieces) be something like k and be done. However, the opponent's spoiling strategy would be to "drip-feed" small mass that aggregates to larger pieces only slowly (this is similar to the situation in decanter constructions.) In particular, fixing some small 2^{-k}, the opponent will first give us k pieces (of incomparable use) one after the other (so as to change A and remove one before giving us a new one.) At each such occurrence we would need to use the input x devoted to the first 2^{-k} piece, because at each such stage we only see one. Once the amount of errors we get from using x for testing is filled (T_x fills up to the maximum allowed size) the opponent gives us one correct piece of size $2^{-(k-1)}$ and then moves on to gives us k more incorrect pieces which we test on the next x. Overall, we get k errors on *each* x used for 2^{-k}-pieces. As we already agreed that we need something like 2^k many such x's, we are back in trouble.

Every error helps us make progress as the opponent has to give up one possible value in some T_x; fewer possible mistakes on x are allowed in the future. The solution is to make every single error count in our favour in all future testings of pieces of size 2^{-k}. In other words, what we need to do is to maximize the benefit that is given by a single mistake; we make sure that a single mistake on *some* piece will mean one less possible mistake on *every* other piece. In other words, we again use meta-boxes.

In the beginning, rather than just testing a piece on a single input x, we test it simultaneously on a large set of inputs and only believe it is correct if the use shows up in the trace of every input tested. If this is believed and more pieces show up then we use them on other large sets of inputs. If, however, one of these is incorrect, then we later have a large collection of inputs x for which the number of possible errors is reduced. We can then break up this collection into 2^k many smaller collections and keep working only with such x's.

This can be geometrically visualised as follows. If the naive strategy was played on a sequence of inputs x, we now have an m_k-dimensional cube of inputs, each side of which has length 2^k. In the beginning we test each piece on one hyperplane. If the testing on some hyperplane is believed and later found to be incorrect then from then on we work in that hyperplane, which becomes the new cube for testing pieces of size 2^{-k}; we test on hyperplanes of the new cube. If the size of T_x for each x in the cube is at most m_k then we never "run out of dimensions".

15.5.7 The formal construction and proof

Given $c < omega$ (say $c > 1$), we partition ω into intervals $\langle M_k^c\rangle_{k<\omega}$ such that $|M_k^c| = 2^{k(k+c)}$. For $x \in M_k^c$ we let $h_c(x) = k + c - 1$. By Lemma **??**, we get an order function $\tilde{h}$ such that for all c and x, $\tilde{h}(\alpha_c(x)) \leqslant h_c(x)$. We fix a trace for J^A with bound $\tilde{h}$. From this trace, we can, uniformly in c, get a trace for Ψ_c^A with bound h_c.

In our construction, we define a functional Ψ; by the recursion theorem we know some c such that for all $X \in 2^\omega$, $\Psi^X = \Psi_c^X$. We let $M_k[0] = M_k^c$ and let $\langle T_x\rangle$ be the trace for Ψ^A with bound $h = h_c$.

Usage of Ψ. Again, the axioms that we enumerate into Ψ are all of the form $\Psi^\rho(x) = \rho$ for some $\rho \in 2^{<\omega}$ and $x < omega$. We only enumerate such an axiom at stage s if $\rho \subset A[s]$.

Let $R_k = \left\{m2^{-k} \,:\, m = 0, 1, 2, \ldots, 2^k\right\}$, and let $R_k^+ = R_k \setminus \{0\}$.

The boxes

We can label the elements of $M_k[0]$ so that

$$M_k[0] = \left\{x_f \;\;:\;\; f\colon (k+c) \to R_k^+\right\}.$$

[So $M_k[0]$ is a $(k+c)$-dimensional cube; the length of each side is 2^k.]

At stage s, for each k we have a function $g_k[s]\colon d_k[s] \to R_k^+$ (where $d_k[s] < k+c$) which determines the current value of M_k:

$$M_k[s] = \{x_f \in M_k[0] \,:\, g_k[s] \subset f\}$$

(so $d_k[0] = 0$ and $g_k[s]$ is the empty function.) Thus $M_k[s]$ is a $(k+c-d_k)$-dimensional cube.

For $q \in R_k^+$, we let

$$N_k(q)[s] = \{x_f \in M_k[s] \,:\, f(d_k[s]) = q\}\,;$$

this is the $(2^k \cdot q)^{\text{th}}$ hyper-plane of $M_k[s]$.

Strings

Recall that for any string $\rho \in 2^{<\omega}$, we let Ω^ρ be the measure of the domain of U^ρ, the universal machine with oracle ρ. Note that $\rho \mapsto \Omega^\rho$ is monotone: if $\rho \subset \nu$ then $\Omega^\rho \leqslant \Omega^\nu$. We assume that the running time of any computation with oracle $|\rho|$ is at most $|\rho|$ steps, and so:

- The maps $\rho \mapsto U^\rho$ and so $\rho \mapsto \Omega^\rho$ are computable;
- For all $\sigma \in \mathrm{dom} U^\rho$, $|\sigma| \leqslant |\rho|$.

It follows that Ω^ρ is a multiple of $2^{-|\rho|}$, in other words, is an element of $R_{|\rho|}$. Also note that since $\langle\rangle \notin U^X$ for any X, the assumption implies that $U^{\langle\rangle}$ is empty and so $\Omega^{\langle\rangle} = 0$.

Let q be any rational. For any $\nu \in 2^{<\omega}$ such that $\Omega^\nu \geqslant q$, we let $\varrho^\nu(q)$ be the shortest string $\rho \subseteq \nu$ such that $\Omega^\rho \geqslant q$. This operation is monotone with q: if $q < q'$ and $\Omega^\nu \geqslant q'$ then $\varrho^\nu(q) \subseteq \varrho^\nu(q')$.

The standard configuration

At the beginning of stage s of the construction, we are given A at some point of its enumeration, which we denote by $A[s]$ (more than one number may go into A at each stage, as we describe below.)

At the beginning of the stage, the cubes $\langle M_k \rangle$ will be in the *standard configuration* for the stage. Fix $k \leqslant s$ and $q \in R_k^+$.

- If $q \leqslant \Omega^{A[s]\restriction s}$ then for all $x \in N_k(q)[s]$ we have $\Psi^\rho(x)\downarrow = \rho\,[s]$, where $\rho = \varrho^{A[s]\restriction s}(q)$.
- If $q > \Omega^{A[s]\restriction s}$ then for all $x \in N_k(q)\,[s]$, we have $\Psi^{A[s]}(x)\uparrow\ [s]$.

Further, for all $k > s$ and all $x \in M_k\,[s]$, no definition of $\Psi(x)$ (for any oracle) was ever made.

Suppose that $\rho \subseteq A[s] \restriction s$. We say that ρ is *semi-confirmed* at some point during stage s if for all x such that $\Psi^\rho(x)\downarrow = \rho$ at stage s, we have

$\rho \in T_x$ at that given point (which may be the beginning of the stage or later.) We say that ρ is *confirmed* if every $\rho' \subseteq \rho$ is semi-confirmed.

Note that the empty string is (emptily) confirmed at every stage. This is because for no x do we ever define $\Psi^{\langle\rangle}(x)\downarrow = \langle\rangle$; this is because $\Omega^{\langle\rangle} = 0$ and so for no s and no $q > 0$ do we have $\langle\rangle = \varrho^{A[s]\restriction s}(q)$.

Construction

At stage s, do the following:

1. Speed up the enumeration of A and of $\langle T_x\rangle$ (to get $A[s+1]$ and $T_x[s+1]$) so that for all $\rho \subseteq A[s] \restriction s$, one of the following holds:
   (a) ρ is confirmed.
   (b) ρ is not an initial segment of A anymore.

   One of the two must happen because $\langle T_x\rangle$ traces Ψ^A.

2. For any $k \leqslant s$, look for some $q \in R_k^+$ such that $q \leqslant \Omega^{A[s]\restriction s}$ and such that for $\rho = \varrho^{A[s]\restriction s}(q)$ we have:
   - ρ was confirmed at the beginning of the stage; but
   - $\rho \not\subset A[s+1]$.

   If there is such a q, pick one, and extend g_k by setting $g_k(d_k) = q$. Thus $d_k[s+1] = d_k[s] + 1$ and $M_k[s+1] = N_k(q)[s]$.

3. Next, define Ψ as necessary so that the standard configuration will hold at the beginning of stage $s+1$.

Justification

We need to explain why the construction never gets stuck. There are two issues:

1. Why don't we "run out of dimensions"? That is, why can we always increase d_k if we are asked to?

2. Why can we always return to the next standard configuration?

For the first, we prove the following.

Lemma 15.5.15. *For every $x \in M_k[s]$, there are at least $d_k[s]$ many strings $\rho \in T_x[s]$ which lie (lexicographically) to the left of $A[s]$.*

Proof. Suppose that during stage s, we increase d_k by one. This is witnessed by some $q \in R_k^+$ and a string $\rho = \varrho^{A[s]\restriction s}(q)$ which was confirmed at the beginning of the stage; we set $M_k[s+1] = N_k(q)[s]$. The confirmation implies that for all $x \in N_k(q)[s]$, $\rho \in T_x$. But we also know that $\rho \subset A[s]$ and $\rho \not\subset A[s+1]$. As A is c.e., it had to move to the right of ρ. If we increase d_k at stages $s_1 < s_2$ (witnessed by strings ρ_1 and ρ_2) then ρ_1 lies to the left of $A[s_1+1]$ whereas ρ_2 is an initial segment of $A[s_2]$ (which is not left of $A[s_1+1]$.) Thus ρ_1 lies to the left of ρ_2, and in particular, they are distinct. $\square$

Since for all $x \in M_k[0]$, $h(x) = k+c-1$, we know that for all such x, $|T_x| \leqslant k+c-1$, which implies that for all s we must have $d_k[s] < k+c$.

For the second issue, let $k < \omega$.

If $M_k[s+1] \neq M_k[s]$, witnessed by some $q \in R_k^+$ and by $\rho = \varrho^{A[s]\restriction s}(q)$, then for all $x \in M_k[s+1]$ we know that $\Psi^\rho(x)\downarrow = \rho$; so for no proper initial segment $\rho \subsetneq \rho$ do we have $\Psi^{\rho'}(x)\downarrow\ [s]$. As ρ is not an initial segment of $A[s+1]$ we must have $\Psi^{A[s+1]}(x)\uparrow$ so we are free to make any definitions we like (recall that no definitions to right of $A[s]$ are made before stage s.)

For $k = s+1$, we know that M_k was empty up to stage s, so we have a clean slate there.

Suppose that $k \leqslant s$ and that $M_k[s+1] = M_k[s]$. Let $q \in R_k^+$ such that $q \leqslant \Omega^{A[s+1]\restriction s+1}$, and let $x \in N_k(q)[s+1]$ $(= N_k(q)[s])$. We want to define $\Psi^\rho(x)\downarrow = \rho$ where $\rho = \varrho^{A[s+1]\restriction s+1}(q)$.

If $\rho \not\subset A[s]$ then ρ lies to the right of $A[s]$, and so $\Psi^\rho(x)\uparrow$ for all $x \in M_k[s]$. Suppose that $\rho \subset A[s]$. There are two possibilities:

1. If $|\rho| \leqslant s$ then $\rho = \varrho^{A[s]\restriction s}(q)$ and so we already have $\Psi^\rho(x)\downarrow = \rho$ for all $x \in N_k(q)[s]$.
2. If $|\rho| = s+1$ then (since we know that for every proper initial segment ρ' of ρ we have $q > \Omega^{\rho'}$) we have $q > \Omega^{A[s]\restriction s}$. Since the standard configuration held at the beginning of stage s, we have $\Psi^{A[s]}(x)\uparrow$ at the beginning of the stage (for all $x \in N_k(q)$). Thus we are free to define $\Psi^\rho(x)$ as we wish.

This concludes the justifications.

Verification

Let s be a stage. We let $\rho^*[s]$ be the longest string (of length at most s) which is a common initial segment of both $A[s]$ and $A[s+1]$. Thus $\rho^*[s]$ is the longest string which is confirmed at the beginning of stage $s+1$.

We define

$$L = \bigcup \left\{ U^{\rho^*[s]} \ :\ s < \omega \right\}.$$

This is a c.e. set. We will show that:

1. $U^A \subseteq L$; and that
2. $$\sum_{(\sigma,\tau)\in L} 2^{-|\sigma|}$$ is finite.

The second fact shows that $\{(|\sigma|, \tau) : (\sigma, \tau) \in L\}$ is a Kraft-Chaitin set and so there is some constant e such that for all $(\sigma, \tau) \in L$, $K(\tau) \leqslant |\sigma| + e$. Together with the first fact, we see that A is low for K: for all τ, $K(\tau) \leqslant K^A(\tau) + e$.

Let us first verify (1).

Lemma 15.5.16. *$U^A \subseteq L$.*

Proof. Suppose that $U^A(\sigma) = \tau$. Let $\rho \subset A$ some string such that $U^\rho(\sigma) = \tau$. Let $s > |\rho|$ be late enough so that $\rho \subset A[s], A[s+1]$. Then $\rho \subseteq \rho^*[s]$ and so $(\sigma, \tau) \in L$. □

Next we verify (2). Now L has two parts: U^A and $L \setminus U^A$. We know of course that $\mu\left(\mathrm{dom} U^A\right)$ is finite, and so we need to show that

$$\sum_{(\sigma,\tau)\in L\setminus U^A} 2^{-|\sigma|}$$

is finite.

Let s be a stage. For $k \leqslant s$, let $q_k[s]$ be the greatest element of R_k not greater than $\Omega^{\rho^*[s]}$. This is monotone: if $k < k' \leqslant s$ then $q_k[s] \leqslant q_{k'}[s]$ because $R_k \subset R_{k'}$. Note that $|\rho^*[s]| \leqslant s$ and so $\Omega^{\rho^*[s]}$ is an integer multiple of 2^{-s}; it follows that $q_s[s] = \Omega^{\rho^*[s]}$. Also, since for all ρ we have $\Omega^\rho < 1$, we must have $q_0[s] = 0$.

Let $\nu_k[s] = \varrho^{\rho^*[s]}(q_k[s])$. By the monotonicity just mentioned, if $k < k' \leqslant s$ then $\nu_k[s] \subseteq \nu_{k'}[s]$ and so $\Omega^{\nu_k[s]} \leqslant \Omega^{\nu_{k'}[s]}$. Also, $\nu_0[s] = \langle\rangle$, and $\Omega^{\nu_s[s]} = \Omega^{\rho^*[s]}$ (so $U^{\nu_s[s]} = U^{\rho^*[s]}$).

The following is the key calculation.

Lemma 15.5.17. *For all $k \in \{1, 2, \ldots, s\}$,*

$$\Omega^{\nu_k[s]} - \Omega^{\nu_{k-1}[s]} \leqslant 2 \cdot 2^{-k}.$$

Proof. We know that $q_{k-1}[s] \leqslant \Omega^{\nu_{k-1}[s]}$ and that $\Omega^{\nu_{k-1}[s]} \leqslant \Omega^{\nu_k[s]}$. On the other hand, $\Omega^{\nu_k[s]} \leqslant \Omega^{\rho^*[s]}$ and $\Omega^{\rho^*[s]} \leqslant q_{k-1}[s] + 2^{-(k-1)}$. So overall,

$$q_{k-1}[s] \leqslant \Omega^{\nu_{k-1}[s]} \leqslant \Omega^{\nu_k[s]} \leqslant q_{k-1}[s] + 2 \cdot 2^{-k}.$$ □

If $(\sigma, \tau) \in L \setminus U^A$, then we will find some $k < \omega$ and some stage t and "charge" the mistake of adding (σ, τ) to L against k at stage t; we denote the collection of charged mass by $L_{k,t}$. Formally, we will define sets $L_{k,t}$ and show that:

1. For each k and t, the mass of $L_{k,t}$, namely

   $$\sum_{(\sigma,\tau)\in L_{k,t}} 2^{-|\sigma|},$$

   is at most $2 \cdot 2^{-k}$.

2.
$$L \setminus U^A \subseteq \bigcup_{k,t} L_{k,t}.$$

3. For each k, there are at most $k+c$ many stages t such that $L_{k,t}$ is non-empty.

Given these facts, we get that
$$\sum_{(\sigma,\tau)\in L\setminus U^A} 2^{-|\sigma|} \leqslant \sum_{k,t} \sum_{(\sigma,\tau)\in L_{k,t}} 2^{-|\sigma|} \leqslant \sum_k 2(k+c)2^{-k}$$
which is finite as required. We turn to define $L_{k,t}$ and to prove (1)–(3).

Fix t and k such that $1 \leqslant k \leqslant t$. If $\nu_k[t] \not\subset A[t+2]$ then we let
$$L_{k,t} = U^{\nu_k[t]} \setminus U^{\nu_{k-1}[t]}.$$
Otherwise, we let $L_{k,t} = \emptyset$.

Fact (1) follows from Lemma 15.5.17:
$$\sum_{(\sigma,\tau)\in L_{k,t}} 2^{-|\sigma|} = \mu\left(\mathrm{dom}\left(U^{\nu_k[t]} \setminus U^{\nu_{k-1}[t]}\right)\right) = \Omega^{\nu_k[t]} - \Omega^{\nu_{k-1}[t]} \leqslant 2\cdot 2^{-k}.$$

Lemma 15.5.18.
$$L \setminus U^A \subseteq \bigcup_{k,t} L_{k,t}.$$

Proof. Let $(\sigma,\tau) \in L \setminus U^A$.

Let ρ be the shortest string such that $(\sigma,\tau) \in U^\rho$ and for some s, $\rho \subset \rho^*[s]$. Find such a stage s (so $\rho \subset A[s], A[s+1]$). Since $\rho \not\subset A$, there is a stage $t \geqslant s$ such that $\rho \subset A[t], A[t+1]$ but $\rho \not\subset A[t+2]$.

Since $\rho \subset \rho^*[t]$ and $U^{\rho^*[t]} = U^{\nu_t[t]}$, by minimality of ρ, we have $\rho \subseteq \nu_t[t]$. Since $\nu_0[t] = \langle\rangle$, there is some $k \in [1,t]$ such that $\nu_{k-1}[t] \subsetneq \rho \subseteq \nu_k[t]$.

Since $\rho \subseteq \nu_k[t]$, we have $(\sigma,\tau) \in U^{\nu_k[t]}$. Since $\nu_{k-1}[t] \subset \rho^*[t]$, the minimality of ρ implies that $(\sigma,\tau) \notin U^{\nu_{k-1}[t]}$. Finally, $\rho \not\subset A[t+2]$ and so $\nu_k[t] \not\subset A[t+2]$. Thus $(\sigma,\tau) \in L_{k,t}$. □

Finally, we prove fact (3) by showing the following:

Lemma 15.5.19. *Suppose that $L_{k,t} \neq \emptyset$. Then $M_k[t+1] \neq M_k[t+2]$.*

Proof. Suppose that $L_{k,t} \neq \emptyset$, so $\nu_k[t] \not\subset A[t+2]$. Let $q = q_k[t]$. Then $\nu_k[t] = \varrho^{\rho^*[t]}(q) = \varrho^{A[t]\upharpoonright t}(q)$. Since $\nu_k[t] \subseteq \rho^*[t]$, it was confirmed at the beginning of stage $t+1$. Also, $q > 0$ because otherwise $\nu_k[t] = \langle\rangle$ and then $U^{\nu_k[t]}$, and so $L_{k,t}$, would be empty.

But then all the conditions for redefining M_k during stage $t+1$ are fulfilled. □

□

We remark that using more intricate combinatorics it is also possible to establish the following.

Theorem 15.5.20 (Cholak, Downey and Greenberg [**?**]). *There is an order h such that if A is c.e. and jump traceable at order h, then A is not Martin-Löf cuppable above $\mathbf{0}'$ via any incomplete random degree.*

Recent work by Ng has proven the following.

Theorem 15.5.21 (Keng Meng Ng [224]). *The c.e. sjt reals form a Π^0_4 complete set of reals.*

Ng has also studied relativized versions of these concepts. For example we say that A is *ultra jump traceable* iff A is strongly jump traceable relative to all c.e. X. Ng has shwon that no real is jump traceable realtive to all Δ^0_2 sets. Ng has shown that the ultra jump traceable reals form a proper subclass of the sjt reals. They *cannot* be promptly simple, the first class defined by a cost function construction with this property. This class is not yet understood.

15.5.8 The general case

From the previous sections, we see that the c.e. sjt reals are reasonably well understood. The general case is not yet understood. Being jump traceable for an order h does not even imply that a real is Δ^0_2.

Theorem 15.5.22 (Nies [225]). *There are $2^{\aleph_0}$ many reals which are jump traceable for order 2.4^n. These can be constructed as the set of paths through a fixed perfect Π^0_1 class, and all mjump traceable relative to a fixed trace $\{V_e : e \in \omega\}$..*

Proof. The following proof is drawn from Nies [225]. We build a sequence of maps $F_s : 2^{<\omega} \to 2^{<\omega}$, such that $\lim_s F_s = F$ exists. Let $F_0(\nu) = \nu$. At stage $s+1$, look for the lex least ν, with $|\nu| = e$ and $J^{F_s(\tau)}(e) \downarrow \notin V_{e,s}$ for some τ with $\nu \preccurlyeq \tau$. Enumerate $J^{F_s(\tau)}(e) \searrow V_{e,s+1}$, and for all $\rho \in 2^{<\omega}$, define $F_{s+1}(\nu\rho) = F_s(\tau\rho)$. Since the value of $F_t(\nu)$ can change at most $2^{e+1} - 1$ many times, we see that $\lim_s F_s(nu)$ exists for all ν and by construction the jump computations along each path is traced. $\square$

However, only countably many reals are *strongly* jump traceable. In fact more is true. The proof of the result below is an intricate extension of the mathods used in the c.e. case.

Theorem 15.5.23 (Downey and Greenberg [**?**]). *There is an order, roughly, $h(n) = \log\log n$, such that the collection of reals jump traceable at order h is Δ^0_2.*

15.6 On the low for Ω reals

In Chapter 18, we studied Ω as an operator, and Omega operators in general. In the present section we will look at lowness for Ω alone. We begin with the question of trying to determine for which oracles $A \in 2^\omega$ is there a universal prefix-free oracle machine U such that Ω_U^A is a left-c.e. real? We show that this is true for almost every A.

Definition 15.6.1 (Nies, Stephan Terwijn [232], Downey, Hirschfeldt, Miller, Nies [80])**.** We say that Ω is A-random for some—or equivalently any—version of Ω, then $A \in 2^\omega$ is said to be *low for* Ω.

Theorem 15.6.2 (Downey, Hirschfeldt, Miller, Nies [80])**.** *$A \in 2^\omega$ is low for Ω iff there is a universal prefix-free oracle machine U such that Ω_U^A is a left-c.e. real.*

Proof. First assume that there is a universal prefix-free oracle machine U such that $X = \Omega_U^A$ is a left-c.e. real. Every 1-random left-c.e. real is Solovay equivalent to Ω. So $X \leqslant_S \Omega$, which means that $X \leqslant_S^A \Omega$. Both X and Ω are left-c.e. reals, hence they are A-left-c.e. reals. Applying Proposition 18.3.1, because X is A-random, Ω is also A-random. Therefore, A is low for Ω.

For the other direction, assume that $A \in 2^\omega$ is low for Ω. Then Ω is A-random and an A-left-c.e. real. By Corollary 18.3.4, $\Omega = \Omega_U^A$ for some universal prefix-free oracle machine U. $\square$

It follows from the proof and Proposition 18.2.5 that if A is low for Ω, then $\Omega \oplus A \equiv_T A'$. Therefore $A' \equiv \emptyset' \oplus A$, giving a second proof of Corollary **??**: low for Ω reals are GL_1.

Almost every real is low for Ω; in particular, every 2-random real is.

Theorem 15.6.3 (Nies, Stephan, Terwijn [232])**.** *A 1-random real $A \in 2^\omega$ is low for Ω iff A is 2-random.*

Proof. Assume that $A \in 2^\omega$ is 1-random. Recall that $\Omega \equiv_T \emptyset'$. So A is 2-random iff A is Ω-random iff Ω is A-random, where the last equivalence follows from van Lambalgen's theorem. $\square$

More evidence for the ubiquity of low for Ω reals is the following basis theorem. It is an immediate corollary of Theorem 18.5.1 and Proposition 15.6.2, and is due to Downey, Hirschfeldt, Miller and Nies [80].

Corollary 15.6.4 (The low for Ω basis theorem)**.** *Every nonempty Π_1^0 class contains a $\emptyset'$-left-c.e. real which is low for Ω.*

Every K-trivial real is low for 1-random, hence low for Ω. However, by the previous result applied to the Π_1^0 class of completions of Peano arithmetic, there is also a low for Ω real which is neither 2-random nor K-trivial.

Although it is a degression from our primary topic, we finish this section with a generalization of Corollary 15.6.4. The following result is a "low for X" basis theorem for every 1-random real $X \in 2^\omega$; it reduces to the corollary when we take $X = \Omega$. This result was found independently by Reimann and Slaman [?], for whom it is not a digression but a useful lemma.

Proposition 15.6.5 (Downey, Hirschdeldt, Miller, Nies [80], Reimann and Slaman [?])**.** *For every 1-random $X \in 2^\omega$ and every nonempty Π^0_1 class $\mathcal{P} \subseteq 2^\omega$, there is an X-left-c.e. real $A \in \mathcal{P}$ such that X is A-random.*

Proof. Let $\mathcal{P} \subseteq 2^\omega$ be a nonempty Π^0_1 class. Our goal is to construct a Martin-Löf test $\{\mathcal{V}_i\}_{i\in\omega}$ such that if $X \in 2^\omega$ is not A-random for any $A \in \mathcal{P}$, then $X \in \bigcap_{i\in\omega} \mathcal{V}_i$. Fix a universal prefix-free oracle machine U. Whenever an $s \in \omega$ and $\sigma \in 2^{<\omega}$ are found such that

$$(\forall A \in \mathcal{P}_s)(\exists \tau \preccurlyeq \sigma)\; K^A_U(\tau) \leqslant |\tau| - i,$$

then put $[\sigma]$ into $\mathcal{V}_i$. Clearly, each $\mathcal{V}_i$ is a Σ^0_1 class. Fix an $A \in \mathcal{P}$ and note that $\mathcal{V}_i \subseteq \{X \mid (\exists n)\; K^A_U(X \restriction n) \leqslant n - i\}$. Therefore $\mu(\mathcal{V}_i) \leqslant 2^{-i}$, so $\{\mathcal{V}_i\}_{i\in\omega}$ is a Martin-Löf test. Finally, assume that $X \in 2^\omega$ is not A-random for any $A \in \mathcal{P}$. By compactness, for every $i \in \omega$, there is an $\sigma \prec X$ such that $[\sigma] \subseteq \mathcal{V}_i$. Hence, $X \in \bigcap_{i\in\omega} \mathcal{V}_i$.

This proves that if $X \in 2^\omega$ is 1-random, then there is an $A \in \mathcal{P}$ such that X is A-random. We must still prove that A can be taken to be an X-left-c.e. real. For every $i \in \omega$, let $\mathcal{S}_i = \{A \in \mathcal{P} \mid (\forall n)\; K^A_U(X \restriction n) > n - i\}$. Note that each $\mathcal{S}_i$ is a $\Pi^0_1[X]$ class. We proved above that $\mathcal{S}_i$ is nonempty, for large enough $i \in \omega$. So $A = \min(\mathcal{S}_i)$ is an X-left-c.e. real satisfying the theorem. $\square$

15.7 Weakly low for K and strong Chaitin randomness

In this section we will look at a new concept, namely weak lowness, and find surprising connections with the previous sections and with 2-randomness culminating in a proof of Theorem 9.7.14 that strong Chaitin randomness and 2-randomness coincide, and another theorem of Miller that 2-random reals are infinitely often quite trivial as oracles with respect to prefix-free complexity.

Definition 15.7.1 (Miller [214])**.** We say that a real A is *weakly low for K* iff

$$\exists^\infty n (K(n) \leqslant^+ K^(n)).$$

15.7.1 Weak lowness and Ω

Theorem 15.7.2 (Miller [214])**.** *A is weakly low for K iff A is low for Ω.*

Proof. The following proof follows Miller [214]. First we prove that if A is weakly low for K then A is low for Ω. We prove the contrapositive. First, we define a two families of c.e. sets $\{W_\sigma\}_{\sigma\in 2^{<\omega}}$ and $\{D_\sigma\}_{\sigma\in 2^{<\omega}}$. Fix $\sigma \in 2^{<\omega}$. Search for the least stage $s \in \omega$ such that $\sigma \prec \Omega_s$; if no such stage is found, then W_σ and D_σ will be empty. Now, for any $\tau \in 2^{<\omega}$ such that $U(\tau)\downarrow$ after stage s, enumerate $\langle |\tau|, U(\tau)\rangle$ into D_σ. Also enumerate $\langle |\tau|, U(\tau)\rangle$ into W_σ as long as it preserves the condition that $\sum_{\langle d,n\rangle\in W_\sigma} 2^{-d} \leqslant 2^{-|\sigma|}$. Note that if $K_s(n) \neq K(n)$, then $\langle K(n), n\rangle \in D_\sigma$.

We claim that if $\sigma \prec \Omega$, then $W_\sigma = D_\sigma$. It follows from our definition that $\sum_{\langle d,n\rangle\in D_\sigma} 2^{-d} \leqslant \Omega - \Omega_s$. Observe that if $\sigma \prec \Omega$, then $\Omega - \Omega_s \leqslant 2^{-|\sigma|}$. In this case, $\sum_{\langle d,n\rangle\in D_\sigma} 2^{-d} \leqslant 2^{-|\sigma|}$, so $W_\sigma = D_\sigma$. The idea is that we have used an approximation of Ω to efficiently approximate all but finitely many values of $K(n)$.

Next, consider the A-c.e. set

$$W = \{\langle d + |\tau| - |\sigma|, n\rangle \colon U^A(\tau) = \sigma \text{ and } \langle d,n\rangle \in W_\sigma\}.$$

By the construction of $\{W_\sigma\}_{\sigma\in 2^{<\omega}}$,

$$\sum_{\langle e,n\rangle\in W} 2^{-e} = \sum_{U^A(\tau)\downarrow=\sigma} \; \sum_{\langle d,n\rangle\in W_\sigma} 2^{-d-|\tau|+|\sigma|} \leqslant \sum_{U^A(\tau)\downarrow} 2^{-|\tau|} \leqslant 1.$$

This proves that W is a KC set relative to A. Therefore, there is a constant $k \in \omega$ such that if $\langle e,n\rangle \in W$, then $K^A(n) \leqslant e + k$.

Now, assume that Ω is not A-random. For any $c \in \omega$, there are $\tau, \sigma \in 2^\omega$ such that $U^A(\tau) = \sigma$, $|\sigma| - |\tau| \geqslant c$ and $\sigma \prec \Omega$. Let $s \in \omega$ be the least stage such that $\sigma \prec \Omega_s$. There is an $N \in \omega$ such that if $n \geqslant N$, then $K_s(n) \neq K(n)$ (by the usual conventions on stages, $N = s+1$ is sufficient). For all $n \geqslant N$, $\langle K(n), n\rangle \in W_\sigma$, hence $\langle K(n) + |\tau| - |\sigma|, n\rangle \in W$. But this means that $K^A(n) \leqslant K(n) + |\tau| - |\sigma| + k \leqslant K(n) + k - c$, for all but finitely many n. But c was arbitrary, so A is not weakly low for K.

For the other direction, we will use a simple lemma.

Lemma 15.7.3. *Let A be an oracle. If V is any $\Sigma^0_1[A]$ class, then there is a $\Sigma^0_1[A]$ class $\widehat{V}$ such that:*

1. *$\mu(\widehat{V}) \leqslant 3\mu(V)$.*
2. *If X is an endpoint of an open interval in V, then $X \in \widehat{V}$.*

Furthermore, an index for $\widehat{V}$ can be found uniformly from an index from V and is independent of A.

Proof. Let $\widehat{V} = \bigcup\{(a-\varepsilon, a+2\varepsilon) \colon [a, a+\varepsilon] \subseteq V\}$. It is easy to check that $\widehat{V}$ has the required properties. □

Now we need to prove that if A is low for Ω, then A is weakly low for K. Assume that A is not weakly low for K. Let $U \subseteq 2^\omega$ be a Σ^0_1 class such that $\mu(U) \leqslant 1/2$ and $2^\omega \setminus U$ contains only 1-random reals. For example, we could take $U = U_1$, where $\{U_n\}_{n\in\omega}$ is a universal Martin-Löf test. Let $X = \inf(2^\omega \smallsetminus U)$. Because X is a 1-random c.e. real, we know that $X \equiv_S \Omega$ by the Kučera-Slaman Theorem, Theorem 13.3.4. We will prove that X is not A-random. It follows that Ω is not A-random, and hence that A is not low for Ω.

For each n, we define a $\Sigma^0_1[A]$ class V_n such that $\mu(V_n) \leqslant 2^{-n-2}$. It will not be the case that $X \in V_n$; in fact, we will have $V_n \subseteq U$. On the other hand, it *will* always be true that X is an endpoint of an open interval in V_n. We claim that this is sufficient. By the previous lemma, we can form a computable sequence $\{\widehat{V_n}\}_{n\in\omega}$ of $\Sigma^0_1[A]$ classes such that $X \in \bigcap_{n\in\omega} \widehat{V_n}$ and $\mu(\widehat{V_n}) \leqslant 3\mu(V_n) \leqslant 3\cdot 2^{-n-2} < 2^{-n}$. Therefore, X is covered by a Martin-Löf test relative to A, so X is not A-random.

We turn to the definition of $\{V_n\}_{n\in\omega}$. Assume that $U = \bigcup_{s\in\omega}[\sigma_s]$, where $\{\sigma_s\}_{s\in\omega}$ is a prefix-free computable sequence of strings. Fix $n \in \omega$. If $m = |\sigma_s|$, put $[\sigma_s]$ into V_n as long as σ_s is among the first $2^{m-K^A(m)-n-2}$ strings of length m in $\{\sigma_s\}_{s\in\omega}$. In other words, V_n is built from the same sequence that defines U but with the restriction that strings of length m can contribute at most $2^{-K^A(m)-n-2}$ to its measure. Note that the stage-wise approximations to $2^{m-K^A(m)-n-2}$ approach it from below, so V_n is $\Sigma^0_1[A]$. Also note that

$$\mu(V_n) \leqslant \sum_{m\in\omega} 2^{-K^A(m)-n-2} = 2^{-n-2}\sum_{m\in\omega} 2^{-K^A(m)} \leqslant 2^{-n-2},$$

where the last step uses Kraft's inequality.

Next we prove that there is a $v \in \omega$ such that if $|\sigma_s| \geqslant v$, then $[\sigma_s] \subseteq V_n$. Let $J(m) = |\{s \in \omega\colon |\sigma_s| = m\}|$. We claim that $I(m) = m - \log(J(m))$ is an information content measure. Clearly, I is computable from above. Note that $2^{-I(m)} = J(m)2^{-m}$ is exactly the contribution to the measure of U made by the strings in $\{\sigma_s\}_{s\in\omega}$ of length m. Since $\{\sigma_s\}_{s\in\omega}$ is prefix-free, $\sum_{m\in\omega} 2^{-I(m)} = \mu(U) \leqslant 1/2$. This shows that I is an information content measure, so there is c such that $(\forall m)\ K(m) \leqslant I(m) + c$. Because A is not weakly low for K, there is a v large enough that $K^A(m) \leqslant K(m) - c - n - 2$, for all $m \geqslant v$. For such m,

$$2^{m-K^A(m)-n-2} \geqslant 2^{m-K(m)+c} \geqslant 2^{m-I(m)} = 2^{\log(J(m))} = J(m).$$

Therefore, $[\sigma_s]$ is put into V_n as long as $|\sigma_s| \geqslant v$.

We can now show that X is an endpoint of an open interval in V_n. This is because X is not a binary rational and thus not an endpoint of $[\sigma_s]$, for any s. Since there are only finitely many strings in $\{\sigma_s\}_{s\in\omega}$ of length less than v, there is an ε small enough such that $(X-\varepsilon, X)$ is disjoint from all corresponding intervals. But $(X-\varepsilon, X) \subseteq U$, so $(X-\varepsilon, X) \subseteq V_n$. This completes the proof. $\square$

Corollary 15.7.4 (The weakly low for K basis theorem, Miller [214]). *Every nonempty Π^0_1 class has a weakly low for K member.*

Proof. This now follows from Corollary 15.6.4. □

The following corollary improves an earlier unpublished result of Miller that 3-random reals are weakly low for K.

Corollary 15.7.5 (Nies, Stephan and Terwijn [232]+Miller [214]). *A random real is weakly low for K iff A is 2-random.*

Proof. This follows from Theorem 15.7.2 and the Nies, Stephan, Terwijn [232] Thorem, Theorem [**?**] □

15.7.2 Strong Chaitin randomness

We recall from Chapter 9 that $A \in 2^\omega$ is *strongly Chaitin random* iff $(\exists^\infty n)\ K(A \upharpoonright n) \geqslant^+ n + K(n)$. Recall also that A is Koomogorov random iff $\exists^\infty n C(A \upharpoonright n) \geqslant^+ n$. In that section, we have seen by Lemma 9.7.7 that every 3-random real is strongly Chaitin random. Furthermore by Solovay's Theorem, we know that every strongly Chaitin random real is Kolmogorov random. Finally we know by Theorem 9.7.11, 2-randomness and Kolmogorov randomness coincide. Thus it was a fundamental question whether strong Chaitin randomness and Kolmogorov randomness coincided. Notice that coincidence for reals would contrast greatly with Solovay's proof that that there are Kolmogorov random strings which are *not* strongly Chaitin random. Using the above material Joe Miller proved that Kolmogorov random reals are indeed strongly Chaitin random.

Theorem 15.7.6 (Miller [214]). *If A is 2-random then it is strongly Chaitin random.*

Proof. Assume that A is 2-random. Then A is 1-random relative to $\emptyset'$. Since $\emptyset' \equiv_T \Omega$, we have that A is Ω-random. By van Lambalgen's theorem, Ω is A-random. In other words, A is low for Ω. (We have already see a proof that 2-random is low for Ω in Theorem 15.6.3.) Theorem 15.7.2 implies that A is weakly low for K.

By the Ample Excess Lemma, Theorem 9.4.1, $K^A(n) \leqslant^+ K(A \upharpoonright n) - n$. Rearranging, we have $K(A \upharpoonright n) \geqslant^+ n + K^A(n)$. Because A is weakly low for K, there are infinitely many n such that $K^A(n) \geqslant^+ K(n)$. Note that $K(A \upharpoonright n) \geqslant^+ n + K(n)$ for these n, so A is strongly Chaitin random. □

15.8 Ω^A for K-trivial A

In the previous section, we considered the reals which can be mapped to left-c.e. reals by *some* Omega operator. Now we look at $A \in 2^\omega$ such that Ω_U^A is a left-c.e. real for *every* universal prefix-free oracle machine U. We will see that these are exactly the K-trivial reals.

The lemma below is a spinoff of the golden run construction from [225, Theorem 6.2]. It holds for any prefix-free oracle machine M.

Lemma 15.8.1 (Nies [231]). *Let $A \in 2^\omega$ be K-trivial. Then there is a computable sequence of stages $q(0) < q(1) < \dots$ such that*

$$\widehat{S} = \sum\{\widehat{c}(x,r) \mid x \text{ is minimal s.t. } A_{q(r+1)}(x) \neq A_{q(r+2)}(x)\} < \infty, \quad (15.6)$$

where

$$\widehat{c}(z,r) = \sum\left\{2^{-|\sigma|} \;\middle|\; \begin{array}{c} M^A(\sigma)[q(r+1)]\downarrow \;\wedge \\ z < \mathrm{use}(M^A(\sigma)[q(r+1)]) \leqslant q(r) \end{array}\right\}.$$

Informally, $\widehat{c}(x,r)$ is the maximum amount that $\Omega_M^A[q(r+1)]$ can decrease because of an $A(x)$ change after stage $q(r+1)$, provided we only count the $M^A(\sigma)$ computations with use $\leqslant q(r)$.

Theorem 15.8.2 (Downey, Hirschfeldt, Miller, Nies [80]). *Let U be a universal prefix-free oracle machine. The following are equivalent for $A \in 2^\omega$:*

1. *A is K-trivial.*
2. *A is Δ_2^0 and Ω_U^A is a left-c.e. real.*
3. *$A \leqslant_T \Omega_U^A$.*
4. *$A' \equiv_T \Omega_U^A$.*

Proof. (ii) $\Longrightarrow$ (iii) follows from the fact that each 1-random left-c.e. real is Turing complete. (iii) $\Longrightarrow$ (i) because A is a base for 1-randomness; see the end of Section **??**. (iii) is equivalent to (iv) by Proposition 18.2.5.

(i) $\Longrightarrow$ (ii). Assume that A is K-trivial. It is known that A is Δ_2^0. We show that there is a $r_0 \in \omega$ and an effective sequence $\{\omega_r\}_{r\in\omega}$ of rationals such that $\Omega_U^A = \sup_{r\geqslant r_0} \omega_r$, and hence Ω_U^A is a left-c.e. real. Applying Lemma 15.8.1 to U, we obtain a computable sequence of stages $q(0) < q(1) < \dots$ such that (15.6) holds. The desired sequence of rationals is

$$\omega_r = \sum\{2^{-|\sigma|} \mid U^A(\sigma)[q(r+1)]\downarrow \;\wedge\; \mathrm{use}(U^A(\sigma)[q(r+1)]) \leqslant q(r)\}.$$

Thus ω_r measures the computations existing at stage $q(r+1)$ whose use is at most $q(r)$. We define r_0 below; first we verify that $\Omega_U^A \leqslant \sup_{r\geqslant r_0} \omega_r$ for any $r_0 \in \omega$. Given $\sigma_1, \dots, \sigma_m \in \mathrm{domain}(U^A)$, choose $r_1 \in \omega$ so that each computation $U^A(\sigma)$ has settled by stage $q(r_1)$, with use $\leqslant q(r_1)$. If $r \geqslant r_1$, then $\omega_r \geqslant \sum_{1\leqslant i\leqslant m} 2^{-|\sigma_i|}$. Therefore, $\Omega_U^A \leqslant \limsup_{r\in\omega} \omega_r \leqslant \sup_{r\geqslant r_0} \omega_r$.

Now define a Solovay test $\{I_r\}_{r\in\omega}$ as follows: if x is minimal such that $A_{q(r+1)}(x) \neq A_{q(r+2)}(x)$, then let

$$I_r = [\omega_r - \widehat{c}(x,r), \omega_r].$$

Then $\sum_{r\in\omega} |I_r|$ is finite by (15.6), so $\{I_r\}_{r\in\omega}$ is indeed a Solovay test. Also note that, by the comment after the lemma, $\min I_r \leqslant \max I_{r+1}$ for each $r \in \omega$.

Since Ω_U^A is 1-random, there is an $r_0 \in \omega$ such that $\Omega_U^A \notin I_r$ for all $r \geqslant r_0$. We show that $\omega_r \leqslant \Omega_U^A$ for each $r \geqslant r_0$. Fix $r \geqslant r_0$. Let $t \geqslant r$ be the first non-deficiency stage for the enumeration $t \mapsto A_{q(t+1)}$. Namely, if x is minimal such that $A_{q(t+1)}(x) \neq A_{q(t+2)}(x)$, then

$$(\forall t' \geqslant t)(\forall y < x)\ A_{q(t'+1)}(y) = A_{q(t+1)}(y).$$

The quantity $\omega_t - \widehat{c}(x,t)$ measures the computations $U^A(\sigma)[q(t+1)]$ with use $\leqslant x$. These are stable from $q(t+1)$ on, so $\Omega_U^A \geqslant \min I_t$. Now $\Omega_U^A \notin I_u$ for $u \geqslant r_0$ and $\min I_u \leqslant \max I_{u+1}$ for any $u \in \omega$. Applying this to $u = t-1, \ldots, u = r$, we obtain that $\Omega_U^A \geqslant \max I_r = \omega_r$. Therefore, $\Omega_U^A \geqslant \sup_{r \geqslant r_0} \omega_r$. □

One consequence of this theorem is the fact that every Omega operator is degree invariant at least on the K-trivial reals. The next example shows that they need not be degree invariant anywhere else.

Example 15.8.3. There is an Omega operator which is degree invariant only for K-trivial reals.

Proof. Let M be a prefix-free oracle machine such that

$$\Omega_M^A = \begin{cases} A, & \text{if } A(0) = 0 \\ 0, & \text{if } A(0) = 1. \end{cases}$$

For any $A \in 2^\omega$, define a real $\widehat{A}$ by $\widehat{A}(n) = A(n)$ iff $n \neq 0$. Define a universal prefix-free oracle machine V by $V^A(00\sigma) = U^A(\sigma)$, $V^A(01\sigma) = U^{\widehat{A}}(\sigma)$ and $V^A(1\sigma) = M^A(\sigma)$, for all $\sigma \in 2^{<\omega}$. Then $|\Omega_V^A - \Omega_V^{\widehat{A}}| = A/2$, for all $A \in 2^\omega$. Assume that $\Omega_V^{\widehat{A}} \leqslant_T \Omega_V^A$ for some $A \in 2^\omega$. Then $A \leqslant_T \Omega_V^A$, so A is a base for 1-randomness and hence K-trivial by [81]. If $\Omega_V^A \leqslant_T \Omega_V^{\widehat{A}}$, then again A is K-trivial. Therefore, if $A \in 2^\omega$ is not K-trivial, then $\Omega_V^A \mid_T \Omega_V^{\widehat{A}}$. □

The following corollary summarizes Theorem 15.8.2 and Example 15.8.3.

Corollary 15.8.4 (Downey, Hirschfeldt, Miller, Nies [80])**.** *The following are equivalent for $A \in 2^\omega$:*

1. *A is K-trivial.*
2. *Every Omega operator takes A to a left-c.e. real.*
3. *Every Omega operator is degree invariant on $\mathbf{deg}_T(A)$.*

We have seen in Theorem 18.5.7 that no Omega operator is degree invariant. We have also seen that if $A \in 2^\omega$ is not K-trivial, then there are Omega operators which are not invariant on $\mathbf{deg}_T(A)$. Can these two results be combined?

Question 15.8.5. *For a universal prefix-free oracle machine U and a real $A \in 2^\omega$ which is not K-trivial, is there a $B \equiv_T A$ such that $\Omega_U^B \not\equiv_T \Omega_U^A$?*

Finally, a simple but interesting consequence of Example 15.8.3 is the following.

Corollary 15.8.6 (Downey, Hirschfeldt, Miller, Nies [80]). *Every K-trivial is a d.c.e. real (i.e., the difference of two left-c.e. reals).*

Proof. Let V be the machine from the example. Assume that $A \in 2^\omega$ is K-trivial. Then Ω_V^A and $\Omega_V^{\widehat{A}}$ are both left-c.e. reals by Theorem 15.8.2. Therefore, $A = 2\,|\Omega_V^A - \Omega_V^{\widehat{A}}|$ is a d.c.e. real. $\square$

It is known that the d.c.e. reals form a real closed field [224, 241]. The corollary gives us a nontrivial real closed subfield: the K-trivials reals. To see this, note that the K-trivial reals form an ideal in the Turing degrees ([81] for closure under $\oplus$ and [225] for downward closure). Because a zero of an odd degree polynomial can be computed relative to the coefficients, the K-trivial reals are also a real closed field.

15.9 The Csima-Montalbán function

In this section, we give the original proof of the existence of a minimal pair of K-degrees. That is, we construct a pair of non-K-trivial sets A and B such that for all X, if $X \leqslant_{\mathrm{K}} A, B$ then X is K-trivial. We met such a pair constructed by Merkle and Stephan indexMerkle, W. in Theorem 13.10.7. The construction below due to Csima and Montalbán [58] and involves a lemma of significant independent interest.

A reasonable strategy for building such a pair of sets would be to ensure that, for some constant c and all n,

$$K(A \upharpoonright n) \leqslant K(n) + c \quad \text{or} \quad K(B \upharpoonright n) \leqslant K(n) + c, \tag{15.7}$$

while preventing A and B from being K-trivial. For any string σ, we know that $\sigma 0^\omega$ is K-trivial, so we could try to do the following. We start building A and B (thought of as sequences that we build up over time) by making A look random and adding 0's to B. Then, once we have ensured that A has an initial segment of fairly high K-complexity, we start adding 0's to A to bring down the K-complexity of its initial segments, while still adding 0's to B. Once we have reached an m such that $K(A \upharpoonright m)$ is sufficiently low, we start making B look random while adding 0's to A. Once we have ensured that B has an initial segment of fairly high K-complexity, we start adding 0's to B to bring down the K-complexity of its initial segments, while still adding 0's to A. Once we have reached an n such that $K(B \upharpoonright n)$ is sufficiently low, we go back to the beginning, making A look random while adding 0's to B.

The problem with such a construction is that, although $\sigma 0^\omega$ is indeed K-trivial for any σ, *the constant of K-triviality depends on σ*. Thus, while adding 0's to a previously defined initial segment of A might bring the complexity of longer initial segments down quite a bit, it may never result in an n such that $K(A \upharpoonright n) \leqslant K(n) + c$ for a given c fixed ahead of time.

The clever solution to this problem, found by Csima and Montalbán [58], is to prove the following surprising "gap result", which shows that we can replace c in (15.7) by a sufficiently slow-growing function of n.

Theorem 15.9.1 (Csima and Montalbán [58]). *There is an unbounded nondecreasing function f such that the following are equivalent.*

(i) *X is K-trivial.*

(ii) *$K(X \upharpoonright n) \leqslant K(n) + f(n)$ for almost all n.*

Proof. Let $\mathrm{KT}(e)$ denote the class of sets that are K-trivial with constant e. Obviously, (i) implies (ii) for any unbounded nondecreasing function f. For the other direction, for each $e > 0$, we build an unbounded nondecreasing function f_e such that $f_e(0) = e-1$ and if $K(X \upharpoonright n) \leqslant K(n)+f_e(n)$ for all n, then $X \in \mathrm{KT}(e)$. Given such functions, let $f(n) = \min\{f_{2e}(n) - e : e > 0\}$. It is easy to check that f is nondecreasing and unbounded. Furthermore, if $K(X \upharpoonright n) \leqslant K(n) + f(n)$ for almost all n then there is an $e > 0$ such that $K(X \upharpoonright n) \leqslant K(n) + f(n) + e$ for all n, which means that $K(X \upharpoonright n) \leqslant K(n) + f_{2e}$ for all n, and hence $X \in \mathrm{KT}(2e)$.

Fix $e > 0$. We want to define a sequence $0 = n_0 < n_1 < \cdots$ and let $f_e(k) = e + i - 1$ whenever $k \in [n_i, n_{i+1})$. First let $n_0 = 0$. Let n_1 be such that for any set $Y \in \mathrm{KT}(e+1) - \mathrm{KT}(e)$, there is an $m < n_1$ with $K(Y \upharpoonright m) > K(m) + e$. Such a number must exist because, by Corollary 15.1.9, $\mathrm{KT}(e+1)$ is finite.

We similarly choose n_2 so that for any set $Y \in \mathrm{KT}(e+2) - \mathrm{KT}(e)$, there is an $m < n_2$ such that $K(Y \upharpoonright m) > K(m) + e$.

Now we come to the clever part of this argument. We choose n_3 so that for any set $Y \in \mathrm{KT}(e+3) - \mathrm{KT}(e)$, there is an $m < n_3$ such that $K(Y \upharpoonright m) > K(m) + e$, but we also impose an extra condition on n_3. Let Z be a set such that the least m with $K(Z \upharpoonright m) > K(m) + e$ is in $[n_1, n_2)$. Then Z cannot be in $\mathrm{KT}(e+1)$ by the choice of n_1, so we can require that n_3 be such that for each such Z there is an $l < n_3$ such that $K(Z \upharpoonright l) > K(l) + e + 1$.

It is important to consider more carefully why such an n_3 exists. Suppose it did not. Then for each l there would be a string σ of length l such that the least m with $K(\sigma \upharpoonright m) > K(m)+e$ is in $[n_1, n_2)$ and yet $K(\sigma) \leqslant K(l)+e+1$. Thus, by König's Lemma, there would be a set Z such that the least m with $K(Z \upharpoonright m) > K(m) + e$ is in $[n_1, n_2)$ and yet $K(Z \upharpoonright l) \leqslant K(l) + e + 1$ for all l. Such a Z would be in $\mathrm{KT}(e+1)$, contradicting the choice of n_1.

The definition of n_i for $i > 3$ is analogous. That is, we choose n_i so that

1. for any set $Y \in \mathrm{KT}(e+i) - \mathrm{KT}(e)$, there is an $m < n_i$ such that $K(Y \upharpoonright m) > K(m) + e$, and

2. for any Z such that the least m with $K(Z \upharpoonright m) > K(m) + e$ is in $[n_{i-2}, n_{i-1})$, there is an $l < n_i$ such that $K(Z \upharpoonright l) > K(l) + e + i - 2$.

The same argument as in the $i = 3$ case shows that such an n_i exists.

As mentioned above, we let $f_e(k) = e+i-1$ whenever $k \in [n_i, n_{|A'_{e+1}|i+1})$. Suppose that $K(X \upharpoonright n) \leqslant K(n) + f_e(n)$ for all n. We need to show that $X \in \mathrm{KT}(e)$. Suppose not, and let i be least such that there is an $m \in [n_i, n_{i+1})$ with $K(X \upharpoonright m) > K(m) + e$. Then, by the choice of n_{i+2}, there is an $l < n_{i+2}$ such that $K(X \upharpoonright l) > K(l) + e + i$. On the other hand, $K(X \upharpoonright l) \leqslant K(l) + f_e(l) \leqslant K(l) + e + i$, which is a contradiction. □

Corollary 15.9.2 (Csima and Montalbán [58])**.** *There is a minimal pair of K-degrees.*

Proof. Let f be as in Theorem 15.9.1. We build A and B by finite extensions. Let $A_0 = B_0 = \lambda$. At stage $e + 1$ with e even, first define $A'_{e+1} \succ A_e$ so that $K(A'_{e+1}) > K(|A'_{e+1}|) + e$. Let $m_e = |A'_{e+1}| - |A_e|$. Now let c_e be such that $A'_{e+1}0^\omega \in \mathrm{KT}(c_e)$, and let n_e be such that $f(n_e) > c_e$. Let $A_{e+1} = A'_{e+1}0^{n_e}$ and $B_{e+1} = B_e 0^{m_e+n_e}$.

At stage $e + 1$ with e odd, do the same with the roles of A and B interchanged.

It is straightforward to check that this construction ensures that A and B are not K-trivial, and that $\min(K(A \upharpoonright n), K(B \upharpoonright n)) \leqslant K(n) + f(n)$ for all n, which implies that if $X \leqslant_{\mathrm{K}} A, B$ then X is K-trivial. □

Analysis of the above proofs shows that f can be chosen to be Δ^0_4, an hence so can A and B. It is unknown whether there is a minimal pair of K-degrees of Δ^0_2 sets. More interestingly, it is also unknown whether there is a minimal pair of K-degrees of left-c.e. reals. The constrution of Theorem 13.10.7 gave a minimal pair of Σ^0_2 sets.

15.10 Presentations of K-trivial reals

Recall from Chapter 8 that a presentation of a left-c.e. real α is a c.e. prefix-free set $W \subset 2^{<\omega}$ such that $\alpha = \sum_{\sigma \in W} 2^{-|\sigma|}$. In Theorem 8.4.2 we saw that there are noncomputable left-c.e. reals α such that any presentation of α is computable. The following result of Stephan and Wu [293] shows that such α cannot be K-trivial. Its proof exploits the fact that, if α is K-trivial, then there are relatively few possibilities for each initial segment of α, and hence a left-c.e. approximation $\alpha = \lim_s \alpha_s$ must have an occasional relatively big jump $\alpha_{s+1} - \alpha_s$.

Theorem 15.10.1 (Stephan and Wu [293])**.** *Every noncomputable K-trivial left-c.e. real has a noncomputable presentation.*

Proof. Let α be a noncomputable K-trivial left-c.e. real. Then there is a constant c such that $K(\alpha \upharpoonright n) \leqslant \frac{n}{3} + c$ for all n. (Here, $\frac{n}{3}$ is used as an upper bound for $K(n)$, as it is enough for the purposes of this proof.) Furthermore, there is an approximation $\{\alpha_s\}_{s \in \mathbb{N}}$ to α such that each string α_s has length s and $K(\alpha_s \upharpoonright n) \leqslant \frac{n}{3} + c$ for $n \leqslant s$.

We must build a prefix free c.e. set $P \subset 2^{<\omega}$, presenting α, to satisfy the following requirement for each $e \in \mathbb{N}$.

$$\mathcal{R}_e : \ P \neq \overline{W_e},$$

where we think of W_e as the e-th prefix free c.e. subset of $2^{<\omega}$.

Construction. We enumerate a KC set L, and let P be the domain of the corresponding prefix-free machine. At stage $s+1$, let $r_s = \alpha_{s+1} - \alpha_s$, where we think of α_{s+1} and α_s as rationals. Thinking of r_s as a string, for each $d < |r_s|$ such that $r_s(d) = 1$, add $\langle d, \lambda \rangle$ to L.
End of Construction.

Verification. It is clear from the construction that $\sum_{\sigma \in P} 2^{-|\sigma|} = \sum_s r_s = \alpha$. We need to show that each $\mathcal{R}_e$ is met.

Suppose that $\mathcal{R}_e$ is not met, and let

$$l(e, s) = \max\{n : \forall \sigma \, (|\sigma| \leqslant n \to \sigma \in W_e[s] \cup P_s)\}.$$

Since $P = \overline{W_e}$, we have $\lim_s l(e, s) = \infty$.

Since α is not computable, there are infinitely many n such that $\alpha - \alpha_t > 2^{-n}$ for the least stage $t > 3n$ such that $l(e, t) > 3n$. Fix such an $n > c+2$. Since $K(\alpha_s \upharpoonright 3n) \leqslant n + c$ for all $s \geqslant t$, we have $|\{\alpha_s \upharpoonright 3n : s \geqslant t\}| \leqslant 2^{n+c}$. Thus there is a stage $s \geqslant t$ such that $r_s = \alpha_{s+1} - \alpha_s > 2^{-2n-c-2}$. At stage s, we issue a KC request $\langle d, \lambda \rangle$ for some $d \leqslant 3n$, which means that $P - P_s$ contains a string σ with $|\sigma| \leqslant 3n$. But $l(e, t) > 3n$, so $\sigma \in W_e[t] \cup P_t$. Since $\sigma \notin P_t$, we have $\sigma \in W_e \cap P$, and hence $\mathcal{R}_e$ is met. $\square$

16
Lowness and triviality for other randomness notions

16.1 Schnorr lowness

We now turn to lowness notions for other notions of randomness. We begin with Schnorr randomness. Since it there is no universal Schnorr test, it is not clear that the notions "Schnorr low" and "Low for Schnorr tests" will be the same. (In fact this had been a question of Ambos-Spies and Kučera [8] till it was recently solved by Kjos-Hanssen, Stephan, and Nies in [144].

16.1.1 Lowness for Schnorr null sets

In such cases it is usually easiest to begin with the test set version. Remarkably, there was a complete characterization of lowness for Schnorr null tests obtained by Terwijn and Zambella [304]. In the last section we met the notion of c. e. traceable.

Definition 16.1.1 (Terwijn and Zambella [304])**.** We say that a degree $\mathbf{a}$ is *computably traceable* iff there is a function h, such that for all $f \leqslant_T \mathbf{a}$, there is a *strong* array of (canonical) finite sets $\{D_{g(n)} : n \in \mathbb{N}\}$ such that for all n,

(i) $|D_{g(n)}| < h(n)$ and

(ii) $f(n) \in D_{g(n)}$.

We remark that in the literature, computable traces are often represented by a single computable set T where $T^{(n)} = \{\langle y, n\rangle : y \in D_{g(n)}\}$. We will use this method, and by using the notation where we use square brackets for the index to get rid of the $\langle \cdot, n\rangle$. That is

$$T^{[n]} = \{y : \langle y, n\rangle \in T^{(n)}\}.$$

This notion was inspired by arguments from set theory, particularly Raisonnier's [243] proof of Shelah's result that you cannot take the inaccessible out of Solovay's [283] construction of a model with every set of reals Lebesgue measurable.

In Chapter 5 we made an observation after the Miller-Martin 5.14.3 construction of a hyperimmune-free degree which was the following.

Observation 16.1.2 (Terwijn and Zambella [304]). *The hyperimmune-free degree of the Miller-Martin 5.14.3 construction is computably traceable.*

Actually, it is easy to see that being computably traceable is a *uniform* version of being hyperimmune-free.

Observation 16.1.3 (Terwijn and Zambella [304]). *If* $\mathbf{a}$ *is computably traceable, then* $\mathbf{a}$ *is hyperimmune-free.*

Proof. Suppose, as above that $\mathbf{a}$ is computably traceable, and $h(n)$ is the computable bound. If $f \leqslant_T \mathbf{a}$, then choose $g(n)$ as above, and define $k(n) = \max\{p : p \in D_{g(n)}\}$. Then $k(n)$ dominates $f(n)$, and hence every function computable in $\mathbf{a}$ is dominated by a computable one. This is the definition of being hyperimmune free. $\square$

The difference between being hyperimmune-free and being traceable is that for the latter there is a single computable bound h that works for *all* $f \leqslant_T \mathbf{a}$, whereas for the latter for each $f \leqslant_T \mathbf{a}$ there is a computable bound h_f.

This difference can be turned around into a proof that the concepts are different. We can derive this more easily via the hyperimmune-free basis theorem and the main result of this section.

Theorem 16.1.4 (Terwijn and Zambella [304]). *There are* ($2^{\aleph_0}$) *many degrees that are hyperimmune-free yet not computably traceable.*

Proof. Choose any Martin-Löf random A of hyperimmune free degree. This is certainly Schnorr random and, since we can A-computably construct a Schnorr null test $\{U_n : n \in \mathbb{N}\}$ covering A, there cannot be a Schnorr test containing $\{U_n : n \in \mathbb{N}\}$. Hence A cannot be computably traceable since then it would be Schnorr low by the Theorem 16.1.5 below. $\square$

Since all computably traceable degrees are hyperimmune-free, in particular, no computably traceable degree is Δ^0_2. That is, not Martin-Löf low set is computable traceable. This is particularly interesting in view of the following.

Theorem 16.1.5 (Terwijn and Zambella [304]). *A degree* $\mathbf{a}$ *is computably traceable iff* $\mathbf{a}$ *is low for Schnorr null tests.*

We first use the following easy lemma which says that any order function can be chosen for h, which we have seen is *not* true for the case of K-triviality by the Cholak, Downey, Greenberg Theorem **??**, at least for jump traceability.

Lemma 16.1.6 (Terwijn and Zambella [304]). *Let h be any computable nondecreasing function with $h(n) \to \infty$. Then if $\mathbf{a}$ is computably traceable, it is computable traceable with a bound growing slower than h.*

Proof. We show how to arbitrarily slow down the tracing process. Let $\widehat{h}$ be any computable trace for $\mathbf{a}$. Let $g \leqslant_T A$. Let q be an increasing computable function arbitrarily quickly. Let T be a computable trace with bound $\widehat{h}$ that captures the function $i \mapsto g \restriction q(i)$ (the string that codes the first $q(i)$ values of g). Let S be the set defined by

$$S^{[k]} = \{\sigma(k) : \sigma \in T^{[i_k]}\},$$

where i_k is least i such that $|\sigma| = q(i) > k$. Clearly, S is a computable trace. The cardinality of $S^{[k]}$ is bounded by $\widehat{h}(i_k)$. So, the faster q grows, the slower the cardinality of $S^{[k]}$ grows. It is easy to design an q that makes S attain a given computable bound h. $\square$

Proof. (of Theorem 16.1.5) This proof follows that of Zambella and Terwijn. For the "only if" direction, let A be a computably traceable. Let $\{U_n^A : n \in \mathbb{N}\}$ be an A-Schnorr test. We need a Schnorr test $\{V_n : n \in \mathbb{N}\}$ with $\cap_n V_n \supseteq \cap_n U_n^A$. The idea is to trace the finite A-approximations $U_{n,s}^A$ by some computable trace T, and use T to compute V. Thus $U_{n,s}^A \in T^{[\langle n,s\rangle]}$. The crucial idea of Terwijn and Zambella is to make sure that the bulk of $U_{n,s}^A$ is enumerated before we compute trace it, so the approximation generated by T is a good one. Thus we will speed up the A-enumeration of U so that

$$\mu(U_{n,s}^A) > 2^{-n}(1 - 2^{-s}).$$

We will also use Lemma 16.1.6 to make h grow slow enough so that the combinatorics work.

Next we define a new trace $\widehat{T}$ as follows. Let $\widehat{T}^{[\langle n,s\rangle]}$ be the set of those $D \in T^{[\langle n,s\rangle]}$ such that D is a finite subset[1] of $2^{<\omega}$ and

$$2^{-n} - 2^{-s} \leqslant \mu D \leqslant 2^{-n} \quad \text{and} \quad C \subseteq D \text{ for some } C \in \widehat{T}^{[\langle n,s-1\rangle]}$$

(for $s = 0$, $\widehat{T}^{[\langle n,s-1\rangle]}$ is defined to be empty). Here we have pruned those members of T which are not possible $U_{n,s}^A$. Observe that $\widehat{T}$ is still a computable trace that captures $U_{n,s}^A$. We are now ready to define the Schnorr test.

$$V_{n,r} = \bigcup_{s<r} \widehat{T}^{[\langle n,s\rangle]} \qquad \text{and} \qquad V_n = \bigcup_{r\in\omega} V_{n,r}.$$

[1] Remember, finite sets are identified with their codes.

We notice that $\mu(V_n) \leqslant 2^{-n}|\widehat{T}^{[\langle n,0\rangle]}| + \sum_{s\in\mathbb{N}} 2^{-s}2^{-n}|\widehat{T}^{[\langle n,s\rangle]}|$. The idea of Terwijn and Zambella is that, using Lemma 16.1.6, we can choose h so that $|\widehat{T}^{[\langle n,s\rangle]}|$ is small. Note that $\mu(\bigcup_{s>r}\widehat{T}^{[\langle n,s\rangle]}) < \sum_{s>r} 2^{-s}\cdot|\widehat{T}^{[\langle n,s\rangle]}|$, and hence, again using Lemma 16.1.6, if we choose h to grow sufficiently slowly, we can ensure that $\mu(V_{n,r})$ converges computably to $\mu(V_n)$.

The other direction is more difficult. For $l, k \in \mathbb{N}$, define the clopen set

$$B_{k,l} = \{[\tau\widehat{\ }1^k] : \tau \in 2^{<\omega} \wedge |\tau| = l\}.$$

Evidently $\mu(B_{k,l}) = 2^{-k}$. The Terwijn-Zambella idea is to use these clopen sets to code an arbitrary function $g \leqslant_T A$, where A is low for Schnorr tests.

To do this we let $U_n^g = \cup_{k>n} B_{k,g(k)}$. Clearly, $\{U_n^g : n \in \mathbb{N}\}$ is an A-Schnorr test, and hence since A is low for Schnorr tests, we can find a Schnorr test that contains $\cap_{n\in\mathbb{N}} U_n^g$.

• We[2] can find a computable open set V containing the intersection $\cap_{n\in\mathbb{N}} U_n^g$. We can choose the enumeration $V = \cup_s V_s$ to converge computably to V and also have $\mu(V) < \frac{1}{4}$. As observed by Terwijn and Zambella, it is certainly possible to make sure that for all l, and k, that

$$\mu(B_{k,l} - V) \neq 2^{-(l+3)},$$

by adding things to V to ensure this[3].

We now make the simplifying assumption that $\mu(U_n^g - V) = 0$ for some n; eliminating this assumption later. We are ready to define the trace for g. Let

$$T^{[k]} = \{l : \mu(B_{k,l} - V) < 2^{-(l+3)}\}.$$

First T traces g since $\mu(U_n^g - V) = 0$, except for possibly the first n values of g. The key is to show that T is computable. It is enough to show how to enumerate the complement of T, $\overline{T}$.

Let $s_0 = 0$ and define s_{i+1} and ε_i such that

$$\varepsilon_i = \mu(B_{k,l} - V_{s_i}) - 2^{-(l+3)} \qquad \text{and} \qquad \mu(V_{s_{i+1}}) > \mu(V) - \frac{\varepsilon_i}{2}.$$

Now suppose $l \notin T^{[k]}$. Then $\varepsilon_i > 0$ for all i. It is clear that ε_i converges to a limit ε and, by the assumption that $\mu(B_{k,l} - V) \neq 2^{-(l+3)}$, we have that $\varepsilon > 0$. So, $\varepsilon_i/2 < \varepsilon$ for some i. Therefore $\varepsilon_i/2 < \varepsilon_{i+1}$ for some i. So, enumerating V up to stage s_{i+1} we know for sure that $l \notin T^{[k]}$. Hence T is computable.

It remains to show that $|T^{[k]}|$ is computably bounded, to show that the degree is computably traceable. This is really the subtlest part of the proof, since *first* we show that $|T^{[k]}|$ has a computable bound (which shows that **a**

---

[2]Here, we will mark a spot in this proof to be used later.

[3]The idea is that if we see the difference between $B_{k,l}$ and V_s getting close then add to V enough of $B_{k,l}$ to ensure that equality will not happen, but still ensure that for all $\varepsilon > 0$, $\mu(V' - V) < \varepsilon$ for the "padded" version V' of V thus obtained, and $\mu(V')$ can still be computably approximated.

is hyperimmune-free) and then remove the dependence on the enumeration on V to get a *uniform* bound.

To prove that there is computable bound, it suffices to show that we can effectively find an l_k such that $l \notin T^{[k]}$ for all $l > l_k$. Find a stage s such that $\mu V_s > \mu V - 2^{-(k+2)}$. Let l_k be larger than k and larger than the length of all strings in V_s. From the definition of $B_{k,l}$ it is clear that V_s and $B_{k,l}$ are independent for every $l > l_k$. This implies immediately that $\mu(B_{k,l} - V_s) = 2^{-k}(1 - \mu V_s) > 2^{-k}(3/4)$. Consequently, we cannot have that $\mu(B_{k,l} - V) < 2^{-(k+2)}$ and a fortiori that $\mu(B_{k,l} - V) < 2^{-(l+3)}$.

Now we deal with the uniformity issue. The problem is that l_k depends on the computable enumeration of V and, indirectly, on g, so we still have to show that there is an uniform bound on $|T^{[k]}|$. We claim that $|T^{[k]}| < 2^k k$ for every k. This is a calculation. The definition of $T^{[k]}$ guarantees that $\sum_{l \in T^{[k]}} \mu(B_{k,l} - V) < \frac{1}{4}$. hence we see

$$\mu(\bigcup_{l \in T^{[k]}} B_{k,l}) - \mu(V) \;\leqslant\; \mu \bigcup_{l \in T^{[k]}} (B_{k,l} - V) \;\leqslant\; \frac{1}{4}.$$ From this

we conclude that $\mu \bigcup_{l \in T^{[k]}} B_{k,l} \leqslant \frac{1}{2}$. Now $\mu(B_{k,l}) = 2^{-k}$ by definition, and for a fixed k, the $B_{k,l}$'s are independent as soon as the l's are sufficiently far apart. Thus we obtain

$$1 - \left(1 - 2^{-k}\right)^{\frac{|T^{[k]}|}{k}} \;\leqslant\; \mu\left(2^\omega - \bigcap_{l \in T^{[k]}} (2^\omega - B_{k,l})\right) \;\leqslant\; \frac{1}{2}.$$

Finally this gives $|T^{[k]}| \leqslant 2^k k$, as required, which is independent of V and g.

The final part of the Terwijn-Zambella proof is some technical calculations showing that we can remove the hypothesis that $\mu(U_n^g - V) = 0$; weakening it to the hypothesis:

$\mu_\sigma(U_n^g - V) = 0$ for some σ and some n such that $\mu_\sigma(V) < \frac{1}{4}$,

where μ_σ denotes the measure conditioned to σ:

$$\mu_\sigma(U) = \frac{\mu(U \cap [\sigma])}{\mu([\sigma])}$$

First we show that the proof still works with this weaker hypothesis, and then later we prove that the weaker hypothesis is true.

Suppose that $\mu_\sigma(U_n^g - V) = 0$ and $\mu_\sigma(V) < \frac{1}{4}$. For a set of strings W we use the notation

$$W|\sigma = \{\tau \in 2^{<\omega} : [\sigma\widehat{\ }\tau] \subseteq W\}.$$

We may assume that $g(k) > k$ for every k because a trace for $g(k) + k$ immediately gives a trace for g. Clearly we can also assume that $n > |\sigma|$. We claim that $\mu(U_n^{\tilde{g}} - \tilde{V}) = 0$ where $\tilde{V} = V|\sigma$ and $\tilde{g}$ is the translation of g defined by $k \mapsto g(k) \dot{-} |\sigma|$. Namely, if $l > |\sigma|$ then $B_{k,l}|\sigma = B_{k,l-|\sigma|}$. Since $g(k) > k$ and $n > |\sigma|$ we have that $U_n^g|\sigma = U_n^{\tilde{g}}$, so $\mu(U_n^{\tilde{g}} - \tilde{V}) = \mu_\sigma(U_n^g - V) = 0$. This proves the claim. Now, it is clear that $\mu(\tilde{V}) < \frac{1}{4}$ has also a computably approximable measure. So the proof given above is valid when $\tilde{V}$ and $\tilde{g}$ are substituted for V and g and ensures the existence of a computable trace for $\tilde{g}$. But from a trace of $\tilde{g}$ we immediately obtain a trace for g.

Finally, suppose that no σ and n exist such that $\mu_\sigma(U_n^g - V) = 0$ and $\mu_\sigma(V) < \frac{1}{4}$. We shall obtain a contradiction by constructing a real in $\bigcap U^g - V$. Let σ_0 be the empty string and assume we have defined σ_n such that $\mu_{\sigma_n}(V) < \frac{1}{4}$. By hypothesis $\mu_{\sigma_n}(U_n^g - V) > 0$, so there is a $\tau \in U_n^g$ such that $\mu_{\sigma_n}([\tau] - V) > 0$. In particular $\tau \supseteq \sigma_n$ and $\mu_\tau V < 1$. Apply the Lebesgue density theorem to find $\sigma_{n+1} \supseteq \tau$ such that $\mu_{\sigma_{n+1}}(V) < \frac{1}{4}$. Let R be the real that extends all σ_n's constructed in this way. Since $[\sigma_{n+1}] \subseteq U_n^g$ for all n we have that $R \in \bigcap U^g$. But $[\sigma_n] \nsubseteq V$ for every n, so, since V is open, $R \notin V$. This contradiction completes the proof. $\square$

16.1.2 Lowness for Schnorr randomness

In this section, we will look at the apparently weaker notion, Schnorr lowness for randoms. It is clear that if A is low for tests then A is low for Schnorr randoms. But the converse is not at all clear and had been an open question of Ambos-Spies and Kučera [8]. The question was finally solved by Kjos-Hanssen, Stephan, and Nies in [144]. As we see they proved that the two notions coincide. On the way they also proved some intermediate results of independent interest.

Kjos-Hanssen modified the proof of Theorem 16.1.5 to prove the following

Theorem 16.1.7 (Kjos-Hanssen see [144])**.** *A set A is c.e. traceable iff every Schnorr null set relative to A is contained in a Martin-Löf null set.*

Proof. Suppose that every Schnorr null set relative to A is Martin-Löf null. Follow the proof of Theorem 16.1.5 *exactly* until you get to the $\bullet$.

Now instead of finding a *computable* set V containing $\cap_n U_n^g$, we have a *computably enumerable* V with exactly the same properties, having a computable enumeration $V_s \to V$, and $\mu(V) < \frac{1}{4}$. Now follow the proof of Theorem 16.1.5 exactly, the only thing to check is that the T we get is a computably enumerable trace. This is straightforward.

For the other direction, suppose that A is c.e. traceable. Let $\{U_n^A : n \in \mathbb{N}\}$ be an A-Schnorr test with $\mu(U_n^A) = 2^{-n}$. We can use Lemma **??** to choose the function h for the cardinality of the trace to be $h(i) = i$. Now we define a function $f \leqslant_T A$.

We will describe the building of the Martin-Löf test from the trace of f, and define f simultaneously, since it will motivate the construction. Let $f(1)$ be the code of $U_n^A[s]$ at the first stage s where $\mu(U_1^A[s])$ is within a fraction of $\frac{15}{16}$ of its final measure of $\frac{1}{2}$.

Note that since $T^{[1]}$ has only $h(1) = 1$ member, we know U_1^A to within $\frac{1}{32}$, that is, to within $\frac{15}{16}$ of its final measure.

We begin the definition of the Martin-Löf test by letting $V_1[1] = T^{[1]}$.

For the second step, we need to keep the exponential growth of the fraction of the measures enumerated to use the fact that h is slow growing to limit the size of possible errors.

Let $f(2)$ be the concatenation of $U_1^A[s] - U_1^A[s2]$ and $U_4^A[s2]$ at the first stage where we know $U_1^A \supseteq U_4^A[s2]$ and both to within a fraction of, say, $\frac{2^{k_2}-1}{2^{k_2}}$ of their respective final measures of $\frac{1}{2}$ and $\frac{1}{16}$ respectively. Here k_2 will be chosen sufficiently large.

Now we calculate $T^{[2]}$ tracing $f(2)$. This has at least one, and possibly two, elements. When we see an element of $T^{[2]}[t]$ we can see if its first coordinate is consistent with $V_1[1]$. If so, then that part we add to V_1, and similarly the second part we add to V_2. The total measure we add to V_1 through this action is at most twice $\frac{1}{16} - \frac{1}{2^{k_2}}$ of U_1^A's total measure of $\frac{1}{2}$, so below $\frac{1}{16}$. Similarly we begin to build V_2 using the second coordinate of those elements entering $T^{[2]}$. Again we can make at most two errors, and so this step would add into V_2 at most twice $\frac{2^{k_2}-1}{2^{k_2}}$ of the total size of U_4^A which is $\frac{1}{16}$, and hence total measure of $\frac{2^{k_2}-1}{2^{k_2}} \times \frac{1}{8}$.

But, for both of these sets there is at most $\frac{1}{2^k}$ of their measure left to be enumerated.

We can continue in this manner, targeting $U_{2^k}^A$ for V_3, and using $\frac{2^{k'}-1}{2^{k'}}$ of the measures to appear in U_1^A, U_4^A and $U_{2^k}^A$, etc. The errors this time would at most triple up.

Choosing the k's sufficient fast growing, we can bound the size of V_n by 2^{-n}.

They are clearly define a Martin-Löf test and this test covers the A-Schnorr test. $\square$

Lemma 16.1.8 (Stephan see [144])**.** *Suppose that A is hyperimmune-free and c.e. traceable. Then A is computably traceable.*

Proof. Let T be the c.e. trace of some function $g \leqslant_T A$. Then $g(n) \in T^{[n]}$, and $|T^{[n]}| < h(n)$ for a computable h. Let $f(n) = \mu s(g(n) \in T_s^{[n]})$. Since A is hyperimmune-free, f is majorized by some computable function c. Then $\tilde{T}$ is a computable trace for g where $\tilde{T}^{[n]} = T_{c(n)}^{[n]}$. $\square$

Therefore, we will be finished if we can prove that every low for Schnorr random A is hyperimmune free. Actually, this decisive final step was the first result proved about lowness for Schnorr randoms, and is due to Bedregal and Nies.

Theorem 16.1.9 (Bedregal and Nies [27])**.** *Suppose that A is either low for computably random reals or low for Schnorr random reals, then A is of hyperimmune-free degree.*

Proof. Suppose that A is hyperimmune. Then there is a function $g \leqslant_T A$ such that, for any computable function h, $\exists^\infty x(h(x) < g(x))$. The basic idea of Bedregal and Nies is to perform a construction of a Schnorr random real R, and, at the same time, build an A-computable martingale F which (strongly) succeeds on R. g is used to all us to argue the success. Hence we show that if A is hyperimmune, then A is not computably or Schnorr low for randoms.

We now turn to details. We follow Bedregal and Nies [27]. For this proof, we let $\alpha, \beta, \ldots$ denote finite subsets of $\mathbb{N}$, coded by $n_\alpha = \sum_{i \in \alpha} 2^i$.

Let M_e denote the e-th partial computable martingale, and we assume that the range is included in $[\frac{1}{2}, \infty)$. Of course, if R is not random then $M_e(R) \to \infty$ for some e.

Let T denote the collection of e with M_e total. Then we will have a collection, to be defined, of α with $e \in T \to e \in \alpha$, for which we will define strings σ_α such that $\alpha \subseteq \beta \to \sigma_\alpha \preccurlyeq \sigma_\beta$. The strings σ_α are chosen so that, for each e, $M_e(\sigma_\alpha)$ is bounded by a fixed constant $c = c(e)$. Then the real

$$R = \cup_{\alpha \in T} \sigma_\alpha,$$

is computably random.

However, we will construct an A-computable martingale F that strongly succeeds on R. The key idea of Bedregal and Nies is to give an inductive definition of the strings σ_α, what they call *scaling factors* p_α, and partial computable martingales M_α such that, if σ_α is defined, then

$M_\alpha(\sigma_\alpha) \downarrow < 2$ and converges in $\leqslant g((|\sigma_\alpha|)$ many steps.

A will be able to decide if $\tau = \sigma_\alpha$ given τ and σ_α. $\sigma_\emptyset = \lambda$. If $\alpha = \beta \cup \{e\}$ with $e > \max\{j : j \in \beta\}$. Suppose that $\bullet$ holds for β. Then we define

$$p_\alpha = \frac{1}{2} 2^{-|\sigma_\beta|}(2 - M_\beta(\sigma_\beta), \text{ and}$$

$$M_\alpha = M_\beta + p_\alpha M_e.$$

Since M_e is a martingale on its domain, $M_e(\nu) < 2^{|\nu|}$ by Kolmogorov's inequality, and hence $M_\alpha(\sigma_\beta) < 2$ if it is defined.

The construction is to define σ_α we look for a sufficiently long extension σ of σ_β, such that

(i) M_α does not increase from σ_β to σ, and

(ii) $M_\alpha(\sigma) \downarrow$ in $\leqslant g(|\sigma|)$ steps.

That is, for longer and longer $m > 4n_\alpha$, whilst no string has been designated σ_α of length below m, and if $M_(\nu) \downarrow$ in $\leqslant g(m)$ steps for all ν with $|\nu| = m$, then use the averaging property to find a string σ of length m which extends σ_β and such that M_α does not increase from σ_β to σ.

The verification is a few relatively easy lemmas.

Lemma 16.1.10. *If $\alpha \subset T$, then σ_α and p_α are defined.*

Proof. Suppose the lemma for β and let $\alpha = \beta \cup \{e\}$. Consider the function

$$f(m) = \mu s \forall e \in \alpha \forall \nu \in 2^{\leqslant m}(M_\alpha(\nu)^s \downarrow).$$

Then f is a computable function, and hence there will be a infinitely many m with $g(m) > f(m)$, so that we will eventually define σ_α and p_α. $\square$

Lemma 16.1.11. *R is computably random.*

Proof. If M_e is total, then let $\alpha = T \cap [0, e]$. If $\alpha \subseteq \beta$ and $\beta' = \beta \cup \{i\}$, and $\beta' \subset T$, then for each σ, if $\sigma_\beta \prec \sigma \prec \sigma_{\beta'}$,

$$p_\alpha M_e(\sigma) \leqslant M_\beta(\sigma) \leqslant M_\sigma(\sigma_\beta) < 2.$$

Hence $M_e(\sigma) < \frac{2}{p_\alpha}$ for each $\sigma \prec R$. $\square$

Lemma 16.1.12. *There is a martingale $F \leqslant_T A$ that which strongly succeeds on A. Indeed,*

$$\exists^\infty \sigma \prec R(F(\sigma) > \lfloor \frac{|\sigma|}{4} \rfloor.$$

Proof. Let $r(\sigma) = \lfloor \frac{|\sigma|}{2} \rfloor$. Let $F = \sum_\alpha F_\alpha$, where F_α is a martingale with initial capital $F_\alpha(\lambda) = 2^{-n_\alpha}$, and which bets everything along σ_α from $\sigma_\alpha \upharpoonright r(\sigma_\alpha)$ onwards. That is, if σ_α is undefined then F_α is a constant with value 2^{-n_α}. Otherwise we let $\sigma = \sigma_\alpha \upharpoonright r(\sigma_\alpha)$, and

(i) Define $F_\alpha(\tau) = 2^{-n_\alpha}$ unless $\tau \upharpoonright r(\sigma_{\preccurlyeq}\tau$, and

(ii) If σ and τ are incompatible, then define $F_\alpha(\tau) = 0$, and

(iii) Otherwise, define $F_\alpha(\tau) = 2^{-n_\alpha} 2^{\min\{|\tau| - r(\sigma), r(\sigma)\}}$.

Then we see that

$$F_\alpha(\sigma_\alpha) = r(\sigma_\alpha) - n_\alpha.$$

Since $r(\sigma_\alpha) \geqslant 2n_\alpha$, this means that $F_\alpha(\sigma_\alpha) \geqslant \frac{|\sigma_\alpha|}{4} \rfloor$.

To complete the proof, we check that $F \leqslant_T A$. On input τ we determine $F_\alpha(\tau)$ for each α with $n_\alpha \leqslant |\tau|$. Using the function $G \leqslant_T A$, we determine if some string σ with $|\sigma| \leqslant 2|\tau|$ is equal to σ_α. If not then we will have $F_\alpha(\tau) = 2^{-n_\alpha}$. If yes, then we determine the value of F_α from the definition of σ_α, and F_α. $\square$

$\square$

16.2 Lowness for computable machines

In Chapter 15, we saw the the coincidence of a number of natural "anti-randomness" classes associated with prefix-free Kolmogorov complexity. That is, we have seen that for every real A, the following are equivalent.

(i) A is low for K.

(ii) A is K-trivial.

(iii) A is low for Martin-Löf randomness.

Relative to Schnorr randomness we have seen that lowness for Schnorr tests is the same as lowness for Schnorr randomness and this coincides with computable traceability. We might wonder whether there are analogs of (i) and (ii) above which hold for Schnorr randomness. To achieve such analogs we would need some analog of the characterization of Martin-Löf randomness in terms of prefix-free complexity. In Chapter 10 we met such a characterization by Downey and Griffiths [72]. Recall that we defined a prefix-free Turing machine M to be *computable* if the domain of M has computable measure, that is, $\sum_{\{\sigma : M(\sigma)\downarrow\}} 2^{-|\sigma|}$ is a computable real. We recall the following result:

Theorem 16.2.1 (Downey and Griffiths [72])**.** *R is Schnorr random iff for all computable machines M, for all n, $K_M(R \upharpoonright n) \geqslant n - O(1)$.*

In this section we will loog at the analog for lowness for K. Armed with the machine characterization of Schnorr randomness, we give the following definition.

Definition 16.2.2 (Downey, Greenberg, Mihailović, and Nies [**?**])**.** A real A is *low for computable machines* iff for all A-computable machines M there is a computable machine N such that for all x,

$$K_M^A(x) \geqslant K_N(x) - O(1).$$

The reader might be concerned about whether for an A-computable machine M^A as in the definition above, M^B is B-computable for other oracles B. However, given a such a machine, we can obtain another oracle machine $\widetilde{M}$ such that $M^A = \widetilde{M}^A$, and such that $\widetilde{M}^B$ is prefix-free and B-computable for every oracle B. [4]

A relativized version of the Kraft-Chaitin Theorem can be used to show that Theorem 16.2.1 relativizes. Namely, we have that R is A-Schnorr random iff for all A-computable machines M, for all n, $K_M^A(R \upharpoonright n) \geqslant n - O(1)$. Therefore, every real A that is low for computable machines is low for Schnorr randomness, and by the results of the previous section, it follows that A is low for Schnorr tests and thus is computably traceable. The machine lowness notion coincides with the lowness for randomness notion.

[4]Indeed, define the machine $\widetilde{M}$ as follows. First, we may assume that for every oracle B, M^B is prefix-free. Now let F be a computable functional such $F(A)$ is total and the measure of the set $\{x \leqslant F(A,n) \colon M^A(x)$ is defined after $F(A,n)$ steps$\}$ approximates $\mu(M^A)$ to within 2^{-n}. Define $\widetilde{M}^B$ inductively: at stage n, first wait for $F(B,n)$ to halt (in the meantime, no new $\widetilde{M}^B$-computations are recognised.) Next, allow M^B to run for $F(B,n)$ many steps and accept new computations as $\widetilde{M}^B$-computations; if at a later stage we see that $\mu(M^B) > \mu(M^B)[F(B,n)] + 2^{-n}$ then we stop accepting new $\widetilde{M}^B$-computations altogether. Then move to stage $n+1$. Note that the construction is uniform in M, F but not in M alone.

Theorem 16.2.3 (Downey, Greenberg, Mihailović, and Nies [**?**])**.** *A real A is low for computable machines iff A is computably traceable.*

Proof. We note that if we enumerate a Kraft-Chaitin set with a computable sum then the machine produced is computable: (See the proof of Theorem 6.6.1.)

Lemma 16.2.4. (Kraft-Chaitin) *Let $\langle d_0, \tau_0\rangle, \langle d_1, \tau_1\rangle, \ldots$ be a computable list of pairs consisting of a natural number and a string. Suppose that $\sum_{i<\omega} 2^{-d_i}$ is a computable real (in particular, is finite). Then there is a* computable *machine N such that for all i, $K_N(\tau_i) \leqslant d_i + O(1)$.*

To prove Theorem **??** we need to show that every computably traceable set A is low for computable machines. So let A be a computably traceable set and let M be an oracle machine such that M^A is A-computable. The idea is to "break up" the machine M^A into small and finite pieces which we trace. We view M^A as a function from strings to strings. We will partition M^A into finite pieces $g, f_0, f_1, f_2, \ldots$ where for $n < \frown$, the measure of the domain of f_n is smaller than some small rational ε_n. We then trace the sequence $\langle f_n\rangle$; so for every n, we get $h(n)$ many candidates for f_n, each with domain with measure smaller than ε_n. If we keep $\sum_n h(n)\varepsilon_n$ finite, the union of all of the candidates can be translated into a Kraft-Chaitin set that produces the machine we want.

Let h be the computable order function (again we recall Lemma 16.1.6 which says that we can pick any reasonable function; it doesn't matter for this proof.) Fix a computable, decreasing sequence of positive rationals $\varepsilon_0, \varepsilon_1, \ldots$ such that $\sum_{n<\frown} h(n)\varepsilon_n$ is finite; moreover, we want the convergence to be quick, say for every $m < \frown$,

$$\sum_{n \geqslant m} h(n)\varepsilon_n \ < \ 2^{-m}.$$

Let $\langle(\sigma_i, \tau_i)\rangle_{i<\frown}$ be an A-computable enumeration of M^A. We let M^A_s, the machine M^A at stage s, be $\{(\sigma_i, \tau_i) \,:\, i < s\}$, and similarly let $M^A_{\geqslant s} = M^A \setminus M^A_s = \{(\sigma_i, \tau_i) \,:\, i \geqslant s\}$, and for $s < t$, $M^A_{[s,t)} = M^A_t \setminus M^A_s$.

Let t_n be the least stage t such that $\mu(\mathrm{dom} M^A_{\geqslant t}) < \varepsilon_n$. We let $g = M^A_{t_0}$; for $n < \frown$, we let $f_n = M^A_{[t_n, t_{n+1})}$. The point is that the sequence $\langle t_n\rangle$, and so the sequence $\langle f_n\rangle$, are A-computable, as $\mu(\mathrm{dom} M^A_{\geqslant t}) = \mu(\mathrm{dom} M^A) - \mu(\mathrm{dom} M^A_t)$; the first number is A-computable by assumption, and the latter a rational, computable from the sequence $\langle(\sigma_i, \tau_i)\rangle$ and so from A. For all $n < \frown$, $\mu(\mathrm{dom} f_n) < \varepsilon_n$.

Each f_n is a finite function (and so has a natural number code.) We can thus computably trace the sequence $\langle f_n\rangle$; there is a computable sequence of finite sets $\langle X_n\rangle_{n<\frown}$ (i.e. $X_n = D_{r(n)}$ where r is computable) such that for each n, $|X_n| \leqslant h(n)$, and for each n, (the code for) $f_n \in X_n$. By weeding out elements, we may assume that for each $n < \frown$, every element of X_n

is a code for a finite function f from strings to strings whose domain is prefix-free and has measure at most ε_n.

Enumerate a Kraft-Chaitin set L as follows. Let $\langle d, \tau\rangle \in L$ if there is some σ such that $|\sigma| = d$, and one of the following holds:

- $(\sigma, \tau) \in g$;
- For some n and for some $f \in X_n$, $(\sigma, \tau) \in f$.

The set L is computably enumerable. Further, the total of the requests $s = \sum_{(d,\tau)\in L} 2^{-d}$ is a finite, computable real, as we know that for any m,

$$\sum \{2^{-|\sigma|} \colon (\exists n \geqslant m)(\exists f \in X_n)[\sigma \in \mathrm{dom} f]\} \ \leqslant \ \sum_{n \geqslant m} h(n)\varepsilon_n \ \leqslant \ 2^{-m}.$$

From the "computable" Kraft-Chaitin theorem we get a computable machine N such that for some constant c, if $(d, \tau) \in L$, then $K_N(\tau) \leqslant d+c$. On the other hand, we know that if τ is in the range of M^A then $(K_M^A(\tau), \tau) \in L$ because $f_n \in X_n$ for all n. Thus N is as required. □

16.3 Schnorr trivials

In Chapter 10, we defined a notion of Schnorr reducibility where $\alpha \leqslant_{Sch} \beta$ iff for all computable machines M, there is a computable machine $\widehat{M}$ such that $K_M(\beta \upharpoonright n) - \mathcal{O}(1) > K_{\widehat{M}}(\alpha \upharpoonright n)$, for all n. Naturally, this will give rise to a notion of triviality.

Definition 16.3.1 (Downey and Griffiths [72])**.** We say that a real α is *Schnorr trivial* iff $\alpha \leqslant_{Sch} 0^\omega$.

The first construction of a Schnorr trivail set awas by Downey and Griffiths [72]. As we will see, Schnorr trivials behave quite differently than both K-trivials and Schnorr lows, although one implication does hold.

Theorem 16.3.2 (Franklin [**?**])**.** *Suppose that A is Schnorr low. Then A is Schnorr trivial.*

Proof. (Downey, Greenberg, Mihailović, Nies [**?**]) Suppose that A is Schnorr low. Then A is low for computable machines by Theorem 16.2.3. Now let let N be a computable machine. Let L be an A-computable machine such that for all n, $K_L^A(A \upharpoonright n) = K_N(n)$ (for all x, if $N(x) = n$ then let $L(x) = A \upharpoonright n$.) Then by lowness, there is some computable machine M such that for all x, $K_M(x) \leqslant K_L^A(x) + O(1)$; M is as required to witness that A is trivial. □

Downey, Griffiths and LaForte [74], and Johanna Franklin [**?**] began systematic investigations of the concept of Schnorr triviality. Downey, Griffiths and LaForte proved the following result, generalizing the Downey-Griffiths result and clearly highlighting the differences.

Theorem 16.3.3 (Downey, Griffiths, and Laforte [74])**.** *There is a c.e. Turing complete Schnorr trivial real.*

Proof. As pointed out in Observation 10.3.8, any computable machine M is equivalent to some machine M′ such that $\mu(\mathrm{M}') = 1$. This fact helps to simplify the proof. We call a computable machine M total if $\mu(\mathrm{M}) = 1$ and $\{ 1^n : n \in \omega \} \subset \mathrm{ran}(\mathrm{M})$. It is not hard to approximate whether or not a machine is total in a Π^0_2 manner. To prove the result, we build a c.e. set A and function $g \leqslant_T A$ satisfying the following two sequences of requirements

$$\mathrm{R}_e : \mathrm{M}_e \text{ is a total machine} \Longrightarrow \exists \text{ computable } \mathrm{M}'_e\, \exists c\, \forall n\, K_{\mathrm{M}'_e}(A \upharpoonright n) \leqslant K_{\mathrm{M}}(1^n) + c, \text{ and}$$

$\mathrm{K}_i : i \in K \Longrightarrow g(i) \in A$,
where $\langle \mathrm{M}_e : e \in \mathbb{N} \rangle$ is a computable enumeration of all Turing machines with $\mu(\mathrm{M}_e) \leqslant 1$. This clearly suffices to establish the result.

The requirement R_e is essentially negative, since the main problem faced in ensuring it is to control the growth of $\mu(M'_e)$. The main conflict involved in the construction is that arising between a negative requirement R_e and the infinitely many coding markers $g(i)$ used by the K_i for $i \geqslant e$. The idea is to progressively move each such $g(i)$ to a number large enough to guarantee that $\sum_{j \geqslant i} K_{M_e}(1^{g(j)})$ is so small that the total measure that must be added to M'_e for the sake of keeping track of the different membership possibilities for all the $g(j)$ is less than 2^{-i}. What makes this possible is that, if M_e is a computable machine, one can wait for a stage such that $1 - \mu(M_e)$ is very small, so that one has a very tight estimate on $\sum \{ K_{M_e}(1^k) : M_e \text{ has not yet produced the string } 1^k \}$. At such a point, it is easy to move $g(j)$ for $j \geqslant i$ to these large numbers and ensure thereby that for all $j \geqslant i$ with $g(j)$ so defined, $\sum_{j \geqslant i} K_{M_e} 1^{g(j)}$ is small enough to allow future changes in $\mu(\mathrm{M}')$ to be computably bounded.

Construction A number is *fresh at stage s* if it is larger than the length of any string in the range of any Turing machine at stage s. It is also helpful to normalize each machine M so that if $1^n \in \mathrm{ran}(M)[s]$, then for all $k < n$, $1^k \in \mathrm{ran}(M)[s]$. Since we are only interested in total machines, this makes no difference to the satisfaction of any requirement. We use the priority tree $2^{<\omega}$ to control the construction.

Stage 0: $A[0] = \emptyset$, for all k, $g(k)[0] = k$, and all other functionals are undefined.

Stage $s{+}1$. We define an *approximation to the true path*, $f[s]$ of length $s{+}1$ consisting of the nodes accessible at stage $s{+}1$. For each $e < s$ we perform the following pair of actions:

First, to satisfy K_e, if $e \in K[s{+}1] - K[s]$, we enumerate $g(e)[s] \in A$.

Next, we allow $\alpha = f \upharpoonright e[s]$ to act if necessary as follows: If $s{+}1$ is the first stage since α was last initialized, we declare all $m \leqslant s$ to be stable for α, let $j^{\alpha}[s{+}1] = 1$, and immediately end stage $s{+}1$. Let k be the least number that is not stable for α at s. If

1. $0 \leqslant 1 - \mu(M_e)[s] < 2^{-2j^{\alpha}[s]-3}$,
2. for all β such that $\beta 0 \subseteq \alpha$, k is stable for β at s, and
3. $1^{g(k)[s]} \in \mathrm{ran}(M_e)$,

then enumerate $g(k)[s] \in A[s{+}1]$ and choose, for all $m \geqslant k$, fresh numbers $g(m)[s{+}1]$ in increasing order. Declare k *stable* for α, set $j^{\alpha}[s{+}1] = j^{\alpha}[s]{+}1$, sand let $f(n)[s] = 0$, so that $\alpha 0$ is accessible at stage $s{+}1$. Otherwise, take no action for α, and let $f(n)[s] = 1$, so that $\alpha 1$ is accessible at $s{+}1$.

Verification:

First, note that for all k, $g(k)[s]$ is only moved finitely often. There are only a finite number of $\alpha \in 2^{<\omega}$ that are accessible before stage k, and only such α will ever move k. For each such α, there is at most one stage s at which k is the least number that is not yet stable for α and at which $g(k)[s]$ is enumerated into $A[s{+}1]$ by the α-strategy. Hence each such α moves $g(k)[s]$ only finitely often, and so $g(k)[s]$ eventually stops moving. Since $g(k)[s{+}1] \neq g(k)[s]$ implies $g(k)[s] \in A[s{+}1]$, the action taken at the beginning of each stage guarantees that $K \leqslant_T A$.

Let $f = \liminf_{s\to\infty} f[s]$. Fix e, let $\alpha = f \upharpoonright e$, and let s_0 be the least stage at which α is accessible and is never again initialized . If $\beta 0 \subseteq \alpha$, then eventually each number $k > s_0$ must become stable for β. Also, eventually, each $g(k)[s]$ never changes value. Hence, :if $\alpha 1 \subset f$, then either $\mu(M_e) \neq 1$ or $\left\{ 1^n \ : \ n \in \omega \right\} \not\subseteq \mathrm{ran}(M_e)$, so that the requirement is immediately satisfied. So, we may assume $\alpha 0 \subset f$. So, for every $j > 0$ there is a stage $s(j)$ such that $j^{\alpha}[s] = j$ and $j^{\alpha}[s{+}1] = j + 1$. At each stage $s(j)$,

(a.) $1 - \mu(M_e)[s(j)] < 2^{-2j-3}$,

(b.) $g(s_0 + j)[s(j)] \in A[s(j) + 1]$,

(c.) for all $m < g(s_0 + j)[s]$, $1^m \in \mathrm{ran}(M_e[s])$,

(d.) for all $k \geqslant s_0 + j$, $1^{g(k)[s(j)+1]} \notin \mathrm{ran}(M_e[s])$, and

(e.) for all $k \geqslant s_0 + j$ and for all $s > s(j)$, $g(k)[s] \neq g(k)[s(j)]$.

For each $m \in \mathrm{ran}(M_e)[s]$, let $\sigma_m[s]$ be the shortest, lexicographically least string such that $M_e(\sigma_n)[s] = 1^m$. Clearly, $\lim_{s\to\infty} |\sigma_m[s]| = K_{M_e}(1^n)$. We will use the Kraft-Chaitin theorem to construct a computable machine M' satisfying the requirement, by enumerating pairs $\langle s, \sigma \rangle$ into a c.e. set R. $g(s_0{+}1)[s(1){+}1]$ is the least number that we have to worry about. So, let $\tau^- = A \upharpoonright (g(s_0{+}1)[s(1){+}1])$, and, for each $m \leqslant |\tau^-]$, enumerate $\langle K_{M_e}(1^m)+2, A \upharpoonright m\rangle$ into $R[s(1)]$. Notice that this adds at most 2^-2 to the measure of M', since $\mu(M_e) = 1$. Let $m_1 = |\tau^-|$, and for each $j > 1$, let $m_j = \max\{\, m \,:\, 1^m \in \operatorname{ran}(M[s(j)])\,\}$. Notice that $g(s_0{+}j)[s(j)] \leqslant m_j < g(s_0{+}j)[s(j){+}1]$. For each bit string σ with $|\sigma| < j$, let $\tau_\sigma[s(j){+}1]$ be defined by $\tau_\sigma(g(s_0+k))[s{+}1] = \sigma(k)$ for all $k < j$ and $\tau_\sigma(m)[s(j){+}1] = A(m)$ for all other $m \leqslant m_j$. For each $m \leqslant m_j$, if $\langle|\sigma_m[s(j)]|+3, \tau_\sigma[s(j){+}1] \upharpoonright m\rangle \notin R[s(j)]$, then enumerate $\langle|\sigma_m[s(j)]|+3, \tau_\sigma[s(j){+}1] \upharpoonright m\rangle$ into $R[s(j){+}1]$. Note that for each $m \leqslant m_j$, $\langle|\sigma_m[s(j)]|+3, A[s(j){+}1] \upharpoonright m\rangle \in R[s(j){+}1]$. Since $\lim_{j\to\infty} m_j = \infty$, this shows that for all m, $K_{M'}(A \upharpoonright m) \leqslant K_{M_e}(1^m)+3$. If $m \leqslant m_{j-1}$, then $\langle|\sigma_m[s(j)]|+3, \tau_\sigma[s(j){+}1] \upharpoonright m\rangle$ is enumerated into $R[s(j){+}1]$ only if $\sigma_m[s(j-1)] \neq \sigma_m[s(j)]$. If $m_{j-1} < m \leqslant m_j$, then $1^m \notin \operatorname{ran}(M[s(j-1)])$. Hence, if

$$S = \{\, m \,:\, \langle|\sigma_m[s(j)]|+3, \tau_\sigma[s(j){+}1] \upharpoonright m\rangle \text{ is enumerated into } R[s(j){+}1]\,\},$$

then $\sum_{m\in S} 2^{-|\sigma_m|-3} < 2^{-2(j-1)-6}$. Since there are only 2^j bit strings of length j, this means that the measure of the machine M' is increased by at most at most $2^j \cdot 2^{-2(j-1)-6} = 2^{-j-4}$ at stage $s(j){+}1$. This shows M' is computable. So M' satisfies the requirement. □

16.3.1 More on the degrees of Schnorr trivials

Because both complete Schnorr trivial reals exist, and computable Schnorr trivial reals exist, one might wonder whether every c.e. degree contains a Schnorr trivial real. The following theorem yields a negative answer to this question.

Theorem 16.3.4. *There exists a c.e. set A such that for all sets B if $B \equiv_T A$, then there exists a computable machine M' such that for all c.e. machines M and numbers c there exists an n such that $K_M(B \upharpoonright n) > K_{M'}(1^n) + c$.*

Proof. We must build a c.e. set satisfying the following sequence of requirements:

$$\mathrm{R}_{\Phi,\Psi} : \ \Psi(\Phi(A)) = A \Longrightarrow$$
$$\exists \text{ computable } \mathrm{M}' \, \forall \mathrm{M}, c \, \exists n \, K_{\mathrm{M}}(\Phi(A) \upharpoonright n) > K_{\mathrm{M}'}(1^n) + c.$$

The strategy for a requirement $\mathrm{R}_{\Phi,\Psi}$ is composed of an infinite sequence of strategies for subrequirements $\mathrm{S}_{\Phi,\Psi,i} : \ K_{\mathrm{M}_i}(\Phi(A) \upharpoonright n) > K_{\mathrm{M}'}(1^n) + i$, that are only allowed to act on a sequence of stages at which $\Psi(\Phi(A)) = A$ appears more and more likely to be the case. Each such substrategy has a large number m associated to it and picks a sequence of witnesses $x_1, \ldots, x_m \notin A$ such that for every $1 \leqslant i < m$, $\psi(\Phi(A); x_i) < x_{i+1}$. Once these witnesses have been chosen, we enumerate the pair $\langle \log m{-}i{-}1, 1^{\psi(\Phi(A);x_m)}\rangle$ into a c.e. set defining M'. If there is no stage s and string σ such that $|\sigma| \leqslant \log m{+}i$ and $\mathrm{M}_i(\sigma) = \Phi(A) \restriction \psi(\Phi(A); x_m)$, then there is no need to ever take further action. At any stage s where there is a string σ such that $|\sigma| \leqslant \log m{+}i$ and $\mathrm{M}_i(\sigma) = \Phi(A) \restriction \psi(\Phi(A); x_m)$, we enumerate the greatest $x_j \notin A[s]$ into $A[s{+}1]$. If $\Psi(\Phi(A)) = A$, $\Phi(A)$ must change on $\psi(\Phi(A); x_j)$, so that M_i will be forced to converge on at least $m{+}1$ different strings of length less than or equal to $\log m{-}i$, thereby adding $(m{+}1) \cdot 2^{-\log m - i} > 2^{i+1} \geqslant 2$ to the measure of M_i. By Kraft's inequality, $\mu(M_i) \leqslant 1$, so this is not a possibility.

The priority organization of the requirements involves interleaving the subrequirements needed for strategies of type R, a task that is straightforward, although a little involved.

Construction

We use the tree of strategies $2^{<\omega}$ to control the construction, and adopt the convention that all uses with c.e. oracles are nondecreasing in the stage and increasing in the argument. The priority arrangement of the requirements is accomplished by a list function L, defined recursively on the nodes in $2^{<\omega}$ and the natural numbers. For all $n \in \omega$, $L(\lambda, n) = \mathrm{R}_{\Phi,\Psi}$ where $n\langle\Phi, \Psi\rangle$ under some standard enumeration of pairs of computable functionals. For any $\sigma \in 2^{<\omega}$, if $L(\sigma, 0) = R_{\Phi,\Psi}$ for some Φ and Ψ, then for every $n \in \omega$, $L(\sigma^\frown 1, n) = L(\sigma, n{+}1)$, $L(\sigma^\frown 0, 2n) = L(\sigma, n{+}1)$, and $L(\sigma^\frown 0, 2n{+}1) = S_{\Phi,\Psi,n}$. Otherwise, $L(\sigma^\frown 0, n) = L(\sigma^\frown 1, n) = L(\sigma, n{+}1)$. For each $\sigma \in 2^{<\omega}$, σ has requirement $L(\sigma, 0)$ assigned to it.

A node is initialized by having all its associated parameters undefined and associated sets set to $\emptyset$. A node α with a requirement $\mathrm{R}_{\Phi,\Psi}$ assigned to it has a machine M^α assigned to it that is built by enumerating pairs $\langle k, \tau\rangle$ into a c.e. set W^α. By the Kraft-Chaitin theorem, if $\sum\limits_{\langle k,\tau\rangle \in W^\alpha} 2^{-k} \leqslant 1$, this defines a prefix-free machine M^α such that for every $\langle k, \tau\rangle \in W^\alpha$, there is a string σ with $|\sigma| = k$ such that $M^\alpha(\sigma)\downarrow = \tau$. A node α with a requirement $\mathrm{S}_{\Phi,\Psi,i}$ assigned to it has parameter for a *starting number* $s^\alpha[s]$, and a sequence of witness parameters $x(\alpha, 1)[s], \ldots, x(\alpha, 2^{s^\alpha+i+1})[s]$. The construction of A and the necessary machines proceeds in stages.
Stage 0: We initialize all nodes in $2^{<\omega}$.
Stage $s{+}1$: We define an *approximation to the true path*, $f[s]$, of length at most s and allow each node $\alpha \subset f[s]$ to act. If $\alpha \subset f[s]$, then we call s an *α-stage*. Let $n = |\alpha|$. Let s^- be the most recent stage at which $\alpha \subset f[s^-]$, or the most recent stage at which α was initialized, whichever is greater.

Suppose α has requirement $\mathrm{R}_{\Phi,\Psi}$. In this case, we define the length-of-agreement function
$l^\alpha[s] = \max\{\, y : \forall x < y\, (\Psi(\Phi(A); x) = A(x))[s] \,\}$.

Let s_0 be the stage at which α was last initialized. A stage s is α-expansionary if $l^\alpha[s] > \max\{\, l^\alpha[t] : s_0 < t < s \text{ and } t \text{ is an } \alpha \text{ stage} \,\}$.

If s is not α-expansionary, then initialize all nodes β such that $\alpha^\frown 1 <_L \beta$ and let $f(n)[s] = 1$, so that $\alpha^\frown 1$ is accessible at stage $s{+}1$. If s is α-expansionary, then we let $\alpha 0$ be accessible at stage $s{+}1$ and initialize all nodes β such that $\alpha^\frown 0 <_L \beta$.

Suppose α has requirement $\mathrm{S}_{\Phi,\Psi,i}$ assigned to it. If there exists a node $\alpha' \subset \alpha$ such that α' has requirement $\mathrm{S}_{\Phi',\Psi',i'}$ for some Φ', Ψ', and i', and β is the longest node such that $\beta^\frown 0 \subset \alpha'$ and β has requirement $\mathrm{R}_{\varphi',\Psi'}$ assigned to it, and $(\langle s^\alpha, 1^{\psi(\Phi(A);x(\alpha',2^{s^{\alpha'}+i+1})}\rangle \notin W^\beta)[s]$ then immediately end stage $s{+}1$ and initialize all $\gamma \geqslant \alpha$.

Otherwise there are several cases to consider. Let β be the longest node with requirement $\mathrm{R}_{\Phi,\Psi}$ assigned to it such that $\beta^\frown 0 \subseteq \alpha$. If $x(\alpha,1)\!\uparrow\![s]$, then let $s^\alpha[s] = x(\alpha,1)[s] = s$. If there exists some least $j \leqslant 2^{s^\alpha[s]+i+1}$ such that $x(\alpha,j)\!\uparrow\![s]$ and $(x(\alpha,j{-}1) < l^\beta)[s]$, then let

$$x(\alpha,j)[s] = \max\big\{\, \varphi(A;y)[s] \,:\, y < \psi(\Phi(A);x(\alpha,j{-}1))[s] \,\big\}{+}1.$$

Immediately end stage $s{+}1$ and initialize all $\gamma \geqslant \alpha$.

Suppose $x(\alpha,j)\!\downarrow\![s]$ for all $j \leqslant 2^{s^\alpha+i+1}$. If $l^\beta[s] > x(\alpha,2^{s^\alpha+i+1})$ but $l^\beta[s^-] \leqslant x(\alpha,2^{s^\alpha+i+1})$, then enumerate $\langle s^\alpha[s], 1^{\psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1}))[s]}\rangle$ into $W^\beta[s{+}1]$. Immediately end stage $s{+}1$ and initialize all $\gamma \geqslant \alpha$. If there exists σ such that $\big(\mathrm{M}_i(\sigma) = \Phi(A) \restriction \psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1}))\big)[s]$ and $|\sigma| < s^\alpha[s]{+}i$, and j is greatest such that $(x(\alpha,j) \notin A)[s]$, then let $(x(\alpha,j) \in A)[s{+}1]$. Immediately end stage $s{+}1$ and initialize all $\gamma \geqslant \alpha$.
This completes the construction.

Verification

Let the *true path* f be $\liminf_{s\to\infty} f[s]$. Each $\alpha \subset f$ has some stage after which $\alpha \leqslant f[s]$ for every subsequent s. Once a node chooses a sequence of witnesses and is never again initialized, it only acts to change A or initialize other nodes a finite number of times. It follows, therefore, by a straightforward induction, that every $\alpha \subset f$ is initialized only finitely often.

Lemma 16.3.5. *Suppose $\alpha \subset f$ and there exist Φ, Psi, and i such that α has requirement $S_{\Phi,\Psi,i}$ assigned to it, and β is the longest node such that $\beta^\frown 0 \subset \alpha'$ and β has requirement $R_{\varphi',\Psi'}$ assigned to it, Then there is some stage t such that for all $s \geqslant t$ and $j \leqslant 2^{|\alpha|+i+1}$, $x(\alpha,j)\!\downarrow\![t] = x(\alpha,j)[s]$, and $(\langle s^\alpha, 1^{\psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1})}\rangle \in W^\beta)[s]$.*

Proof. By induction on $|\alpha|$, for all $\gamma \subset \alpha$ and Φ', Ψ', i' such that $\mathrm{S}_{\Phi',\Psi',i'}$ is assigned to γ, there is some stage after which $x(\gamma,j)\!\downarrow$ with the same value for every $j \leqslant 2^{|\gamma|+i'+1}$. Let t_0 be the either this stage or the last stage at which α is initialized, The requirement $\mathrm{S}_{\Phi,\Psi,i}$ can only be assigned to a node extending some $\beta^\frown 0$ such that β has requirement $\mathrm{R}_{\Phi,Psi}$ assigned to it. For such a $\beta \subset f$, $\limsup_{s\to\infty} l^\beta[s] = \infty$. Hence, after t_0, nothing can prevent α from choosing all its witnesses $x(\alpha,1)$, $x(\alpha,2)$,*etc.*, and nothing can cause these witnesses to later diverge, once chosen. □

Naturally, we just write s^α and $x(\alpha,j)$ without reference to the stage for these final values. Note that for any $\gamma \subset \alpha \subset f$ and j' and j, $x(\gamma,j') < x(\alpha,j)$.

By Lemma 16.3.5, the true path is infinite, and it follows, again by a straightforward induction, that every requirement $\mathrm{R}_{\Phi,\Psi}$ is assigned to some node along it. It remains to be shown that all these requirements are satisfied by the strategies of the associated nodes on the true path.

Suppose $\beta \subset f$ with requirement $\mathrm{R}_{\Phi,\Psi}$ assigned to it. If it is not the case that $\Psi(\Phi(A)) = A$, then there is nothing to prove, so suppose that this is the case. In this case, $\beta^\frown 0 \subset f$. Since requirements are only added to a list $L(\gamma,\cdot)$ when $L(\gamma^\frown 0,\cdot)$ and $L(\gamma^\frown 1,\cdot)$ are defined, each subrequirement $\mathrm{S}_{\Phi,\Psi,i}$ is assigned to some node included in f. The following lemmas about β verify that the requirement is satisfied.

Lemma 16.3.6. *$\mu(M^\beta)$ is a computable real.*

Proof. Clearly $\mu(M^\beta)$ is c.e. We show that there is a nonincreasing computable function $e^\beta[s]$ such that $\lim_{s\to\infty} e^\beta[s] = 0$ and for all s $\mu(M^\beta) - \mu(M^\beta)[s] < e^\beta[s]$ If W^β is finite, then there is nothing to prove. Otherwise, set $e^\beta[0] = 1$. Given $s > 0$, we wait for the next stage $t > s$ such that a new element is enumerated into $W^\beta[t+1]$. Since at each $t' > s$ such that a new element is enumerated into $W^\beta[t'+1]$, all subsequent enumerations into W^β individually add some unique number less than $2^{-t'}$ to $\mu(M^\beta)$, it follows that $\mu(M^\beta) - \mu(M^\beta)[s] < \sum_{t'>t} 2^{-t'} = 2^{-t}$. Hence setting $e^\beta[t'] = e^\beta(s-1)$ for all t' with $s \leqslant t' \leqslant t$ and $e^\beta(t+1) = 2^{-t}$ works as claimed. Notice that this also shows that $\mu(M^\beta) < 2^0 = 1$, so that M^β is a well-defined prefix-free machine. This suffices to prove the lemma. □

Lemma 16.3.7. *For M is a prefix-free machine, then for all c there exists some k such that $K_M(\Phi(A) \upharpoonright k) > K_{M^\beta}(1^k) + c$*

Proof. By a suitable version of the padding lemma, there exist infinitely many i such that M is equivalent to M_i. Choose such an $i \geqslant c$. Let $\alpha \supset \beta^\frown 0$ be the unique node included in f such that $\mathrm{S}_{\Phi,\Psi,i}$ is assigned to α. Suppose we have passed the stage after which α is last initialized, so that only strategies assigned to node $\gamma \geqslant \alpha$ act at any later stage. We show that $K_{\mathrm{M}}(\Phi(A) \upharpoonright \psi(\Phi(A); x(\alpha, 2^{s^\alpha+i+1})))) > s^\alpha + i \geqslant K_{\mathrm{M}^\beta}(1^{\psi(\Phi(A); x(\alpha, 2^{s^\alpha+i+1})))}) + i$. If there exists a witness $x(\alpha,j)$ such that $x(\alpha,j) \notin A$, then Suppose $x(\alpha,j)$ is added to A at stage $s{+}1$. Note that $x(\alpha,j) > \max\{\, \varphi(A;y)[s] \ :\ y < \psi(\Phi(A); x(\alpha,j{-}1))))[s]\,\}$, and all nodes $\gamma > \alpha$ are initialized at $s{+}1$ Hence if s^+ is any subsequent β-expansionary stage before $x(\alpha,j{-}1)$ is added to A, we must have $\Phi(A)[s^+] \upharpoonright$

$1^{\psi(\Phi(A);x(\alpha,j-1)))}[s] = \Phi(A)[s] \upharpoonright \psi(\Phi(x(\alpha,j-1)))$. Thus if $s_1 < s < 2 <, \ldots, s_{2^{s^\alpha+i+1}}$ is the sequence of stages such that $x(\alpha,j) \in A[s_j{+}1] - A[s_j]$, there must be a sequence of distinct strings σ_0, σ_2,...,$\sigma_{2^{s^\alpha+i+1}}$ such that for each j, $\mathrm{M}(\sigma_j){\downarrow} = \Phi(A)[s_j] \upharpoonright \psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1}))$ and $|\sigma_j| \leqslant s^\alpha + i$. But then $\mu(\mathrm{M}) \geqslant 2^{s^\alpha+i+1} \cdot 2^{-s^\alpha-i} \geqslant 2^1 > 1$, a contradiction. Hence not all witnesses can be added, so that $K_\mathrm{M}(\Phi(A) \upharpoonright \psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1}))) > s^\alpha + i \geqslant K_{\mathrm{M}^\beta}(1^{\psi(\Phi(A);x(\alpha,2^{s^\alpha+i+1}))}) + i$.

□

The last two lemmas establish the result.

□

The following two corollaries are immediate:

Corollary 16.3.8. *There is a c.e. degree containing no K-trivial real.*

Corollary 16.3.9. *There is a c.e. degree containing no Schnorr trivial real.*

16.3.2 Schnorr triviality and strong reducibilities

Theorem 16.3.10. *No c.e. real α can be both wtt-complete and Schnorr trivial.*

Proof. Suppose α is wtt-complete. We will construct a c.e. set D that forces α to change too often to be Schnorr trivial. Using the method of standard proofs of Lachlan's Non-diamond theorem (see Soare, [280], Chapter IX) we can assume that a wtt-reduction Γ such that $\Gamma(\alpha) = D$ is given in advance. More precisely, we define an infinite sequence of constructions of c.e. sets D_e, each one using a p.c. functional Φ_e. Because for each c.e. D_e, D_e is uniformly m-reducible to $K \leqslant_{\mathrm{wtt}} A$, we have a computable index $g(e)$ for a p.c. functional $\varphi_{g(e)}$ such that $\varphi_{g(e)}(A) = D_e$. By the recursion theorem, for some e, $\varphi_e = \varphi_{g(e)}$, so that we can take $\varphi_{g(e)} = \Gamma$.

We must satisfy the sequence of requirements
R_e: $\exists x\, K_{\mathrm{M}_e}(\alpha \upharpoonright x) \geqslant K_\mathrm{M}(1^x){+}e$.
for all $e \in \omega$. The strategy is straightforward: we choose some (large) number m, and followers $x_1 < x_2 < \ldots < x_m$ to use in satisfying this requirement. We then enumerate $\langle -e{+}\log m, 1^{\gamma(x_m)+1}\rangle$ into a c.e. set defining M at some stage s_{m+1}. In general, given s_k, we wait for a stage s such that $s > s_k$ at which some σ_m appears with $|\sigma_m| < \log m$ such that $(\mathrm{M}_e(\sigma) = \alpha \upharpoonright \gamma(x_m){+}1)[s]$, then we enumerate $x_{k-1)}$ into D. In order for $\Gamma(\alpha;y)[s]$ to change value on a $y \leqslant x_m$, $\alpha[s]$ must change on some $z < \gamma(x_m)$. Now, since $\Gamma(\alpha) = D$, we must have some $s_{k-1} > s$ such that $\alpha \upharpoonright \gamma(x_{k-1}{+}1)[s] \neq \alpha \upharpoonright \gamma(x_{k-1}){+}1)[s_{k-1)}]$. Since α is c.e., this means the approximation to α must increase by some amount greater than $2^{-\gamma(x,m)}$.

Hence, if

$$K_{\mathrm{M}_e}(\alpha \upharpoonright \gamma(x_m){+}1) < K_{\mathrm{M}}(1^{\gamma(x_m)+1}){+}e = \log m,$$

there must exist a sequence $\sigma_m, \ldots, \sigma_1$ of distinct strings of length less than $\log m$ such that for all k, $\mathrm{M}_e(\sigma_k)\downarrow$. But then $\mu(\mathrm{M}_e) > m2^{\log m} = 1$. This contradicts M_e being a prefix-free machine.

The only difficulty involves choosing m and the witnesses $x_1 \ldots, x_m$ so that strategies for different requirements don't interfere with each other. We use a finite-injury priority argument to achieve this.

At stage 0 we let $m^0[0] = 1$, $x_1^0[0] = 0$, $x_2^0 = 1$.

Stage $s{+}1$:

First, choose e least so that $m^e\downarrow[s]$, $x^e_{m^e}\downarrow[s]$, $s^e\downarrow[s]$, $\Gamma(\alpha; x^e_{m^e})\downarrow[s]$, and there exists some $\sigma \in 2^{<\omega}$ such that $|\sigma| < \log m^e[s]$, $(\mathrm{M}_e(\sigma) = \alpha \upharpoonright \gamma(x^e_{m^e}){+}1)[s]$, and $\mu(\mathrm{M}_e)[s] \leqslant 1$. If k is greatest such that $x^e_{k-1} \notin D[s]$, then let $x^e_{k-1)} \in D[s{+}1]$. For all $e' > e$, undefine all functionals and parameters associated to the strategy for $\mathrm{R}_{e'}$.

Next, choose e least so that $m^e\downarrow[s]$, $x^e_{m^e}\downarrow[s]$, $\Gamma(\alpha; x^e_{m^e})\downarrow[s]$, but $s^e\uparrow[s]$. Enumerate the pair $\langle -e{+}\log m^e, 1^{\gamma(x^e_m)+1}\rangle$ into the c.e. set defining machine M. Set $s^e\downarrow[s] = s{+}1$.

Finally, choose e least so that $m^e\uparrow[s]$. Let $m_e\downarrow[s]$ with value the least number greater than or equal to 2^{s+e} that has never yet been a value $m^{e'}[s']$ for any e' and $s' \leqslant s$. Let b be the least number greater than any yet mentioned in the construction. for all j with $1 \leqslant j \leqslant m^e[s]$, let $x^e_j\downarrow[s] = b{+}j$.

This completes the construction.

Notice first that M is a computable machine, since at stage s, the set $\{\, e : m^e\downarrow[s] \,\}$ is finite. All values $m^e[s']$ that are defined after stage s are greater than 2^s, and they are all distinct. If we wait for a stage $t > s$ such that for all such e, either $m^e[s] \neq m^e[t]$, or $\langle -e{+}\log m^e, 1^{\gamma(x^e_m)+1}\rangle$ is in the c.e. set that defines M, then, $\mu(\mathrm{M}) - \mu(\mathrm{M})[s] \leqslant \sum_{m>s} 2^{-m} = 2^{-s}$. This gives a computable nonincreasing function with limit 0 that bounds the error, so that $\mu(\mathrm{M})$ is computable.

Consider a fixed requirement, R_e, and suppose that for all $e' < e$, $\mathrm{R}_{e'}$ is satisfied, and the strategy for $\mathrm{R}_{e'}$ only changes D and M finitely often. Once no strategy for any such $\mathrm{R}_{e'}$ ever acts again, the functionals and parameters for R_e are, once defined, defined permanently. Thus the actions taken in the third phase of the construction at stage $s{+}1$ guarantee that $m^e\downarrow$, $s^e\downarrow$, and for all j with $1 \leqslant j \leqslant m^e$, $x^e_j\downarrow$ with final values. At stage s^e, $x^e_j \notin D$ for all $j \leqslant m$. Now, $m^e > 2^{e+1}$, $-e{+}\log m^e > 1$, and Γ is a total function. Hence, the action in the second phase of the construction guarantees that $K_{\mathrm{M}}(1^{\gamma(x^e_m)+1}) \leqslant -e{+}\log m^e$. There can exist only m^e different stages after this point at which D is changed for the sake of the R_e-strategy. As pointed out before the description of the construction, if

$K_{\mathrm{M}_e}(\alpha \upharpoonright \gamma(x_m^e)+1) < K_{\mathrm{M}}(1^{\gamma(x_m^e)+1})+e$, then, the action taken in the first phase of the construction at stage $s+1$ guarantees that there must exist a sequence $\sigma_m, \ldots, \sigma_1$ of distinct strings of length less than $\log m^e$ such that for all k, $\mathrm{M}_e(\sigma_k)\downarrow$. But then $\mu(\mathrm{M}_e) > m^e(2^{\log m^e}) = 1$. By the Chaitin-Kraft inequality, this is a contradiction since M_e is a prefix-free machine. $\square$

In Chapter 10 we saw that Schnorr reducibility was related to truth table reducibility. This is also true for Schnorr trivials.

Theorem 16.3.11 (Downey, Griffiths, LaForte [74])**.** *If y is Schnorr trivial and $x \leqslant_{tt} y$ then x is Schnorr trivial.*

Proof. We must show that for any computable machine M, there exists some computable M_x and constant c such that for every $n \in \omega$, $K_{\mathrm{M}_x}(x \upharpoonright n) \leqslant K_M(1^n) + c$.

Suppose that the truth-table reduction is given by $x = \Gamma^y$ with use bounded by the strictly increasing recursive function $\gamma(n)$. Given any computable machine M we first define another computable machine M_u such that

$$(\forall n) K_{\mathrm{M}_u}(1^{u(n)}) \leqslant K_M(1^n).$$

To define M_u simply follow the enumeration of axioms into M. Every time $\langle \sigma, \tau \rangle$ enters M, then put the same axiom into M_u unless $\tau = 1^k$ for some k. In that case put $\langle \sigma, 1^{\gamma(k)} \rangle$ into M_u. $\mu(\mathrm{M}_u) = \mu(\mathrm{M})$, so that M_u is also a computable machine. Evidently, M_u is as required.

Now as y is Schnorr trivial, there exists a computable machine M_y and constant c such that $(\forall k) K_{\mathrm{M}_y}(y \upharpoonright k) \leqslant K_{\mathrm{M}_u}(1^k) + c$. In particular, for all n,

$$K_{\mathrm{M}_y}(y \upharpoonright \gamma(n)) \leqslant K_{\mathrm{M}_u}(1^{\gamma(n)}) + c \leqslant K_{\mathrm{M}}(1^n) + c$$

Now we define a machine M_x with the same domain as M_y to show that x is Schnorr trivial. If $\mathrm{M}_y(\sigma) = \tau$ then let $\mathrm{M}_x(\sigma) = (\Gamma^{\tau \upharpoonright \gamma(\widehat{n})} \upharpoonright \widehat{n})$ for the largest $\widehat{n}$ with $\gamma(\widehat{n}) \leqslant |\tau|$. Then, if $\mathrm{M}_y(\sigma) = y \upharpoonright \gamma(n)$, we have $\mathrm{M}_x(\sigma) = (\Gamma^{y \upharpoonright \gamma(n)} \upharpoonright n) = x \upharpoonright n$, so that

$$K_{\mathrm{M}_x}(x \upharpoonright n) = K_{\mathrm{M}_y}(y \upharpoonright \gamma(n)) \leqslant K_{\mathrm{M}_u}(1^{\gamma(n)}) + c \leqslant K_{\mathrm{M}}(1^n) + c.$$

Since the tt reduction converges with any string as an oracle, $\mu(\mathrm{M}_x) = \mu(\mathrm{M}_y)$, and so M_x is a computable machine. $\square$

From this it would be easy to show that the Schnorr trivials form an ideal in the tt degrees. All we would need would be a positive solution to the following question.

Question 16.3.12. *Suppose x and y are Schnorr trivial reals. Then is $x \oplus y$ Schnorr trivial?*

However, the question remains open.

16.4 Kurtz lowness

Again note that we have two possible notions of lowness: Kurtz lowness for tests, and lowness for Kurtz randoms. At the time of the writing of this book, it is unknown if these two notions differ. The first result of the present section gives sandwiches the Kurtz low reals between the class of hyperimmune free ones and the computably traceable ones. We also give the proof of Stephan and Yu [295] showing that the Kurtz low reals do not coincide with the Schnorr low ones. Finally we will mention some relationships with lowness for weak 2-genericity. For the remainder of this section, we "Kurtz low" will mean the strong notion of being Kurtz low for tests.

Theorem 16.4.1 (Downey, Griffiths, Reid [75]). *(i) If a set A is Schnorr low, then it is low for Kurtz null tests.*

(ii) If a set A is low for Kurtz null tests, then it is of hyperimmune-free degree. Indeed, if A is low for Kurtz randomness, then A has hyperimmune free degree.

Proof. (i) We show that for every Kurtz null test computable in a computably traceable set A with null set N_A there is a Kurtz test with no oracle capturing the same reals. Let $g : \omega \to \omega$ be defined by $g(n) = \langle d(n), D_k\rangle$ where $d(n)$ specifies the length of strings in the n^{th} set of a Kurtz null test computable relative to A, and D_k specifies which of the $2^{d(n)}$ strings of this length are included in the set. Choose a recursive trace T for g with the identity function as its recursive bound. Let $f(n) = max\{d'(n) \mid \langle d'(n), D_k\rangle \in T^{[k]}$ for some $D_k\}$, then f is a computable function of n, and let E_n be the union of all sets D_k coded in $T^{[n]}$, expressed as strings of length $f(n)$. Then $\mu(E_n) \leqslant n2^{-n}$, and by selecting a suitable computable sequence $n_1 < n_2 < ...$ we can obtain a Kurtz null test $f(n_i), E_{n_i}$, $i \in \omega$, such that $\cap_i E_i \supseteq N_A$.

(ii) In Corollary 11.16.11, we proved that if X is of hyperimmune degree then $A \equiv_T B$ with B Kurtz random. Clearly no degree containing a Kurtz random can be low for Kurtz randomness. □

As it turns out, our old friends *diagonally noncomputable functions* will be important to our story. Recall that a degree $\mathbf{a}$ is DNC iff there is a function $f \leqslant \mathbf{a}$ such that $f(x) \neq \varphi_x(x)$ for all x. In Chapter 11, we saw that all random degrees are DNC. (Theorem 11.5.1.) We need the following which relates being DNC to Kolmogorov complexity. Recall that in Theorem 11.11.7, Kjos-Hanssen, Merkle, Stephan [**?**] established several equivalent forms of being dnc in terms of autocomplexity. That is the following were shown equivalent.

(i) x is DNC or has high degree.

(ii) x is autocomplex or has high degree.

(iii) There is a $g \leqslant x$ such that for every computable function h, $g(n) \neq h(n)$ for almost all n.

(iv) There is no weak computable tracing of x in that there is no computable h such that for all $f \leqslant_T x$ there is a computable function g with $|D_{g(n)}| \leqslant h(n)$ for all n and $\exists^\infty n f(n) \in D_{g(n)}$.

(v) There is an order h and a function $g \leqslant_T x$ such that for almost all n, $C(g(n) \geqslant h(n)$.

Furthermore, Kjos-Hanssen, Merkle, Stephan [**?**] showed that if y is a real which is of hyperimmune free degree and which is not DNC, and $g \leqslant y$, then there there are computable functions h and $\widehat{h}$ such that

$$\forall n \exists m \in \{n, n+1, \ldots, \widehat{h(n)}\}(h(m) = g(m).$$

Theorem 16.4.2 (Stephan and Yu [295])**.** *Suppose that x is of hyperimmune free degree and is not DNC. Then*

(i) Every Σ_1^x class S or measure 1 has a measure 1 Σ_1^0 subclass T.

(ii) Every dense Σ_1^x class has a dense Σ_1^0 subclass.

Proof. We can obtain (ii) from the proof of (i) below. Without loss of generality, we assume that S is dense and has has measure 1. Suppose that x satisfies the hypotheses of the Theorem. There is a function $f \leqslant_T x$ such that for all n, $f(n) > n$, $\forall \sigma \in 2^n \exists \tau \in 2^{f(n)}$ such that τ extends σ and $[\tau] \in S$. Furthermore, f can be chosen so that the $\mu(\{y \in S : [y \restriction f(n)] \subseteq S\}) \geqslant 1 - 2^{-n}$. Since x has hyperimmune degree, there is a computable function q such that $q(n+1) > f(q(n))$ for all n. Consequently, there is a x-computable g such that for all n, $g(n) \subset 2^{q(n+1)}$, $[g(n)] \subseteq S$, $\forall \sigma \in 2^n \exists \tau \in g(n)(\sigma \preccurlyeq \tau)$, and furthermore $\mu([g(n)]) \geqslant 1 - 2^{-n}$.

Now we will apply Theorem 11.11.7. Since x is neither autocomplex nor DNC, there are computable functions h and $\widehat{h}$ such that $h(n) \in 2^{q(n+1)}$, $\forall \sigma \in 2^{q(n)} \exists \tau \in h(n)(\sigma \preccurlyeq \tau)$ and $\mu([h(n)) \geqslant 1 - 2^{-n}$, with

$$\exists m \in \{n, n+1, \ldots, \widehat{h}(n)\}(h(m) = g(m)).$$

Now define the Σ_1^0 class T as

$$T = \{x : \exists n \forall m \in \{n, n+1, \ldots, \widehat{h}(n)\}(x \restriction q(m+1) \in h(m)\}.$$

The class T is dense because each $\sigma \in 2^n$ is extended by a $\tau_{n-1} \in 2^{q(n)}$ as $q(n) > n$ and a sequence of $\tau_m \in h(m)$ extending τ_{m-1} for $m = \{n, n+1, \ldots, \widehat{h}(n)\}$. Thus $[\tau_{\widehat{h}(n)}] \subseteq T$. Finally the measure of T is 1 as, for all n,

$$\mu(\{x : \forall m \in \{n, n+1, \ldots, \widehat{h}(n)\}(x \restriction m \in h(m)\}) \geqslant 1 - 2^{-n}.$$

Also for each $x \in T$ there is an n and $m \in \{n, n+1, \ldots, \widehat{h}(n)\}$ such that $g(m) = h(m)$ and $x \in [h(m)]$. Thus $x \in [g(m)]$ and $T \subseteq S$. $\square$

Corollary 16.4.3 (Stephan and Yu [295]). *Suppose that x is x is of hyperimmune free degree and is not DNC. Then x is low for Kurtz randomness. Thus there are reals that are not computably traceable and yet are Kurtz low.*

Proof. It is enough to show that there are reals which are not computably traceable and yet are of hyperimmune free degree and not DNC. This is a straightforward Spector Forcing plus diagonalization argument. Specifically, at even stages do the usual hyperimmune free construction with perfect trees, yielding a tree T_{2e} which forces Φ_e^A. At odd stages, we deal with Π_e^A making sure that it is not fixed point free. If it total, we make the stem long enough to agree with the patrial computable function which picks out the leftnost path of the e-th computable tree if that tree is infinite. $\square$

One can do rather better. First we could have constructed a perfect tree of such reals in the usual way. The following proves something a little stronger.

Theorem 16.4.4 (Stephan and Yu [295]). *There is a perfect Π_1^0 class of reals which are neither autocomplex nor c.e traceable.*

Proof. We construct a Π_1^0 class $[P]$ in stages. It is convenient to construct the underlying tree P as part of computable tree T such that if σ if on T then $\sigma\widehat{\ }j$ is also where $j = 1, \ldots, n$ where $n = |\sigma|$.

To make all the paths on A not autocomplex, we appeal to Theorem 11.11.7, making each path not DNC. Thus we will meet requirements R_e diagonalizing Φ_e as a possible fixed point free function computable from a path of $[P]$. To do this, we will define $\varphi_{g(n)}$ for infinitely many n (g being given by the Recursion Theorem), and ask that for each e, and path $A \in [P]$ with Φ_e^A total, there is some n with

$$\Phi_e^A(n) = \varphi_{g(n)}(g(n)).$$

For the sake of e there will be a finite collection of cones $[\sigma_1^s], \ldots, [\sigma_m^s]$ with $P_s = \cup_i [\sigma_i^s]$. Working in σ_i^s, we will pick a large number $d = g(n)$ for some n, and wait till there is some τ extending σ_i^s with $\Phi_e^\tau(d) \downarrow [s]$. Should this not happen then Φ_e^A is not total on the cone $[\sigma_i^s]$. Should it happen, then we will define $\varphi_d(d) = \Phi_e^\tau(d)$, and ask that $\sigma_i^{s+1} = \tau$, (that is, killing all ρ on $P_s \cap [\Sigma_i^s]$ not extending τ.) initializing lower priority requirements. R_e will now be met in the cone $[\sigma_i^{s+1}]$. For technical reasons, ask that $\sigma_i^{s+1}\widehat{\ }j$ for all $1 \leqslant j \leqslant |\sigma_i^{s+1}$ be in P_t for $t \geqslant s+1$ with priority that of R_e.

The other requirements Q_e ask that if h_e is the e-th partial computable function (representing a possible order), if h_e is total then h_e is not a witness to the c.e. traceability of $A \in [P]$. For the sake of this we will be defining a total function Ψ_e^A for $A \in [P]$ should h_e be total. Q_e is broken up into diagonalizing against possible $W_{k_i(n)}$, possible c.e. traces with $|W_{k_i(n)}| \leqslant h_e(n)$. This is given priority $Q_{e,i}$.

The strategy is again simple. For the sake of $Q_{e,i}$ in a cone $[\sigma^s]$ as above, we will pick a big number n and wait till $h_e(n) \downarrow [s]$. At this stage, $Q_{e,i}$ will assert control and make $\sigma^{s+1} = \tau$ for some τ extending σ^s whose length is at least $h_e(n) + 1$ and has at least $h_e(n) + 1$ many extensions, $\tau\widehat{\ }j$. Then we will define $\Psi_e^\tau(n')$ for all $n' < n$ not already defined, and $\Psi_e^{\tau\widehat{\ }j} = j$ for all $1 \leqslant j \leqslant h_e(n) + 1$. Initialize al lower priority requirements. Then Ψ_e can't be traced by $W_{k_i(n)}$ as it can't have enough elements. The remainder of the proof is a straightforward application of the finite injury method $\square$

We remark that the proof above is rather different than the original method of Yu and Stephan who proved the following.

Theorem 16.4.5 (Stephan and Yu [295])**.** *There is a partial computable function ψ with coinfinite domain such that each x extending ψ is neither autocomplex nor c.e. traceable.*

As we remarked earlier, Kurtz lowness is related to lowness for weak genericity. Here we recall that A is called weakly 1-geeric iff A meets all dense Σ_1^0 sets of strings. This is hardly surprising since Kurtz randoms and weakly 1-generic reals occur in each hyperimmune degree.

Theorem 16.4.2 (ii) above yields the following.

Corollary 16.4.6 (Stephan and Yu [295])**.** *Suppose that A is hyperimmune free and not DNC. Then A is low for weakly 1-genericity.*

Spethan and Yu also established the following completely characterizing the degrees ow for weak 1-genericity.

Theorem 16.4.7 (Stephan and Yu [295])**.** *The following are equivalent.*

(i) Every dense Σ_1^x class has a dense Σ_1^0 subclass.

(ii) x is low for weakly 1-generic.

(iii) The degree of x is hyperimmune free and each 1*-generic real is weakly $1-x-$generic.*

(iv) The degree of x is hyperimmune free and not DNC.

Proof. Clearly (i) implies (ii), and (ii) implies (iii) since Kurtz showed that each hyperimmune degree is weakly-1-generic. (iv) implies (i) by Theorem 16.4.2 above. Finally (iii) implies (iv) by the lemma below. $\square$

Lemma 16.4.8 (Stephan and Yu [295])**.** *If each $1-$generic real is weakly $1-x-$generic then x is not DNC.*

Proof. Suppose that x is DNC and each 1-generic set is also weakly $1 - x-$generic. Since x is DNC, x is autocomplex by Theorem 11.11.7. Therefore there is a x-computable function f with $K(x \restriction m) \geqslant n$ for all

$m \geqslant f(n)$. Without loss of generality, we may assume $f(n)$ only queries x below $f(n)$ on input n. Define

$$S = \{\sigma(x \upharpoonright f(|\sigma|)) : \sigma \in 2^{<\omega}\}.$$

Then S is dense. Choose y to be K-trivial and 1-generic (since each c.e. degree bounds a 1-generic one). We know that y is weakly $1 - x-$generic. Thus y meets S infinitely often. That is, there are infinitely many n with $(y \upharpoonright n)(x \upharpoonright f(n)) \preccurlyeq y$. Given such an n, to compute $f(n)$ relative to y, we query $y \upharpoonright m + n$ whenever the original computation of $f(n)$ queries $x(m)$. Note $m < f(n)$ and hence $y(m+n) = x(m)$. Now as y is K-trivial, it is low for K, and hence $K^y(\tau) = K(\tau) + \mathcal{O}(1)$ for all τ. We have the following.

$$K(x \upharpoonright f(n)) \leqslant^+ K^y(x \upharpoonright f(n)) \leqslant K^y(y \upharpoonright n+f(n), n)+\mathcal{O}(1) \leqslant K^y(f(n), n)+\mathcal{O}(1).$$

and so $K(x \upharpoonright f(n)) \leqslant K^y(n) + \mathcal{O}(1) \leqslant K(n) + \mathcal{O}(1)$. Therefore there are infinitely many n with $K(n) \geqslant n - \mathcal{O}(1)$, a contradiction. □

The final part of the story here was solved by Greenberg and Miller [**?**] who proved the following.

Theorem 16.4.9 (Greenberg and Miller [**?**]). *A real A is low for weak randomness iff A is low for weak genericity. Hence A is low for weak randomness iff A is hyperimmune free and DNC.*

We end this section by remarking that there are no reals which are low for 1-generic. Lowness for 1-genericity and for weak 1-genericity was first investigated by Nitzpon [**?**]who had proven the earlier result that if A is low for weak genericity then x is hyperimmune free.

Theorem 16.4.10 (Greenberg and Miller unpubl., Yu [**?**]). *Suppose that x is low for 1-genericity. Then x is computable.*

Proof. Let x be noncomputable. Then by theorem 11.15.5, there is a 1-generic z such that $z \oplus x \equiv_T x'$. Therefore

$$(z \oplus x)' \equiv_T x'' > x' \equiv_T z \oplus x'.$$

By Theorem 11.15.4, if z is $1 - x-$generic, then $(z \oplus x)' \equiv_T z \oplus x'$. But this means z is not $1 - x-$generic. □

16.5 Lowness for computable randomness

In Section 16.1 we have seen that the real low for Schnorr and Kurtz randomness are all hyperimmune-free. This is hardly surprising since Schnorr randomness and Kurtz randomness are both concerned with *total* functions, and tests where the measure is a total computable function. Martin-Löf randomness concernes partial computable functions, and this is perhaps why lowness for it is related to *jump* traceability orather than *traceability*.

What of computable randomness? Here the graded tests are *somewhat* computable but the overall measure is not computbale. The situation for computable randomness turns out to be dramatically different. In this section, we prove the beautiful result of Nies [227], below, which verified a conjecture of Downey.

Theorem 16.5.1 (Nies [227]). *Suppose that A is low for computable randomness. Then A is computable.*

We will give another proof than the one given by Nies in [226, **?**]. This proof is also by Andre' Nies (Nies [229]) and is presented with his permission. We will denote the class CR as the computably random reals. That is, the reals where no computable martingale succeeds. Using A-computable martingales gives a generally smaller class CR^A. A is low for computably random (which we denote by Low(CR)) just if CR^A is as large as possible, namely $\mathsf{CR}^A = \mathsf{CR}$. All martingales will be $\mathbb{Q}$-valued which we have already seen poses no restriction.

Theorem 16.5.2. *Each low for computably random real is computable.*

Given v, let

$$\widehat{M}(y) = \max\{M(y') : v \preccurlyeq y' \preccurlyeq y \wedge M(y')\text{defined}\}.$$

By Kolmogorov's inequality, for $M(v) < b$,

$$\mu\{z \succcurlyeq v : \widehat{M}(v) \geqslant b\} \leqslant M(v)/b2^{-|v|} \tag{16.1}$$

The following lemma is trivial but very useful.

Lemma 16.5.3 (Non-ascending path trick, NAPT). *Suppose M is a martingale which is computable (in running time τ). Then, for each string z and each $u > |z|$ we can compute (in time $u\tau(u)$) a string $w \succ z$, $|w| = u$, such that $M(w \upharpoonright q+1) \leqslant M(w \upharpoonright q)$ for each q, $|z| \leqslant q < u$.*

We say that a martingale B has the "savings property" if

$$x \prec y \Rightarrow B(y) \geqslant B(x) - 2. \tag{16.2}$$

We have already seen that martingales can be taken to be $\mathbb{Q}$-valued martingales with the savings property.

A *martingale operator* (MGO) is a Turing functional L such that, for each oracle X, L^X is a total martingale. For a string γ, we write $L^\gamma(x) = p$ if this oracle computation converges with all oracle questions less than γ. To prove Theorem 16.5.2 we will define a MGO L (which can be computed in quadratic time). We will apply the following purely combinatorial Lemma to $N = L^A$ and the family (B_i) of martingales with the savings property characterizing the randomness notion in question. It says that for some positive linear combination M of the martingales B_i, and for some d, $N(w) \geqslant 2^d$ implies $M(w) \geqslant 2$ in an interval $[v]$, while $M(v) < 2$.

Lemma 16.5.4. *Let N be any martingale such that $N(\lambda) \leqslant 1$. Let $(B_i)_{i\in\mathbb{N}}$ be some family of martingales with the savings property (16.2). Assume that*

$$S(N) \subseteq \bigcup_i S(B_i).$$

Then there are $v \in 2^{<\omega}$ and $d \in \mathbb{N}$ and a martingale M which is a finite linear combination $\sum_{i=0}^{n} q_i B_i$ with rational positive coefficients such that $M(v) < 2$ and

$$\forall x \succ v[\ N(x) \geqslant 2^d \Rightarrow M(x) \geqslant 2]. \tag{16.3}$$

Proof. If the lemma fails, then for each linear combination $M = \sum_{i=0}^{n} q_i B_i$, $q_i \in \mathbb{Q}^+$,

$$\forall v \forall d\ [M(v) < 2 \Rightarrow \exists w \succ v\ (N(w) \geqslant 2^d \wedge M(w) < 2)]. \tag{16.4}$$

We define a sequence of strings $v_0 \prec v_1 \prec \dots$ and rationals $q_i > 0$ such that

$$N(v_n) \geqslant 2^n - 1 \wedge \sum_{i=0}^{n} q_i B_i(v_n) < 2. \tag{16.5}$$

Let $v_0 = \lambda$ and $q_0 = 1$, so that (16.5) holds for $n = 0$. Now suppose that $n > 0$ and v_{n-1}, q_{n-1} have been defined, and 16.5 hold for $n-1$. Let

$$q_n = \tfrac{1}{2} 2^{-|v_{n-1}|}(2 - \textstyle\sum_{i=0}^{n-1} q_i B_i(v_{n-1})),$$

so that $M(v_{n-1}) < 2$ where $M = \sum_{i=0}^{n} q_i B_i$ (note that $0 < q_n \leqslant 1$). Applying (16.4) to $v = v_{n-1}, d = n$ there is $v_n = w \succ v$ such that $N(w) \geqslant 2^n$ and $M(w) < 2$.

If $Z = \bigcup_n v_n$, then N succeeds on Z (interestingly, not necessarily in the effective sense of Schnorr). On the other hand, for each $n \geqslant i$, $q_i B_i(v_n) < 2$. Since B_i has the savings property (16.2), $\limsup_n B_i(Z \upharpoonright n) \leqslant 2 + 2/q_i$, so B_i does not succeed on Z. □

A *partial computable martingale* is a partial computable function $M : 2^{<\omega} \mapsto \mathbb{Q}$ such that $\mathrm{dom}(M)$ is $2^{<\omega}$, or $2^{\leqslant n}$ for some n, $M(\lambda) \leqslant 1$, and M has the martingale property $M(x0) + M(x1) = 2M(x)$ whenever $x0, x1$ are in the domain. Clearly there is an effective listing $(M_e)_{e\in\mathbb{N}}$ of partial computable martingales with range included in $[1/2, \infty)$. We let τ_e be the partial recursive function such that $\tau_e(n) \sim$ the maximum running time of $M_e(w)$ for any w of length n (this includes the linear slow down since we need an effective listing).

Proof. (of Theorem 16.5.2)

Fix an effective listing $(\eta_m)_{m\geqslant 1}$ of all triples

$$\eta_m = \langle e, v, d \rangle \tag{16.6}$$

where v is a string, and $e, d \in \mathbb{N}$ (e is an index for a partial computable martingale M_e). We think of η_m as a witness as in Lemma 16.5.4, where (B_i) is the family of all (total) computable martingales with the savings property (not an effective listing).

We will independently build MGOs L_m for each $m \geqslant 1$ which have value 2^{-m} on any input of length $\leqslant m$. L_m is computable in linear time, for a fixed constant. Then $L = \sum_{m \geqslant 1} L_m$ is a MGO (L is $\mathbb{Q}$-valued since the contributions of L_m, $m > |w|$, add up to $2^{-|w|}$), and L is computable (in quadratic time).

We define L in order to ensure that for each A, if $N = L^A$ and $S(N) \subseteq \bigcup_i S(B_i)$ (which is the case if A is low for computably random), then we can compute A. The computation procedure for A is based on a witness $\eta_m = \langle e, v, d \rangle$ given by Lemma 16.5.4, so M_e is total. Since we cannot determine the witness effectively, to make L a MGO we need to consider all η_m together, including those where M_e is partial.

The idea how to compute A is this. Once L is defined, if η_m is a witness for Lemma 16.5.4 where $N = L^A$, let $M = M_e$ and consider the tree

$$T_m = \{\gamma : \forall w \succcurlyeq v (L_m^{\gamma}(w) \geqslant 2^d \Rightarrow M(w) \geqslant 2)\}.$$

Since η_m is a witness and $L^A \geqslant L_m^A$, A is a path of T_m. Let $k = 2^{d+m}$, and let $\mathcal{S}_k$ denote the set of k-element sets of strings of the same length. Let α, β range over elements of $\mathcal{S}_k$. We write α_r for the r-th element in lexicographical order ($0 \leqslant r < k$), and identify α with the string $\alpha_0\alpha_1 \ldots \alpha_{k-1}$. For each α, we ensure that $\alpha \not\subseteq T_m$, in an effective way: given α, we are able to find $s < k$ such that $\alpha_s \notin T_m$. This will allow us to determine a tree $R \supseteq T_m$ such that for each j, the j-th level $R^{(j)}$ has size $< k$ and we can compute that level. Then we can compute A: fix j_0 sufficiently large so that only one extension of $A \upharpoonright j_0$ exists in $R^{(j)}$, for each $j \geqslant j_0$. This extension must be $A \upharpoonright j$ since A is a path of T_m and $T_m \subseteq R$. So, given input $p \geqslant j_0$ to compute $A(p)$ we output the last bit of that extension for $j = p + 1$.

Let $z_0, \ldots, z_{k-1}$ be the strings of length $d + m$ in lexicographical order. We describe in more detail the strategy which, given α, produces an s such that $\alpha_s \notin T_m$. Suppose $w \succ v$ is a string such that $M(w) < 2$, and no value $L^{\gamma}(w')$ has been declared for any $w' \preccurlyeq w$ (we call w an *α-destroyer*). In this case we may define $L_m(w) = 2^{-m}$ regardless of the oracle. For each $s < k$, we ensure $L_m^{\alpha_s}(wz_s) = 2^d$, by betting all the capital along z_s from the end of w on. Since $M(w) < 2$, by the NAPT (16.5.3) we can compute s such that $M(wz_s) < 2$. So $x = wz_s$ is a counterexample to (16.3), so that $\alpha_s \notin T_m$.

We want to carry out this strategy independently for different α. To do so we assign to each α a string y_α. Given η_m, recall $M = M_e$ and

$$\widehat{M}(y) = \max\{M(y') : v \preccurlyeq y' \preccurlyeq y\}.$$

The assignment function $G_m : \mathcal{S}_k \mapsto \{0,1\}^*$ mapping α to y_α (which only is defined when $M_e(v) < 2$) will satisfy the following.

(G1) The range of G_m is an antichain of strings y such that $\widehat{M}(y) < 2$

(G2) G_m and G_m^{-1} are computable in the sense that there is an algorithm to decide if the function is defined and in that case returns the correct value.

We cannot apply the strategy above with $w = y_\alpha$, since we first would need to recover α from w, which may take long or even forever (depending on M_e which may be partial), but also we want L_m to be total, and in fact to be computable in quadratic time. Instead we use the "looking-back" technique. Let $h_m(\alpha)$ be the number of steps required to check that $M_e(v) < 2$, $G_m^{-1}(y_\alpha \upharpoonright p)$ is undefined for $p = 0, \ldots, |y_\alpha| - 1$, and to compute $\alpha = G_m^{-1}(y_\alpha)$. Each $w \succ y_\alpha$ of length $h_m(\alpha)$ is a potential α-destroyer. Now we can recover α from w in linear time, and then define $L_m^{\alpha_r}$ above w according to the strategy above. Figure 16.1 below may be helpful here.

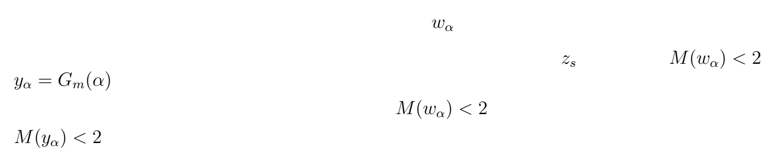

Figure 16.1. Looking back

Given α, to find the actual α-destroyer w, first compute y_α, then $h_m(\alpha)$, and now use the NAPT to find $w \succ y_\alpha$ of length $h_m(\alpha)$ such that $M(w) < 2$. As explained above, use w to determine which α_r is not on T_m.

The choice of G_m is *irrelevant* as long as (G1) and (G2) hold, so we defer defining G_m, but note that the time to compute G_m and G_m^{-1} will be closely related to the running time τ_M of M, since we need to find strings where $\widehat{M}(y) < 2$. Given G_m, the following auxiliary procedure will be used to define L_m and to compute $h_m(\alpha)$.

Procedure P_m ($\eta_m = \langle e, v, d \rangle$)
Input x

1. Let $p = 0$
2. $y = x \upharpoonright p$
3. Attempt to compute $\alpha = G^{-1}(y)$. If defined, output α and $h =$ the number of steps used so far.
4. $p \leftarrow p + 1$. If $p < |x|$ goto 2.

Construction of L_m. We define L_m by declaring axioms of the form $L_m^\gamma(w) = p$, in a way that

(a) $|\gamma| \leqslant |w|$ and one can determine in time $O(|w|)$ whether an axiom $L^\gamma(w) = p$ has been declared, and

(b) whenever distinct axioms $L^\gamma(w) = p$ and $L^\delta(w) = q$ are declared then γ, δ are incompatible.

Then we let $L_m^X(w) = p$ if some axiom $L_m^\gamma(w) = p$ has been declared for $\gamma \subseteq X$, or else if p is the "default value" 2^{-m}. Clearly $L_m^X(w)$ can be determined in time $O(|w|)$ using oracle X.

Given a string x, we declare no axiom for x unless in $|x|$ steps we can determine that $\eta_m = \langle e, v, d\rangle$, and that $M_e(v)$ converges in $< |x|$ steps with value < 2. If so, run at most $|x|$ steps of procedure $P_m(x)$. If there is an output α, h, then let $w = x \upharpoonright h$ and declare axioms as follows (implementing the strategy outlined above): Let $x = wz$. For each $s < k$, let $L_m^{\alpha_s}(x) = 0$ unless z is compatible with z_s. In that case, declare $L_m^{\alpha_s}(x) = 2^{-m+|z|}$ if $z \preccurlyeq z_s$, and $L_m^{\alpha_s}(x) = 2^d$ if $z_s \preccurlyeq z$. *End of construction.*

Clearly (a) holds. Moreover (b) is satisfied since the strings (y_α) form an antichain, we only declare axioms $L_m^\gamma(x) = p$, $y_\alpha \preccurlyeq x$ if $\gamma \in \alpha$, and the individual strings within α are incompatible. Finally L^X is a martingale for each oracle X.

Suppose (B_i) is the family of all (total) computable martingales with the savings property (16.2). If A is Low(CR), then $S(L^A) \subseteq \bigcup_i S(B_i)$. The linear combination M obtained in Lemma 16.5.4 is computable. So the following lemma suffices to compute A, since, as explained above, the existence of the tree R implies that A is computable.

Lemma 16.5.5. *[Computing a thin tree] Suppose $\eta_m = \langle e, v, d\rangle$, where $M = M_e$ is total and M, v, d is a witness for (16.3) in Lemma 16.5.4 where $N = L^A$. Then there is a tree $R \supseteq T_m$ such that for each j, the j-th level $R^{(j)}$ has size $< k = 2^{d+m}$ and we can compute that level from j.*

Proof. Let $R^{(0))} = \{\lambda\}$. Suppose $j > 0$ and we have determined $R^{(j-1)}$. Carry out the following to determine $R^{(j)}$

1. Let F be the set of strings of length j that extend strings in $R^{(j-1)}$ (so $|F| = 2|R^{(j-1)}|$) .
2. While $|F| \geqslant k$: Let α be the lexicographically leftmost size k subset of F.
   (a) Compute $y = G(\alpha)$.
   (b) Apply procedure P_m to y to compute $h = h(\alpha)$.
   (c) By NAPT find $w \succ y_\alpha$ of length h such that $\widehat{M}(w) < 2$.
   (d) Search for $r < k$ such that $\widehat{M}(wz_r) < 2$. Remove α_r from F.
3. Let $R^{(j)} = F$.

□

To conclude the proof it remains to define G_m. We first prove a lemma using the the following instance of Kolmogorov's inequality: for $M(v) < b$,

$$\mu(\{z \succcurlyeq v : \widehat{M}(z) \geqslant b\}|v) \leqslant M(v)/b, \tag{16.7}$$

where $\mu(X|v)$ stands for $2^{|v|}\mu(X \cap [v])$.

Lemma 16.5.6. *Given η_m, suppose that $M(v) < b$, $b \in \mathbb{Q}$. Let*

$$P = \{y \succcurlyeq v : \widehat{M}(y) < b\},$$

and let $r \in \mathbb{N}$ be such that $2^{-r} \leqslant 1 - M_e(v)/b$. Then given i we can compute $y^{(i)}$ of length $i+r+1$ such that $y^{(i)} \in P$ and the strings $y^{(i)}$ form an antichain. If M is partial, we can compute $y^{(i)}$ for each i such that M is defined for strings of length up to $i+r+1$.

Proof. Suppose inductively $y^{(q)}$ has been computed for $q < i$. Since $\sum_{q<i} 2^{-|y^{(q)}|} = 2^{-r}(1-2^{-i})$ and $2^{-r} \leqslant \mu(P|v)$ by Kolmogorov's inequality, one can compute $y \in P$ such that $|y| = i + r + 1$ and $y_q \not\preccurlyeq y$ for all $q < i$. Let $y_i = y$. □

To obtain G_m, Let $b = 2$, and let n_α be a number greater than the length of each string in α, assigned to α in an effective 1-1 way. Let $G_m(\alpha) = y^{(n_\alpha)}$. Clearly (G1) and (G2) hold. □

Extending thses techniques, Nies proves similar results about lowness for partial computable martingales. Recall our notation for lowness properties defined in terms of two randomness notions $\mathcal{C}, \mathcal{D}$. Let $\text{Low}(\mathcal{C}, \mathcal{D})$ denote the class of oracles A such that $\mathcal{C} \subseteq \mathcal{D}^A$.

Theorem 16.5.7 (Nies [229])**.** *Each Low(PrecRand,CRand) real is computable.*

Space considerations preclude us from including a proof of this results, and therefore we refer the reader to Nies [229] for further information.

17
Effective Hausdorff Dimension and s-randomness

17.1 Classical Hausdorff dimension

The study of measure as a way of specifying the size of sets began with work of Borel in 1895, and in work of de la Valee Poisson. In his famous thesis [172], Lebesgue introduced the the measure which is now called Lebesgue measure. In 1914, Carathéodory [40] introduced a more general construction of measure which included Lebesgue measure as a special case. In any n-dimesnional Euclidean space, Carathéodory's general version was to define

$$\mu^s(A) = \inf\{\sum_i |I_i|^s : A \subset \cup_i I_i\},$$

where each I_i is an interval in the space. Finally in 1919 Hausdorff [124] generalized the notion of an s-dimensional measure to include non-integer values.

Classical Hausdorff dimension is a method of classifying measure zero sets, the *leitmotif* being that not all measure zero sets are created equal. In this section we will look at the effective theory of Hausdorff dimension, and its relationship with calibrating randomness.

The main idea is to change the way we measure open sets by an additional factor in the exponent. Classically, this is realized as follows. For $0 \leqslant s \leqslant 1$, the s-measure of a clopen set $[\sigma]$ is

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

Notice that for $s = 1$, the notion of Lebesgue measure and s-measure coincide.

Definition 17.1.1. (i) A set of strings D is called an n-cover if $\sigma \in D$ implies that $|\sigma| \geqslant n$.

(ii) As usual, D covers a set of reals R if $R \subseteq \cup_{\sigma \in D}$.

(iv) Then we define $H_n^s(R) = \inf\{\sum_{\sigma \in D} \mu_s([\sigma]) : D \text{ is a } n\text{-cover of } R\}$.

(v) Then the s-dimensional outer Hausdorff measure of R is

$$H^s(R) = \lim_{n \to \infty} H_n^s(R).$$

This is classically well studied. The fundamental result is the following.

Theorem 17.1.2. *For all $R \subset 2^\omega$, there is an $s \in [0,1]$ such that*

(i) $H^t(R) = 0$ for all $t > s$, and

(ii) $H^u(R) = \infty$ for all $0 \leqslant u < s$.

Proof. Note that for all n, and s', $H_n^{s+s'}(R) = \inf\{\sum_{\sigma \in D} 2^{-s|\sigma|} 2^{-s'|\sigma|} : D\ n$ covers $R\} \leqslant 2^{-s'n} H_n^s(R)$. Then if $H^s(R) = 0$ this formula forces $H_n^{s+s'}(R)$ to be 0 as well. □

Theorem 17.1.2 means that the following definition makes sense.

Definition 17.1.3. For $R \subseteq 2^\omega$, the *Hausdorff dimension* of R is defined as

$$\dim(R) = \inf\{s : H^s(R) = 0\}.$$

Hausdorff dimension has a number of basic properties:

Lemma 17.1.4. *(i) Hausdorff dimension is a refinement of measure zero: If $\mu(X) \neq 0$, then $\dim(X) = 1$.*

(ii) (monotonicity) If $X \subseteq Y$ then $\dim(X) \leqslant \dim(Y)$.

(iii) (countable stability) If P is countable, then $\dim(\cup_{i \in P} Y_i) = \sup_{i \in P}\{\dim(Y_i)\}$. In particular, $\dim(X \cup Y)$ is $\max\{\dim(X), \dim(Y)\}$.

This lemma is well-known but its proof is probably worth sketching. We begin with a lemma.

Lemma 17.1.5. *For any measurable set X, $H^1(X) = \mu(X)$.*

Proof. By definition $\mu(X) = \inf\{\sum_i |I_i| : \{I_i : i \in \mathbb{N}\}$ is a set of intervals covering $X\}$. We can refine any such cover to an n-cover of X and this will not change the value. Hence for any n, $\mu(X) = \inf\{\sum_i |I_i| : \{I_i : i \in \mathbb{N}\}$ is an n-cover of $X\}$. This equals $H_n^1(X)$, and hence since this is true for any n, we see that $H^1(X) = \mu(X)$. □

Thus by definition of Hausdorff dimension, if $\mu(X) \neq 0$, then $\dim(X) = 1$, so that (i) holds. To see that (ii) holds, take any $s > \dim(B)$. Then since any n-cover of B is an n-cover of B, we see that

$$H_n^s(A) \leqslant H_n^s(B),$$

for all n, and since this is true for all n and $s > \dim(B)$, we have that for all $s > \dim(B)$, $H^s(A) = 0$. Hence (ii) holds. (iii) is proven by similar manipulations of covers.

17.2 Orders

The first person to really look at effectivizing Hausdorff dimension was Lutz in the arena of complexity theory, and there he used generalizations of martingales. This idea was, however, in some sense implicit in the work of Schnorr who looked at *orders* in martingales which are the growth rates, in the same way that the s factor affects the growth rate for the measure of the covers.

Definition 17.2.1 (Schnorr [264, 265]). An *order* is a nondecreasing function $h : \mathbb{N} \mapsto \mathbb{N}$. If F is a martingale and h is an order the *h-success* set of F is the set:

$$S_h(F) = \{\alpha : \limsup_{n\to\infty} \frac{F(\alpha \restriction n)}{h(n)} = \infty.$$

Theorem 10.4.4 can be rephrased that *A real α is Schnorr random iff for all computable orders h and all computable martingales F, $\alpha \notin S_h(F)$.* The theory of Hausdorff dimension can be easily phrased in terms of orders on martingales. Lutz [190, 192, 193] defined his randomness and effective dimension notions in terms of what he called s-gales.

Definition 17.2.2 (Lutz [192, 193]). An *s-gale* is a function $F : 2^{<\omega} \mapsto \mathbb{R}$ such that

$$F(\sigma) = 2^{-s}(F(\sigma 0) + F(\sigma 1)).$$

Similarly we can define s-supergale, etc.

Remember here that a martingale has $s = 1$, whereas here $s < 1$ so we are in a "*hostile environment.*" The basic idea here is that not betting on one outcome or the other is bad. It is like inflation is acting. In a usual martingale, we can decide that we are not prepared to favour one side or the other in our bet. Thus we make $F(\sigma\widehat{\ }i) = F(\sigma)$ at some node σ. In the case of an s-gale, then we will be unable to do this, without *automatically losing money due to inflation.*

Theorem 17.2.3 (Lutz [192]). *For a class X the following are equivalent:*

(i) $\dim(X) = s$.

(ii) $s = \inf\{s \in \mathbb{Q} : X \subseteq S[d]$ *for some s-gale* $F\}$.

(iii) $s = \inf\{s \in \mathbb{Q} : X \subseteq S_{2^{(1-s)n}}[d]$ *for some martingale* $d\}$.

The following proof is drawn from Jan Reimann's notes [244].

Proof. It is easy to see that (ii) and (iii) are equivalent.

For suppose that $\limsup_{n\to\infty} \frac{d(\alpha\restriction n)}{2^{(1-s)n}} \to \infty$. Let $F(\sigma) = 2^{|\sigma|s} d(\sigma)$. Then F evidently, is an s-gale. Note that for any k we have an n such that $\frac{d(\alpha\restriction n)}{2^{(1-s)n}} > k$. Thus $d(\alpha \restriction n) > k2^{(1-s)n}$ and hence

$$F(\alpha \restriction n) = 2^{ns} d(\alpha \restriction n) > k2^{(1-s)n}2^{sn} = 2^n k.$$

Thus $\alpha \in S[F]$. Hence (iii) implies (ii) and the converse is symmetric.

Let X be H^s-null via $\{U_k : k \in \mathbb{N}\}$. That is, $X \subseteq [U_k]$ and $\mu_s(U_k) \leqslant 2^{-k}$, so that $\sum_{\sigma\in U_k} 2^{|\sigma|s} \leqslant 2^{-k}$. For each σ and k, let

$$U_k \restriction [\sigma] = \{\tau : (\tau \in U_k \wedge \sigma \preccurlyeq \tau) \vee (\tau = \sigma \wedge (\exists \nu \in U_k)(\nu \prec \tau))\}.$$

This is a restriction of a given cover to within a cylinder $[\sigma]$. Then, for each k, we define

$$d_k(\sigma) = \frac{\sum_{\tau\in U_k\restriction[\sigma]} 2^{-|\tau|s}}{2^{-|\sigma|}},$$

and

$$d(\sigma) = \sum_{k\in\mathbb{N}} d_k(\sigma).$$

Then each d_k is a supermartingale since

$$d_k(\sigma 0) + d_k(\sigma 1) \leqslant \frac{\sum_{\tau\in U_k\restriction[\sigma]} 2^{-|\tau|s}}{2^{-|\sigma|-1}} = 2d_k(\sigma).$$

Thus d is a supermartingale since for each σ, $d(\sigma) \leqslant \sum_{k\in\mathbb{N}} 2^{|\sigma|-k} < \infty$.

Now suppose that $\alpha \in \cap_k [U_k]$. Then for each k there is a number n_k such that $\alpha \restriction n_k \in U_k$. There are uinfinitely many k for which $n_i \leqslant n_k$ for $i \leqslant k$. For such k, we have the following.

$$\frac{d(\alpha \restriction n_k)}{2^{(1-s)n_k}} \geqslant \frac{\sum_{i\leqslant k} d_k(\sigma)}{2^{(1-s)n_k}} \geqslant k\frac{2^{-n_k s}}{2^{-n_k}} = k.$$

Therefore $\limsup_{n\to\infty} \frac{d(\alpha\restriction n)}{2^{(1-s)n_k}} = \infty$. Hence (i) implies (iii).

Conversely, suppose that d is a martingale satisfying (iii) for X. We can rescale so that $d(\lambda) = 1$. The for each k, the sets

$$U_k = \{\sigma : \frac{d(\sigma)}{2^{(1-s)|\sigma|}} \geqslant 2^k\}$$

induce a cover of X. Using only the shortest σ we can refine U_k and can assume that it is prefix-free. Also we have

$$\sum_{\sigma \in U_k} 2^{-|\sigma|s} \leqslant \sum_{\sigma \in U_k} 2^{-|\sigma|s} \frac{d(\sigma)}{2^{(1-s)|\sigma|}} 2^{-k} = 2^{-k} \sum_{\sigma \in U_k} d(\sigma) 2^{-|\sigma|}.$$

By induction, for every prefix-free set of strings C, $d(\lambda) = \sum_{\sigma \in C} d(\sigma) 2^{-|\sigma|}$, and hence $\{U_k : k \in \mathbb{N}\}$ witnesses the fact that X is H^s-null. □

Lutz [194] makes the following remarks about the characterization above: "Informally speaking, the above theorem says the the dimension of a set is the *most hostile environment* (i.e. most unfavorable payoff schedule, i.e. the infimum s) in which a single betting strategy can *achieve infinite winnings* on every element of the set."

17.3 Effectivizing things

Using the martingale definition we can define a version of effective Hausdorff dimension as follows.

Definition 17.3.1 (Lutz [192])**.** For a complexity class $\mathcal{C}$, we say that R has $\mathcal{C}$ dimension s iff $s = \inf\{s \in \mathbb{Q} : R \subseteq S_{2^{(1-s)n}}[F]$ for some martingale $F \in \mathcal{C}\}$.

Note that Lutz' definition is equivalent to saying that

- $s = \inf\{s \in \mathbb{Q} : R \subseteq S[d]\}$ for some s-gale $d \in \mathcal{C}$.

For example, the Σ_1^0-Hausdorff dimension of X, sometimes called the *constructive* Hausdorff dimension, $\dim_1(R)$, is therefore $\inf\{s \in \mathbb{Q} : R \subseteq S_{2^{(1-s)n}}[F]\}$ where F is the universal computably enumerable supermartingale. We will denote this by $\dim_1(X)$.

There is a fundamental characterization of constructive Hausdorff dimension due to Lutz in terms of algorithmic entropy.

Theorem 17.3.2 (Mayordomo [200],)**.** *The constructive Hausdorff dimension*[1] *of a real α is*

$$\liminf_{n \to \infty} \frac{K(\alpha \upharpoonright n)}{n} = (\liminf_{n \to \infty} \frac{C(\alpha \upharpoonright n)}{n})$$

[1]Staiger [290] showed that Theorem 17.3.2 can be obtained from results of Levin in [332]. Also, there had been a number of earlier results indicating the deep relationships between Hausdorff dimension and Kolmogorov complexity, such as those of Cai and Hartmanis [32] Ryabko [255, 256, 258], and Staiger [288, 289]. These results are discussed in Lutz [193] and Staiger [290]. We particularly refer the reader to Lutz [193], Section 6.

Proof. The C and K statements are the same as the quantities are asymptotically equal. (Recall that $C(x) \leqslant K(x) \leqslant C(x) + 2\log x$.) We will work with K. Note that $\dim_1(\alpha) < s$ is equivalent to saying

$$\limsup_{n\to\infty} \frac{F(\alpha \upharpoonright n)}{2^n 2^{-ns}} = \infty.$$

Since $F(\alpha \upharpoonright n) = 2^n 2^{-K(\alpha\upharpoonright n)} + \mathcal{O}(1)$, this happens iff

$$\limsup \frac{2^{-K(\alpha\upharpoonright n)}}{2^{-ns}} = \infty,$$

and hence iff for all $\varepsilon > 0$ there is an n such that $\frac{K(\alpha\upharpoonright n)}{n} \leqslant s+\varepsilon$, and hence

$$\liminf_{n\to\infty} \frac{K(\alpha \upharpoonright n)}{n} \leqslant s.$$

The conclusion is that

$$\liminf_{n\to\infty} \frac{K(\alpha \upharpoonright n)}{n} \leqslant \dim_1(\alpha).$$

Conversely, suppose that $\liminf_{n\to\infty} \frac{K(\alpha\upharpoonright n)}{n} < s$, and hence there exists infinitely many n with $K(\alpha \upharpoonright n) < ns$. Let

$$D = \cup\{[\sigma] : K(\sigma) < |\sigma|s\}.$$

Clearly D is an n-covering α. We calculate its measure.

$$\sum_{\sigma\in C} 2^{-|\sigma|s} \leqslant \sum \sigma \in C 2^{-K(\sigma)}$$

$$= \sum_{\sigma\in C} 2^{-\log m(\sigma) - \mathcal{O}(1)} \leqslant \mathcal{O}(1) \sum_{\sigma\in C} m(\sigma) < \infty,$$

where m is the discrete universal computably enumerable semimeasure; the second last inequality being the Coding Theorem.

By definition, it follows that $\dim_1(\alpha) \leqslant \liminf_{n\to\infty} \frac{K(\alpha\upharpoonright n)}{n}$. □

Schnorr looked at exponential orders on martingales, and hence, as a consequence of Theorem 17.2.3, in some sense, he was *implicitly* looking at s-randomness. We remark that there is no concrete reference in Schnorr's book to either Hausdorff or dimension[2]. After introducing orders of growth of martingales he places special emphasis on exponential orders. The question here is why he did this, and whether he might have been inspired by the theory of Hausdorff. Chapter 17 is titled "Die Zufallsgesetze von exponentieller Ordnung" (The statistical laws of exponential order) and it starts as follows:

[2]Thanks to Sebastiaan Terwijn for the following insightful comments on Schnorr's work.

"Nach unserer Vorstellung entsprechen den wichtigsten Zufallsgesetzen Nullmengen mit schnell wachsenden Ordnungsfunktionen. Die exponentiell wachsenden Ordnungsfunktionen sind hierbei von besonderer Bedeutung[3]"

Satz 17.1 then says that for any measure zero set A the following are equivalent:

- There are a computable martingale F and $a > 1$ such that A is contained in $\{X : \limsup_n \frac{F(X\upharpoonright n)}{a^n} > 0\}$.
- There are a Schnorr test U_n and computable $b > 0$ such that A is contained in $\{X : \limsup_n m_U(X \upharpoonright n) - b \upharpoonright n > 0\}$.

Here m_U is a "Niveaufunktion" from strings to numbers defined as

$$m_U(\sigma) = \min\{n : \sigma \in U_n\}.$$

Note that this is a test-characterization saying something about the speed of reals being covered, just as in Hausdorff dimension.

17.4 s-Randomness

In this section, we will look at generalizing the notion of 1-randomness along the lines we have used for effective Hausdorff dimension. It turns out that there are at least two possible version of, say, s-Martin-Löf randomness as we now see. First we might base the definition upon a straightforward generalization of the original definition.

Definition 17.4.1 (Tadaki [299]). (i) A *weak s-Martin Löf test* is a computable collection of uniformly c.e. open sets V_k for $k \in \mathbb{N}$ such that for all k, $\mu_s(V_k) \leqslant 2^{-k}$.

(ii) We say that α is *weakly s -Martin Löf random* iff for all weak s-Martin Löf tests $\{V_k : k \in \mathbb{N}\}$, $\alpha \notin \cap_{k\in\mathbb{N}} V_k$.

We can also define a real α to be weakly Chaitin s-random iff $K(\alpha \upharpoonright n) \geqslant sn - \mathcal{O}(1)$, for all n. The analog of Schnorr's theorem that Chaitin random is the same as Martin-Löf random is straightforward.

Theorem 17.4.2 (Tadaki [299]). *α is weakly s-Martin Löf random iff α is weakly Chaitin s-random.*

Proof. This directly mimics the proof of Theorem 9.3.9. Suppose that α is weakly s-Martin Löf random. Let $V_k = \cup\{[\sigma] : K(\sigma) \leqslant s|\sigma| - k\}$. Then, by Chaitin's Counting Theorem, $\mu_s(V_k) \leqslant 2^{-k}$, and hence $\alpha \notin \cap_k V_k$, so that for some k, $K(\alpha \upharpoonright n) \geqslant sn - k$.

[3]To our opinion the important statistical laws correspond to null sets with fast growing orders. Here the exponentially growing orders are of special significance.

Conversely, suppose that α is not weakly s-Martin Löf random, and again we have $\alpha \in \cap_k V_{k^2}$ for some weak s-Martin Löf Martin-Löf test. As usual we enumerate a axiom $|\sigma| - k, \sigma$ if we see $[\sigma] \in V_{k^2}$. □

Armed with Tadaki's characterization, we can emulate the proof that Ω is Chaitin random, to show the following:

Theorem 17.4.3 (Tadaki [299]). *Let $0 < s \leqslant 1$. Then if*

$$\Omega^s = \sum_{U(\sigma)\downarrow} 2^{-s^{-1}|\sigma|},$$

then Ω^s is weakly s random.

Similarly, we can construct a universal weak s- Martin Löf test, etc.

Actually this notion squares with our intuition that if $\alpha = a_1 a_2 \ldots$ is random then $\beta = a_1 0 a_2 0 a_2 \ldots$ should be "reasonably" random. Indeed it is $\frac{1}{2}$-random. To see this, suppose that $K(\beta \upharpoonright n) < \frac{1}{2}n - \mathcal{O}(1)$ infinitely often. Consider the prefix free machine M which, on input σ emulates $U(\sigma)$, and, if $U(\sigma) \downarrow$ is is of the form $b_1 0 b_2 0 \ldots b_n$, then M outputs $b_1 \ldots b_n$. Notice that since $K(\beta \upharpoonright n) < \frac{1}{2}n - \mathcal{O}(1)$ infinitely often, $K_M(\beta \upharpoonright k) < k - \mathcal{O}(1)$ infinitely often, and hence β is not random. Similar reasoning applied via martingales was used by Lutz.

Corollary 17.4.4 (Lutz [192]). *(i) For each $s \in \mathbb{Q}$ with $0 < s \leqslant 1$, there is a real of constructive Hausdorff dimension exactly s*

(ii) Indeed, let $s = \frac{s_1}{s_2}$, ($s_2 < s_1$) and define

$$\widehat{\Omega}^s = a_1 a_2 \ldots a_{s_1} 0^{s_2 - s_1} a_{s_1+1} a_{s_1+2} \ldots a_{2s_1} 0^{s_2 - s_1} a_{2s_1+1} \ldots,$$

with $\Omega = a_1 a_2 a_3 \ldots$, then $\widehat{\Omega}^s$ is weakly s-random, and $dim_1(\Omega^s) = s$.

Proof. The proof outlined in the preceding paragraph demonstrates that $\widehat{\Omega}^s$ is weakly s-random. Thus

$$\frac{K(\widehat{\Omega}^s \upharpoonright n)}{n} \to s,$$

as required by the Mayordomo characterization Theorem 17.3.2. □

Actually, with a little more care we can even prove this for computable reals s, using a variation on Ω^s.

Returning to analogs of the basic theorems characterizing 1-randomness in terms of the three paradigms, we run into a little more trouble when we try to use the dimension/martingale version. Now we could define s-randomness as follows. Recall that Schnorr proved that a real is 1-random iff no c.e. martingale succeeded on α. Now we want to say that no c.e. martingale quickly succeeds on α.

Definition 17.4.5 (Lutz)**.** We say that α is *martingale s random* iff for all c.e. martingales F, $\alpha \notin S_{2^{(1-s)n}}[F]$. Equivalently, no c.e. s-(super) gale succeeds on α.

We would like to follow our basic results that say that martingale randomness, test set randomness and incompressibility all coincide. Unfortunately, the proof breaks down for the martingale case. Consider the proof that if a real is Martin-Löf random then no c.e. (super-)martingale can succeed. We will be given some c.e. martingale, and when we see $F(\sigma) > 2^k$ we would put σ in V_k. Now imagine we follow the same proof for the s case. The problem is that F is only c.e.. We might see $F_s(\sigma 0) > 2^k$ and put $[\sigma 0]$ into V_k. At some later stage $t > s$, we might then see $F_t(\sigma) > 2^k$. We would like to put $[\sigma]$ into V_k, but need to keep the set prefix-free. The point is that in the normal case we can do this by putting $[\sigma 1]$ into V_k. But in the s case, $2(2^{-s(|\sigma|+1)})$ might be much bigger than $2^{-s|\sigma|}$.

This problem was overcome by the following definition of Calude, Staiger and Terwijn [38].

Definition 17.4.6 (Calude, Staiger, Terwijn [38])**.** (i) A *strong s-Martin Löf test* is a computable collection of c.e. sets of strings $\{V_k : k \in \mathbb{N}\}$ (n.b. *not necessarily prefix free*) such that for all prefix free subsets $\widehat{V}_k \subseteq V_k$,

$$\sum_{\sigma \in \widehat{V}_k} 2^{-s|\sigma|} \leqslant 2^{-k}.$$

(ii) We say that α is *strongly* s Martin Löf random iff $\alpha \notin \cap_k V_k$ where $\sigma \in V_k$ is identified with $[\sigma]$ by abuse of notation), for all strong Martin Löf tests $\{V_k : k \in \mathbb{N}\}$.

Notice that there is no import in saying that all prefix free subsets of some V have measure $\leqslant 1$, but there is in saying that they all have *s-measure* $\leqslant 1$. Now things work as usual.

Theorem 17.4.7 (Calude, Staiger, Terwijn)**.** *Fix s with $0 < s \leqslant 1$. Then a real α is strongly s Martin Löf random iff no c.e. s-(super) gale succeeds on α.*

The proof is to mimic Schnorr's proof. the point is that we avoid the aforementioned problem since when $F_t(\sigma) > 2^k$, we can put σ into V_k, *without* putting $[\sigma 1]$ in. We remark that the problem only comes from the fact that F is c.e.. For Schnorr and computable randomness the s-notions for weak and strong are the same.

The reader should note that now we have another problem. We have a notion of randomness and now we lack a machine version. We remedy this as follows.

Definition 17.4.8 (Downey and Reid [251])**.** (i) Lef $f : 2^{<\omega} \mapsto 2^{<\omega}$ be a partial computable function computed by a Turing machine M, and

$Q \subseteq \mathrm{ra}(f)$. Then we define the M-pullback of Q as a subset Q^* of $\mathrm{dom}(f)$ satisfying the following.

(a) $f(Q^*) = Q$.
(b) If $x \in Q^*$, and $f(z) = f(x)$, then $|z| \geqslant |x|$.
(c) If $x \in Q^*$ and $f(z) = f(x)$ with $|z| = |x|$ then x occurs in the domain of M before z does[4]. (We assume that elements occur in the domain of M one per step.)

(ii) We say that a machine M is s-measureable iff for all $Q \subseteq \mathrm{ra}(M)$, $\mu_s(Q^*) \leqslant 1$.

Theorem 17.4.9 (Downey, Reid, Terwijn [251]). *A real α is strongly s-Martin Löf random iff for all s-measurable machines M and all n,*

$$K_M(\alpha \upharpoonright n) \geqslant n - \mathcal{O}(1).$$

Before we prove Theorem 17.4.9, we need a lemma. This lemma takes the role of Kraft-Chaitin in the present setting.

Lemma 17.4.10. *If we have an a strong s-test $\{U_k : k \in \mathbb{N}\}$, then there is an s-measureable machine M that maps strings of length $\sigma_{2k+2,i} - (k+1)$ to $\sigma_{2k+2,i}$, where $U_k = \{\sigma_{k,i} : i \in W_k\}$.*

Proof. Certainly, the hypotheses ensure that $2^{-s|\sigma_{2k+2,i}|} \leqslant 2^{-(-k+2)}$. Hence $s|\sigma_{2k+2,i}| \geqslant k+2$. Therefore $|\sigma_{2k+2,i} \geqslant 2k+2$ as $s \leqslant 1$. We conclude that $|\sigma_{2k+2,i}| - (k+1) > 0$. Therefore requiring that a string of length $|\sigma_{2k+2,i}| - (k+1)$ be sent to $\sigma_{2k+2,i}$ does not result in an absurdity. Second, we must ensure that there are not more than 2^t strings of length t needed for the domain of M. By the definition of M, for all $k > t-1$, we have $|\sigma_{2k+2,i}| - (k+1) > t$. Thus we only need worry about U_{2k+2} with $k \leqslant t-1$.

In this case, strings which require a domain element of length t must have $|\sigma_{2k+2,i}| - (k+1) = t$, and hence be strings in U_{2k+2} of length $t+k+1$.

Let $\#(S,t)$ denote the number of strings of length t in S of length t. As the set of all strings in U_{2k+2} of length $t+k+1$ will form a prefix-free set,

$$\sum\{2^{-s(t+k+1)} : \sigma \in U_{2k+2} \wedge |\sigma| = t+k+1\} \leqslant 2^{-(2k+2)},$$

$$\#(U_{2k+2}, t+k+1)2^{-s(t+k+1)} \leqslant 2^{-(2k+2)},$$

and hence

$$\#(U_{2k+2}, t+k+1) \leqslant 2^{s(t+k+1)-(2k+2)} \leqslant 2^{(t+k+1)-(2k+2)} \leqslant 2^{-t-k-1}.$$

Therefore,

$$\#(\mathrm{dom}(M), t) \leqslant \sum_{k<t} \#(U_{2k+2}, t+k+1) \leqslant \sum_{k<t} 2^{t-k-1} \leqslant 2^{t-1} \sum_{k<t} 2^{-k} \leqslant 2^t.$$

[4] Roughly speaking, this definition is an approximation to the notion of x^* for Kolmogorov complexity introduced in Chapter 6.

Finally, we need to show that M is a s-measureable machine. Let $Q \subseteq \rangle(M)$ be prefix-free. Now

$$Q \subseteq \cup_{k\in\mathbb{N}} U_{2k+2},$$

alowing us to define $Q_k = U_{2k+2} \cap Q$, a prefix free subset of U_{2k+2}. Consider Q_k^*. Then

$$\mu_s(Q_k^*) = \sum_{\sigma\in Q_k} 2^{-s|\sigma|}2^{s(k+1)} = 2^{s(k+1)}\mu_s(Q_k)$$

$$\leqslant 2^{s(k+1)}2^{-(2k+2)} \leqslant 2^{-k-1}.$$

The because Q_k is prefix-free, and $s \leqslant 1$, it follows that

$$\mu_s(Q^*) \leqslant \mu_s(\cup_k Q_k^*) \leqslant \sum_{k\in\mathbb{N}} \mu_s(Q_k^*) \leqslant \sum_{k\in\mathbb{N}} 2^{-k-1} \leqslant 1.$$

□

Proof. (Of Theorem 17.4.9) First suppose we have an s-measureable machine M with

$$\forall c \exists n (K_M(\alpha \upharpoonright n) < n - c.$$

As one would expect, we can define

$$U_k = \{\sigma : H_M(\sigma) < |\sigma| - \frac{k}{s}\}.$$

These are computable enumerable sets and $\alpha \in U_k$ for all k. Now we need a calculation to show that the s-measures work out so that the U_k form a strong s-test.

Let $Q \subset U_k$ be a prefix-free subset. By definition, Q^* is defined. Furthermore,

$$\mu_s(Q) = \sum_{\sigma\in Q} 2^{-s|\sigma|} \leqslant \sum_{\sigma\in Q} 2^{s(K_M(\sigma)-\frac{k}{s})}$$

$$\leqslant \sum_{\sigma\in Q} 2^{-k}2^{-sK_M(\sigma)} \leqslant 2^{-k}\mu_s(Q^*) \leqslant 2^{-k}.$$

The last inequality as Q is prefix-free and s-measureable.

Conversely, if α is not strongly s-random, then there is a strong s-test with

$$\alpha \in \cap_k U_k.$$

Lemma 17.4.10 ensures that there is a machine M that maps strings of length $|\sigma_{2k+2,i}| - (k+1)$ to $\sigma_{2k+2,i}$ where $U_k = \{\sigma_{k,i} : i \in W_k\}$. This ensures that for all d there is a j with

$$K_M(\alpha \upharpoonright d) < d - j.$$

□

As usual, there is both a universal s-test and a universal optimal s-gale.

We remark that the two notions of randomness are different as we now prove.

As a final remark for this section we point out that we know of no characterization of a real being strongly s random *purely* in terms of the initial segment complexity measured relative to U alone.

We remark that these s notions are distinct from

17.5 Hausdorff dimension, partial randomness and degrees

There have been a lot of results looking at the dimensions of classes of relevance to classical computability. We will finish this chapter by looking at some of these. Immediately, we notice a fundamental difference between the degrees of random sets and those of (e.g.) dimension 1 sets.

Lemma 17.5.1. *Suppose that α has dimension s. Then if $\alpha \leqslant_m B$ for some set B, then $B \equiv_m \beta$ with β having dimension s.*

Proof. Choose some rapidly growing computable function, such as f being Ackermann's function. For any set X, let $\alpha[f(X)] = \alpha_X$ be the result of replacing the $f(n)$'th bit of α by 1 if $n \in X$ and 0 if $n \notin X$. Then since for all m, $K(\alpha \upharpoonright m)$ is very close to $K(\alpha_X \upharpoonright m)$ (the difference being close to the inverse of Ackermann's function), we see that $\dim_1(\alpha_X) = \dim_1(\alpha)$. Now we can get the theorem by choosing $X = B$. □

As we know Kučera (Corollary 11.5.4) [155] proved that the Turing degrees of 1-random reals in *not* closed upwards. This is true even in the Δ_2^0 degrees. Hence we have the following.

Corollary 17.5.2 (Folklore)**.** *There are Δ_2^0 degrees of dimension 1 containing no 1-random reals.*

Proof. Choose some 1-random $\mathbf{a} < \mathbf{0}'$ such that it has $\mathbf{b}$ with $\mathbf{a} < \mathbf{b} < \mathbf{0}'$ and $\mathbf{b}$ not 1-random. □

Actually, since $\Omega \leqslant_{tt} \emptyset'$, the method above also gives an interesting contrast to Demuth's Theorem, as follows.

Corollary 17.5.3 (Reimann [244])**.** *$\{B : B \equiv_{tt} \emptyset'\}$ has constructive dimension 1.*

The point here is that Demuth's Theorem, Theorem 9.10.1, stated that if $B \leqslant_{tt} A$ with A random, then $B \equiv_{\widehat{B}}$ with $\widehat{B}$ random. Corollary 17.5.3

says that this result cannot be true with "Σ^0_1 dimension 1" in place of 1-random, since every c.e. set is m-reducible to $\emptyset'$ and Corollary 17.5.8 below says that no wtt-incomplete set can have non-zero dimension.

These considerations lead to two fundamental and open questions:

The only way we have seen to make a real of dimension 1 that is *not* random is essentially to take a 1-random real and "mess it up." The question is, *is the only way that we can really do this is?*. If so, are we able to computationally recover the original real? We formalize this question:

Question 17.5.4. *Suppose that α has Σ_1 dimesion 1. Is there a 1-random $\beta \leqslant_T \alpha$?*

The second question is based around the fact that the only way we have seen to make a, say, $\frac{1}{2}$-random real is to take a random real and "thin it out".

Question 17.5.5. *Is there a Turing degree* **a** *of Σ^0_1-dimension $s \notin \{0,1\}$. What about in the Δ^0_2 degrees?*

These questions have attracted a lot of attention, yet no real success. We will briefly look at what is known. We have seen that 1-random degrees are diagonally noncomputable in that they can compute fixed point free functions. (Theorem 11.5.1) In his thesis, Terwijn proved that this is also true of reals with nonzero dimension.

Theorem 17.5.6 (Terwijn [303])**.** *Suppose that $\dim_1(A) > 0$. Then A can compute a fixed point free function.*

Proof. For each e let e_0 and e_1 denote indices chosen in some canonical way so that $W_e = W_{e_0} \oplus W_{e_1}$. For $0 < s1$, define $f(n) = ns^{-1}$, and

$$B_{e,n} = \{\sigma : |\sigma| = f(n) \wedge \exists s(W_{e_0,s} \restriction f(n) = \sigma \wedge W_{e_1,s} \restriction f(n) = \overline{\sigma})\},$$

where $\overline{\sigma}$ is the complement of σ. Note the $B_{e,n}$ is an n cover since $|\sigma| = f(n) \geqslant n$, and

$$\mu_s(B_{e,n} = \sum_{\sigma \in B_{e,n}} 2^{-s|\sigma|} \leqslant 2^{-sf(n)} = 2^{-n}.$$

Thus $B_{e,n}$ is a s-test.

Now suppose that $\dim_1(A) > s$. Then, if $\{U^s_k : k \in \mathbb{N}\}$ is the universal s-Martin-Löf test, there is a k such that $A \notin U^s_k$. For each e we can compute $g(e) > k$ such that $\varphi_{g(e)}(j)$ is an index of $B_{e,j}$. Since $\{B_{e,j} : j \in \mathbb{N}\}$ form a s-Martin-Löf test, we have that $B_{e,g(e)} \subseteq U^s_{g(e)} \subset U^s_k$, and hence $A \notin B_{e,g(e)}$.

To construct the fixed point free function $h \leqslant_T A$, given e, let $h(e)$ be an index so that $W_{h(e)_0} = A \restriction f(g(e))$ and $W_{h(e)_1} = \overline{A} \restriction f(g(e))$. Then $W_{h(e)} \neq W_e$ for every e, for if $W_{h(e)} = W_e$, then $A \restriction f(g(e)) \in B_{e,g(e)}$. $\square$

Corollary 17.5.7 (Terwijn [303])**.** *Suppose that A is c.e. and $A <_T \emptyset'$. Then $\{B : B \leqslant_T A\}$ has Σ^0_1 dimension 0.*

Proof. If $\{B : B \leqslant_T A\}$ does not have Σ_1^0 dimension 0, then A can compute a fixed point free function be Theorem 17.5.6, and hence A is Turing complete by Arslanov's Completeness Criterion, Theorem 5.19.3. $\square$

Actually, the proofs above give weak truth table reductions, so you also obtain the following corollary for free.

Corollary 17.5.8. *Suppose that A is c.e. and $A <_{wtt} \emptyset'$. Then $\{B : B \leqslant_{wtt} A\}$ has Σ_1^0 dimension 0.*

The result above suggests an approach to the basic questions. If you could show that being strong enough to compute a DNR function was strong enough to prove the existence of a Martin-Löf random real then you'd be done. However, recently, Ambos-Spies, Kjos-Hanssen, Lempp and Slaman showed using proof-theoretical methods that there is an ideal in the Turing degrees containing a DNR degree but containing no 1-random degree. Indeed, they construct an ideal I in the Turing degrees such that for every X in the ideal there is a G in the ideal which is diagonally nonrecursive relative to X and such that there is no 1-random degree in I. We refer the reader to [7]

We do know that Question **??** has a positive solution for m-reducibility.

Theorem 17.5.9 (Reimann and Terwijn [247])**.** *There exists a Δ_2^0 set A such that $\{B : B \leqslant_m A\}$ has Σ_1^0 Hausdorff dimension $s \notin \{0, 1\}$.*

Proof. Take a random C, and convert it into a non-random real as follows. Takew the bits of C and in order, for each adjacent pairs, multiply together and make a new bit. Hence 10010110.. would become 000010... Then make every 0 a 1 and every 1 a zero, to create a new real A. This real A has bias $\frac{1}{3}$ towards 1, and by betting $\frac{2}{3}$ of our capital on 1's in a martingale, we see that its effective Hausdorff dimension (even computable Hausdorff dimension) is bounded by $\frac{1}{3}$.

We claim that if $B \leqslant_m A$ then $\dim_{\Sigma_1^0}(B) \leqslant \mathcal{H}(\frac{1}{3})$.

It then follows that $0 \neq \dim(\{B : B \leqslant_m A\}) \leqslant (\frac{1}{3})$, so in particular unequal to 0 or 1.

Let $B \leqslant_m A$ via f. We define a Σ_1^0-martingale d that succeeds on B as follows:

- If $f(x)$ is a fresh query, bet $\frac{1}{3}$rd on 0 and $\frac{2}{3}$rd on 1.
- If $f(x)$ is not fresh, this means that it has been queried before, so we can let d double its value because we can look up the previous answer in B.

Now the set of fresh queries is a KL-selected subset Q of A, which then has frequency $(\frac{1}{3}, \frac{2}{3})$. So on the corresponding bits in B, d will grow at least as fast. On the remaining bits d doubles, so there it grows even faster. $\square$

The same result could be obtained as follows. Take two reals A and C with $A \oplus C$ random. Let $A = a_0 a_1 \dots$ and $C = c_0 c_1 \dots$. Create a new real $\widehat{A} = d_0 d_1 d_2 d_3 \dots$, where one of d_{2i}, d_{2i+1} is a_i, and the other is definitely 1. This is chosen by C. If $c_i = 1$, we choose position $2i$, and if $c_i = 0$ then we choose $2i + 1$.

The best partial result towards resolving the question of broken dimension, is the following result of Nies and Reimann [**?**].

Theorem 17.5.10 (Nies and Reimann [**?**])**.** *For any computable real α with $0 < \alpha < 1$ there is a Δ^0_2 real A such that the effective Hausdorff dimension of $\{B : B \leqslant_{wtt} A\}$ is α.*

Proof. TO ADD □

17.6 Other Notions of Dimension

Hausdorff's extension of Carathéodory's s-dimensional measure is certinaly not the only such generalization of the notion of dimension to non-integer values in geometric measure theory and fractal geometry. We will briefly look at a couple of further dimension concepts, especially those that are important in context of the topics dealt with here. Further details can be found in Federer [**?**].

17.6.1 Box counting dimension

The definition of H^s_n allows only sets of diameter less or equal 2^{-n} in a covering. For $C \subseteq 2^\omega$, let

$$C \upharpoonright n = \{w \in 2^{<\omega} : \exists \alpha \in C (w \prec \alpha)\}.$$

Define the *upper* and *lower box counting dimension* of C as

$$\overline{\dim}_B C = \limsup_{n\to\infty} \frac{\log(|C \upharpoonright n|)}{n} \quad \text{and} \quad \underline{\dim}_B C = \liminf_{n\to\infty} \frac{\log(|C \upharpoonright n|)}{n}. \tag{17.1}$$

If $\overline{\dim}_B$ and $\underline{\dim}_B$ coincide, than this value is simply called the *box counting dimension*, sometimes also *Minkowski dimension* of C. The name box counting comes from the fact that, for each level 2^{-n}, one simply counts the number of boxes needed to cover C. The following is clear.

Lemma 17.6.1. *For any set $C \subseteq 2^\omega$,*

$$\dim_H C \leqslant \underline{\dim}_B C \leqslant \overline{\dim}_B C. \tag{17.2}$$

(Lower) box counting dimension gives an easy upper bound on Hausdorff dimension, although this estimate may not be very exact. For instance, for $\mathbb{Q} \cap [0, 1]$, identified with the set of all infinite binary sequence which are 0 from some point on, we have $0 = \dim_H(\mathbb{Q} \cap [0,1]) < \underline{\dim}_B(\mathbb{Q} \cap [0,1]) = 1$. In fact, this holds for any dense subset of 2^ω. This shows that, in general, box counting dimension is not a stable concept of dimension. Staiger [**?**, 289, 290] has investigated some conditions when Hausdorff and box counting dimension coincide. Probably the most famous example of such sets arises in the context of dynamical systems.

One can modify box counting dimension to obtain a countably stable notion. This yields the concept of *modified box counting dimension*, denoted $\dim_{MB}$, defined as follows:

$$\overline{\dim}_{MB} X = \inf\left\{\sup_i \underline{\dim}_B X_i : X \subseteq \bigcup_{i\in\mathbb{N}} X_i\right\}, \tag{17.3}$$

$$\underline{\dim}_{MB} X = \inf\left\{\sup_i \overline{\dim}_B X_i : X \subseteq \bigcup_{i\in\mathbb{N}} X_i\right\}, \tag{17.4}$$

(That is, we split up a set into countable many parts and look at the dimension of its 'most complicated' part. Then we optimize this by looking for the part with the lowest such 'overall' dimension.)

The modified box counting dimensions behave more stable as their original counterparts, in particular all countable have dimension zero now. However, they are usually hard to calculate, due to the extra inf / sup process involved.

17.6.2 Effective box counting dimension

The effectivization of box counting dimension is due to Jan Reimann, as is its coincidence with effective packing dimension which we see in the next section.

We will only be concerneed with the dimension of reals and hence we will not look at the modified versions of box counting dimension needed for countable stability. We begin with the easy observation of Kolmogorov [151], which we met in Theorem 6.2.2, (ii).

Proposition 17.6.2 (Kolmogorov [151])**.** *Let $A \subseteq \mathbb{N} \times 2^{<\omega}$ be computably enumerable. Suppose $A_m = \{x : (m,x) \in A\}$ is finite. Then, for some constant c and for all $x \in A_m$,*

$$C(x|m) \leqslant \log |A_m| + c.$$

We may use this as a starting point to define effective box counting dimension.

Definition 17.6.3 (Reimann [244])**.** Given a sequence $\beta \in 2^\omega$, call an r. e. set $C \subseteq 2^{<\omega}$ a *effective box cover* (or, if the effective context is clear, just *box cover*) of β, if

$$(\overset{\infty}{\forall} n)[\beta \upharpoonright n \in D].$$

Effective box counting dimension measures how efficient the initial segments of a sequence can be 'enwrapped' in an r.e. set of strings. We fix the following notation: Given a set $D \subseteq 2^{<\omega}$, let

$$D^{[n]} = \{w \in D : |w| = n\}.$$

Definition 17.6.4 (Reimann [244])**.** For a sequence $\beta \in 2^\omega$, we define the *effective lower* and *upper box counting dimension* as

$$\begin{aligned}
\underline{\dim}_B^1 \beta &= \inf\left\{\liminf_{n\to\infty} \frac{\log|C^{[n]}|}{n} : \ C \text{ is an effective box cover of } \beta\right\},\\
\overline{\dim}_B^1 \beta &= \inf\left\{\limsup_{n\to\infty} \frac{\log|C^{[n]}|}{n} : \ C \text{ is an effective box cover of } \beta\right\}.
\end{aligned}$$

It is a trivial observation that, as in the classical case, effective box counting dimension always bounds effective Hausdorff dimension from above, in particular $\dim_1\beta \leqslant \underline{\dim}_B^1\beta$ for any $\beta \in 2^\omega$.

Concerning lower box counting dimension, an effective characterizion is possible in terms of Kolmogorov complexity.

Theorem 17.6.5. *Let $C \subseteq 2^\omega$ be a shift-invariant Π_1^0-class. Then*

$$(\forall \beta \in C)\, \dim_1\beta = \lim_{n\to\infty} \frac{|C \upharpoonright n|}{n} \tag{17.5}$$

For upper algorithmic entropy, a close connection between Kolmogorov complexity and effective box counting dimension holds.

Theorem 17.6.6. *For any sequence $\beta \in 2^\omega$,*

$$\overline{\dim}_B^1\beta = \limsup_{n\to\infty} \frac{K(\beta \upharpoonright n)}{n}.$$

Note that, due to the asymtotic equivalence of plain and prefix complexity, it does not matter which version of complexity we use.

Proof. ($\leqslant$) Assume $\overline{\dim}_B^1\beta < s$. It follows immeadiately from Proposition 17.6.2 that $\limsup_{n\to\infty} \frac{K(\beta\upharpoonright n)}{n} \leqslant s$.

($\geqslant$) Suppose now $\limsup_{n\to\infty} \frac{K(\beta\upharpoonright n)}{n} < s$. We show this implies $\overline{\dim}_B^1\beta \leqslant s$. Define an r.e. set D by letting

$$D = \{w \in 2^{<\omega} : C(w) < s|w|\}.$$

By assumption, D is a box cover of β. An easy combinatorial argument yields that the number of programms of length less than sn is less than $2^{sn} - 1$. Hence

$$D^{[n]} \leqslant 2^{sn},$$

and therefore

$$\overline{\dim}_B^1\beta \leqslant \limsup_{n\to\infty} \frac{|D^{[n]}|}{n} = s,$$

which completes the proof. □

17.6.3 Packing measures and packing dimension

Another classically important fractional dimension is called packing dimension. The effectivization of this dimension was first considered by Athreya, Hitchcock, Lutz, and Mayordomo [15],but we will follow the nice account of Jan Reimann [244], who additionally noted the connections with the box counting dimension of the previous section. We remark that the *classical* characterization of packing dimension in terms of strong success of s-gales (due to Athreya, K., J. Hitchcock, J. Lutz, and E. Mayordomo) came as quite a surprise to the workers in the area, who had thought packing dimension was too complex to have such a simple characterization.

Packing dimension can be seen as a dual to Hausdorff dimension. Hausdorff dimension are defined in terms of economical coverings, that is, enclosing a set from outside, packing measures approximate from the inside, by packing it economically with disjoint sets of small size.

For this purpose, we say that a prefix free set $P \subseteq 2^{<\omega}$ is a *packing* in $X \subseteq 2^{\omega}$, if for every $\sigma \in P$, $\sigma \prec X$. Geometrically speaking, a packing in X is a collection of open balls with centers in X. If the balls all have radius $\leqslant 2^{-n}$, we call it a n-packing in X.

Now one can try to find a packing as 'dense' as possible: Given $s \geqslant 0$, $n > 0$, let

$$\mathcal{P}^s_n(X) = \sup\left\{\sum_{w\in P} 2^{-|w|s} : \ P \text{ is an } n\text{-packing in } X.\right\}. \tag{17.6}$$

Again, as $\mathcal{P}^s_n(X)$ decreases with n, the limit

$$\mathcal{P}^s_0(X) = \lim_{n\to 0} \mathcal{P}^s_n(X)$$

exists. However, this definition leads to the same problems we encountered with box counting dimension: Taking, for instance, the rational numbers in the unit interval, we can find denser and denser packings yielding that for every $0 \leqslant s < 1$, $\mathcal{P}^s(\mathbb{Q} \cap [0, 1]) = \infty$, hence it lacks countable additivity, in particular it is not a measure. This can be overcome by applying a Carathéodory process to $\mathcal{P}^s_n$. Hence define

$$\mathcal{P}^s(X) = \inf\left\{\sum \mathcal{P}^s_0(X_i) : \ X \subseteq \bigcup_{i\in\mathbb{N}} X_i\right\}. \tag{17.7}$$

(The infimum is taken over arbitrary countable covers of X.) $\mathcal{P}^s$ is an (outer) measure on 2^{ω}, and it is Borel regular. (This needs no longer be true if the dimension function $h(x) = x^s$ is replaced by more irrgelar functions not satisfying even weak continuity requirements.) $\mathcal{P}^s$ is called, in correspondence to Hausdorff measures, the *s-dimensional packing measure* on 2^ω. Packing measures were introduced by Tricot [305] and Sullivan [298]. They can be seen as dual concept to Hausdorff measures, and behave in many ways similar to them. In particular, one may define *packing dimension* in the same way as Hausdorff dimension.

Definition 17.6.7. The *packing dimension* of a set $X \subseteq 2^\omega$ is defined as

$$\dim_{\mathrm{P}} X = \inf\{s : \mathcal{P}^s(X) = 0\} = \sup\{s : \mathcal{P}^s(X) = \infty\}. \tag{17.8}$$

Packing dimension has stability properties similar to Hausdorff dimension, e.g. countable stability. With some effort, one can show that it *coincides with* $\overline{\dim}_{MB}$ see Falconer, [103], Chapter 3]. Generally, the following relations between the different dimension concepts hold: any $X \subseteq 2^\omega$,

$$\dim_{\mathrm{H}} X \leqslant \underline{\dim}_{MB} X \leqslant \overline{\dim}_{MB} X = \dim_{\mathrm{P}} X \leqslant \overline{\dim}_B X. \tag{17.9}$$

As we mentioned in the introduction, whilst the traditional definition of packing dimension is rather complicated due to the additonal decomposition/optimization step, there is a simple martingale characterization. This characterization was discovered by Athreya, Hitchcock, Lutz, and Mayordomo [15]. This characterization demonstrates the dual nature of packing measures and packing dimension much more clearly.

Definition 17.6.8. Given $0 < s \leqslant 1$, a martingale $d : 2^{<\omega} \to [0, \infty)$ *s-succeeds strongly* on a sequence α if

$$\liminf_{n \in \infty} \frac{d(\alpha \upharpoonright n)}{2^{(1-s)n}} = \infty \tag{17.10}$$

Obviously, condition (17.10) is equivalent to $\lim_n d(\alpha \upharpoonright n)/2^{(1-s)n} = \infty$. From a game-theoretical perspective, succeeding strongly means not only to accumulate arbitrary high levels of capital, but also to be able to guarantee that the capital stays above arbitrary high levels from a certain time on.

Theorem 17.6.9 (Athreya, Hitchcock, Lutz, and Mayordomo [15])**.** *For any set $X \subseteq 2^\omega$,*

$$\dim_{\mathrm{P}} X = \inf\{s : \textit{some martingale } d \ s\textit{-succeeds strongly on all } \beta \in X\}.$$

Proof. ($\leqslant$) Suppose some martingale d is strongly s-successful on X. By (17.9) it suffices to show that $\overline{\dim}_{MB} X \leqslant s$. Consider the set of strings

$$D_n = \{|\sigma| = n : d(\sigma) > 2^{(1-s)n}\}.$$

Obviously, every $\beta \in X$ is contained in all but finitely many $[\![D_n]\!]$. Therefore,

$$X \subseteq \bigcup_{i \in \mathbb{N}} \bigcap_{j \geqslant i} [\![D_j]\!].$$

If we let $X_i = \bigcap_{j \geqslant i} [\![D_j]\!]$, it is enough to show that $\overline{\dim}_B X_i \leqslant s$ for all i. Note that $X_i \subseteq [\![D_n]\!]$ for all $n \leqslant i$, hence $|X_i \upharpoonright n| \leqslant |D_n|$. Now it follows from Proposition 17.6.2 that

$$|D_n| \leqslant 2^{ns}.$$

Therefore,

$$\overline{\dim}_B X_i = \limsup_{n\to\infty} \frac{\log |X_i \upharpoonright n|}{n} \leqslant \limsup_{n\to\infty} \frac{\log |D_n|}{n} \leqslant s.$$

($\geqslant$) We may assume $\overline{\dim}_{MB} X < 1$. Let s', s, t be such that $\overline{\dim}_{MB} X < s' < s < t < 1$. We show that there exists a martingale d which is strongly t-successful on X. From the definition of modified box counting dimension we can infer the existence of sets X_i such that $X \subseteq \bigcup_i X_i$ and $\overline{\dim}_B X_i < s'$ for all i.

It follows that, for all i, there exists a number n_i,

$$(\forall n \geqslant n_i)\ \frac{\log |X_i|}{n} < s'.$$

We show that for each i we can find a stongly t-successful martingale for X_i. (This is actually enough: Using additivity of martingales, one could combine these into a single martingale which is $(t+\varepsilon)$-successful on X, with $\varepsilon > 0$ arbitrary small.)

Fix an arbitrary $i \in \mathbb{N}$. Let $X_n = X_i \upharpoonright n$. For each $n \geqslant n_i$, define a martingale d_n (inductively) as follows:

$$d_n(\sigma) = \begin{cases} 2^{|\sigma|}|X_n \upharpoonright [\![\sigma]\!]|2^{-sn}, & \text{if } |\sigma| \leqslant n, \\ 2^{n-|\sigma|}d_n(\sigma \upharpoonright n), & \text{if } |\sigma| > n. \end{cases}$$

Note that for $\sigma \in X_n$, $d_n(\sigma) = 2^{(1-s)n}$. Let $d = \sum_{n \geqslant n_i} d_n$. The finiteness of d follows from

$$d(\varepsilon) = \sum_{n\geqslant n_i} d_n(\varepsilon) = \sum_{n \geqslant n_0} |X_n \upharpoonright [\![\sigma]\!]|2^{-sn} < \sum_{n\geqslant n_i} 2^{(s'-s)n} < \infty.$$

A Finally, if $\beta \in X_i$, we have that $\beta \upharpoonright n \in X_n$ for all n, thus, for $n \geqslant n_i$,

$$\frac{d(\beta \upharpoonright n)}{2^{(1-s)n}} \geqslant \frac{d_n(\beta \upharpoonright n)}{2^{(1-s)n}} \geqslant \frac{2^{(1-s)n}}{2^{(1-t)n}} = 2^{(t-s)n}.$$

As $t > s$, this completes the proof. $\square$

17.6.4 Effective packing dimension

Note that the somewhat involved definition of packing measures (see Section 17.6.3 with the extra optimization renders a direct Martin-Löf style effectivization in terms of enumerable covers difficult. This obstacle can be overcome by using the martingale characterizations of measure zero sets.

In view of Theorem 17.6.9, the definition of effective packing dimension is a straightforward affair.

Definition 17.6.10 (Athreya, Hitchcock, Lutz, and Mayordomo [15]). Given $X \subseteq 2^\omega$, define the *effective packing dimension* of X as

$$\dim^1_{\mathrm{P}} X = \inf\{s : \exists \text{ martingale } d \text{ strongly } s\text{-successful on all } \beta \in X\}.$$

In Section 17.6.3 we stated the fact that packing dimension equals upper modified box counting dimension, see (17.9). However, as regards individual sequences, the modifications to box couting dimension leading to can can disregard the modified version of box counting dimension. Namely, a careful effectivization of the proof of Theorem 17.6.9 yields the following.

Theorem 17.6.11 (Reimann [244]). *For every real $\beta \in 2^\omega$, $\dim^1_{\mathrm{P}} \beta = \overline{\dim}^1_B \beta$.*

Combining this with Theorem 17.6.6 gives an easy proof that effective packing dimension and upper algorithmic entropy coincide.

Corollary 17.6.12 (Athreya, Hitchcock, Lutz, and Mayordomo [15]). *For every sequence $\beta \in 2^\omega$, $\dim^1_{\mathrm{P}} \beta = \limsup_{n\to\infty} \frac{K(\beta\upharpoonright n)}{n}$.*

17.7 Dimensions in other classes

In the past sections, we have concentrated upon Σ^0_1 Hausdorff, packing, and box dimensions. In this section we will look at variations on the theme.

As with randomness concepts, there are maindly two ways to do this. One is to look at, for example, Δ^0_2 dimension etc, by changing the *arithmetical* complexity of the gales used for the definitions. There has been a little work here. We can also look at more restrictive notions such as Schnorr dimension etc. The known relationships can be summarized in the following diagram from [82].

The relations between the various notions are as follows:

$$\begin{array}{ccc}
\Delta^0_2\text{-random} & & \\
\Downarrow & & \\
\text{Schnorr } \Delta^0_2\text{-random} & \Longrightarrow & \Delta^0_2\text{-dimension } 1 \\
\Downarrow & & \Downarrow \\
\Sigma_1\text{-random} & \Longrightarrow & \Sigma_1\text{-dimension } 1 \\
\Downarrow & & \\
\text{computably random} & & \Big\Downarrow \\
\Downarrow & & \\
\text{Schnorr random} & \Longrightarrow & \text{computable dimension } 1
\end{array}$$

No other implications hold than the ones indicated.

We finish this with a brief discussion of Schnorr and computable dimensions which will be of relevance for Chapter 19 when we look at computably enumerable sets.

17.8 Schnorr Null Sets and Schnorr Dimension

17.8.1 Basics

Naturally, the basic idea behind Schnorr dimension is to extend the concept of a Schnorr test to Hausdorff measures and show that an effective version of Theorem 17.1.2 holds. Then the definition of Schnorr Hausdorff dimension follows in a straightforward way.

Definition 17.8.1. Let $s \in [0,1]$ be a rational number.

(a) A *Schnorr s-test* is a uniformly c.e. sequence $(S_n)_{n\in\mathbb{N}}$ of sets of strings which satisfies, for all n, the following conditions:

(1) For all n,

$$\sum_{w\in S_n} 2^{-|w|s} \leqslant 2^{-n}. \tag{17.11}$$

(2) The real number $\sum_{w\in S_n} 2^{-|w|s}$ is computable uniformly in n, that is, there exists a computable function f such that for each n, i, $\left|f(n,i) - \sum_{w\in S_n} 2^{-|w|s}\right| \leqslant 2^{-i}$.

(b) A class $\mathcal{A} \subseteq 2^\omega$ is *Schnorr s-null* if there exists a Schnorr s-test (S_n) such that

$$\mathcal{A} \subseteq \bigcap_{n\in\mathbb{N}} [\![S_n]\!].$$

To be compatible with the conventional notation, we denote the Schnorr 1-nullsets simply as *Schnorr null*. The *Schnorr random* sequences are those which are (as a singleton in 2^ω) not Schnorr null. As we have seen in Chapter 10.2 we any Schnorr test is equivalent to one with the n-th part of measure 2^{-n} and the same argument shows this for s-tests. Note further that, for rational s, the sets S_n in a Schnorr s-test are actually uniformly computable, since to determine whether $w \in S_n$ it suffices to enumerate S_n until the accumulated sum given by $\sum 2^{-|v|s}$ exceeds $2^{-n} - 2^{|w|s}$ (assuming the measure of the n-th level of the test is in fact 2^{-n}). If w has not been enumerated so far, it cannot be in S_n. The converse, however, does not hold: If $W \subseteq 2^{<\omega}$ is computable, this does not necessarily imply that the measure of $[\![W]\!]$ is computable. Again, as we have seen in Chapter 10.2, we can use Schnorr Solovay tests. The s-analog is the following.

Definition 17.8.2. Let $s \in [0,1]$ be rational.

(a) A *Solovay s-test* is a c.e. set $D \subseteq 2^{<\omega}$ such that

$$\sum_{w\in D} 2^{-|w|s} \leqslant 1$$

(b) A Solovay s-test is *total* if

$$\sum_{w\in D} 2^{-|w|s}$$

is a computable real number.

(c) A Solovay s-test D *covers* a sequence $A \in 2^\omega$ if

$$(\exists^\infty w \in D)\,[w \sqsubset A].$$

In this case we also say that A *fails* the test D.

Theorem 17.8.3 (Downey, Merkle, Reimann [91]). *For any rational $s \in [0,1]$, a class $\mathcal{X} \subseteq 2^\omega$ is Schnorr s-null if and only if there is a total Solovay s-test which covers every sequence $A \in \mathcal{X}$.*

Proof. ($\Rightarrow$) Let $\mathcal{X}$ be Schnorr s-null via a test $(U_n)_{n\in\mathbb{N}}$. Let

$$C = \bigcup_{n\geqslant 1} U_n.$$

Obviously, C is a Solovay s-test which covers all of $\mathcal{X}$, so it remains to show that C is total. But in order to compute $c = \sum_{v\in C} 2^{-|v|s}$ with precision 2^{-n}, it suffices to compute, for $i = 1,\dots,n{+}1$, U_i up to precision $2^{-(i+n+1)}$.

($\Leftarrow$) Let C be a total Solovay s-cover of $\mathcal{X}$. Given n, compute $c = \sum_{v\in C} 2^{-|v|s}$ up to precision 2^{-n-2}. Now find a finite subset $\widetilde{C} \subseteq C$ such that

$$c - 2^{-n-1} \leqslant \sum_{w\in\widetilde{C}} 2^{-|w|s} \leqslant c - 2^{-n-2}.$$

Then $C \setminus \widetilde{C}$ covers every sequence $A \in \mathcal{X}$. Furthermore, it holds that $\sum_{w\in C\setminus\widetilde{C}} 2^{-|w|s} \leqslant 3/2^{n+2} \leqslant 1/2^n$. Hence, if we define $U_n = C \setminus \widetilde{C}$, the (U_n) will form a Schnorr s-test for $\mathcal{X}$. $\square$

Note that the equivalence between Solovay and Schnorr s-tests does not extend to Martin-Löf s-tests in general. For a Martin-Löf s-test we only require the first condition in Definition 17.8.1 but not the second one. Martin-Löf s-tests and the corresponding dimension notions have been explicitly studied by [**?**], [**?**], and [**?**]. Implicitly, via martingales, they were already present in Lutz's introduction of effective dimension [**?**]. Solovay showed that a set $\mathcal{X} \subseteq 2^\omega$ is covered by a Martin-Löf 1-test if and only if it is covered by Solovay 1-test. However, [**?**] have recently shown that for any rational $0 < s < 1$ there exists a sequence A which is not Martin-Löf s-null but is covered by a Solovay s-test.

17.8.2 Schnorr dimension

Again we can show that dimension is well-defined, as with Theorem 17.1.2.

Proposition 17.8.4. *Let $\mathcal{X} \subseteq 2^\omega$. Then for any rational $s \geqslant 0$, if $\mathcal{X}$ is Schnorr s-null then it is also Schnorr t-null for any rational $t \geqslant s$.*

Proof. It suffices to show that if $s \leqslant t$, then every Schnorr s-test (S_n) is also a Schnorr t-test. So assume $\{S_n\}$ is a Schnorr s-test. Given any real $\alpha \geqslant 0$ and $l \in \mathbb{N}$, let

$$m_n(\alpha) := \sum_{w \in S_n} 2^{-|w|\alpha} \quad \text{and} \quad m_n^l(\alpha) := \sum_{\substack{w \in S_n \\ |w| \leqslant l}} 2^{-|w|\alpha}.$$

It is easy to check that

$$m_n^l(t) \leqslant m_n(t) \leqslant m_n^l(t) + m_n(s) 2^{(s-t)l}.$$

Now $m_n(s)$ is computable, as is $2^{(s-t)l}$, and $2^{(s-t)l}$ goes to zero as l gets larger. Therefore, we can effectively approximate $m_n(t)$ to any desired degree of precision. □

The definition of Schnorr Hausdorff dimension now follows in a straightforward way.

Definition 17.8.5. The *Schnorr Hausdorff dimension* of a class $\mathcal{X} \subseteq 2^\omega$ is defined as

$$\dim_{\mathrm{H}}^{\mathrm{S}}(\mathcal{X}) = \inf\{s \geqslant 0 : \mathcal{X} \text{ is Schnorr } s\text{-null}\}.$$

For a sequence $A \in 2^\omega$, we write $\dim_{\mathrm{H}}^{\mathrm{S}} A$ for $\dim_{\mathrm{H}}^{\mathrm{S}}\{A\}$ and refer to $\dim_{\mathrm{H}}^{\mathrm{S}} A$ as the Schnorr Hausdorff dimension of A.

Note that the Schnorr Hausdorff dimension of any sequence is at most 1, since for any $\varepsilon > 0$ the "trivial" test $W_n = \{w : |w| = l_n\}$, l_n chosen appropriately, will cover all of 2^ω.

17.8.3 Schnorr Dimension and Martingales

We recall from Chapter 10 the Schnorr's characterization of Schnorr randomness in terms of strong success for a martingale.

Definition 17.8.6. Let $g : \mathbb{N} \to \mathbb{R}$ be a computable order function. A martingale is g-successful on a sequence $B \in 2^\omega$ if

$$d(B\restriction_n) \geqslant g(n) \text{ for infinitely many } n.$$

We recall that in Theorem 10.4.4, we characterized Schnorr randomness in terms of strong success. That is, a set $\mathcal{X} \subseteq 2^\omega$ is Schnorr null if and only if there exists a computable martingale d and a computable order g such that d is g-successful on all $B \in \mathcal{X}$.

A martingale being s-successful means it is g-successful for the order function $g(n) = 2^{(1-s)n}$. These are precisely what Schnorr calls *exponential orders*, so much of effective dimension is already, though apparently without explicit reference, present in Schnorr's treatment of algorithmic randomness. We have already seen in Chapter 10 that the concepts of computable randomness and Schnorr randomness do not coincide. There are Schnorr random sequences on which some computable martingale succeeds. However, the differences vanish if it comes to dimension.

Theorem 17.8.7 (Downey, Merkle, Reimann [91]). *For any sequence $B \in 2^\omega$,*

$$\dim_{\mathrm{H}}^{\mathrm{S}} B = \inf\{s \in \mathbb{Q} : \textit{some computable martingale } d \textit{ is } s\textit{-successful on } B\}.$$

Proof. ($\leqslant$) Suppose a martingale d is s-successful on B. (We may assume that $s < 1$. The case $s = 1$ is trivial.) It suffices to show that for any $1 > t > s$ we can find a Schnorr t-test which covers B.

We define

$$U_k^{(t)} = \left\{\sigma : \frac{d(\sigma)}{2^{(1-t)|\sigma|}} \geqslant 2^k\right\}$$

It is easy to see that the $(U_k^{(t)})_{k\in\mathbb{N}}$ cover B. Since d is computable, the cover is effective. The only thing that is left to prove is that $\sum_{w\in U_k^{(t)}} 2^{-s|w|}$ is a computable real number.

To approximate $\sum_{w\in U_k^{(t)}} 2^{-s|w|}$ within 2^{-r}, effectively find a number n such that $2^{(1-t)n} \geqslant 2^r d(\varepsilon)$. If we enumerate only those strings σ into $U_k^{(t)}$ for which $|\sigma| \leqslant n$, we may conclude for the remaining strings $\tau \in U_k^{(t)}$ that $d(\tau) \geqslant 2^{(1-t)n} 2^k \geqslant 2^{r+k} d(\varepsilon)$.

Now we employ an inequality for martingales, which is sometimes referred to as *Kolmogorov's inequality*, but was first shown by [**?**]. If d is a martingale, then it holds for every $k > 0$,

$$\lambda\{B \in 2^\omega : d(B\upharpoonright_n) \geqslant k \text{ for some } n\} \leqslant \frac{d(\varepsilon)}{k},$$

where λ denotes Lebesgue measure on 2^ω.

Using this inequality we get that the measure induced by the strings not enumerated is at most $2^{-(r+k)}$.

($\geqslant$) Suppose $\dim_{\mathrm{H}}^{\mathrm{S}} B < s < 1$. (Again the case $s = 1$ is trivial.) We show that for any $t > s$, there exists a computable martingale d which is s-successful on B.

Let $(V_k)_{k\in\mathbb{N}}$ be a Schnorr t-test for B.

$$d_k(\sigma) = \begin{cases} 2^{(1-s)|w|} & \text{if } \sigma \sqsupset w \text{ for some } w \in V_k, \\ \sum_{\sigma w \in V_k} 2^{-|w|+(1-s)(|\sigma|+|w|)} & \text{otherwise.} \end{cases}$$

We verify that d_k is a martingale. Given $\sigma \in 2^{<\omega}$, if there is a $w \in V_k$ such that $w \sqsubseteq \sigma$, we have

$$d_k(\sigma 0) + d_k(\sigma 1) = 2^{1+(1-s)|w|} = 2d_k(\sigma).$$

If such w does not exist,

$$\begin{aligned} d_k(\sigma 0) + d_k(\sigma 1) &= \sum_{\sigma 0 w \in V_k} 2^{-|w|+(1-s)(|\sigma|+|w|+1)} + \sum_{\sigma 1 w \in V_k} 2^{-|w|+(1-s)(|\sigma|+|w|+1)} \\ &= \sum_{\sigma v \in V_k} 2^{(-|v|+1)+(1-s)(|\sigma|+|v|)} = 2d_k(\sigma). \end{aligned}$$

Besides, $d_k(\varepsilon) = \sum_{w \in V_k} 2^{-|w|+(1-s)|w|} = \sum_{w \in V_k} 2^{-s|w|} \leqslant 2^{-k}$, so the function

$$d = \sum_k d_k$$

defines a martingale as well (using additivity). Finally, note that, for $w \in V_k$, $d(w) \geqslant d_k(w) = 2^{(1-s)|w|}$. So if $B \in \bigcap_k [\![V_k]\!]$, $d(B \upharpoonright_n) \geqslant 2^{(1-s)n}$ infinitely often, which means that d is s-successful on all $B \in \mathcal{X}$.

Since each $d_k(\varepsilon) \leqslant 2^{-k}$, the computability of d follows easily from the computability of each d_k, which is easily verified based on the fact that the measure of the V_k is uniformly computable. (Note that each σ can be in at most finitely many V_k.) □

Thus, in contrast to randomness, the approach via Schnorr tests and the approach via computable martingales to dimension yield the same concept.

We can build on Theorem 17.8.7 to introduce *Schnorr packing dimension*.

Definition 17.8.8 (Downey, Merkle, Reimann [91])**.** Given a sequence $A \in 2^\omega$, we define the *Schnorr packing dimension* of A, $\dim^{\mathrm{S}}_{\mathrm{P}} A$, as

$$\dim^{\mathrm{S}}_{\mathrm{P}} A = \inf\{s \in \mathbb{Q} : \text{some computable martingale } d \text{ is strongly } s\text{-successful on } A\}$$

This definition of Schnorr packing dimension coincides with the notion $\mathrm{Dim}_{\mathrm{comp}}$ defined before by [15]. It follows from the definitions that for any sequence $A \in 2^\omega$, $\dim^{\mathrm{S}}_{\mathrm{H}} A \leqslant \dim^{\mathrm{S}}_{\mathrm{P}} A$. We call sequences for which Schnorr Hausdorff and Schnorr packing dimension coincide *Schnorr regular*, following [305] and [15]. It is easy to construct a non-Schnorr regular sequence, however, in Section 17.10.1 we will see that such sequences already occur among the class of c.e. sets.

17.9 Examples of Schnorr Dimension

The easiest way to construct examples of non-integral Schnorr dimension is obtained by 'inserting' zeroes into a sequence of dimension 1. Note that it easily follows from the definitions that every Schnorr random sequence has Schnorr Hausdorff dimension one. On the other hand, it is not hard to show that not every sequence of Schnorr Hausdorff dimension 1 is also Schnorr random.

The second class of examples is based on the fact that Schnorr random sequences satisfy the law of large numbers, not only with respect to Lebesgue measure (with corresponds to the uniform Bernoulli measure on 2^ω), but also with respect to other computable Bernoulli distributions. One can modify the definition of Schnorr tests to obtain randomness notions for arbitrary computable measures μ. Given a computable measure μ, a sequence is called Schnorr μ-random if it is not covered by any μ-Schnorr test.

Theorem 17.9.1 (Downey, Merkle, Reimann [91]). *(1) Let $S \in 2^\omega$ be Schnorr random, and let Z be a computable, infinite, co-infinite set of natural numbers such that $\delta_Z = \lim_n |\{0, \dots, n-1\} \cap Z|/n$ exists. Define a new sequence S_Z by*

$$S_Z \restriction_Z = S \quad \textit{and} \quad S_Z \restriction_{\overline{Z}} = 0,$$

where 0 here denotes the sequence consisting of zeroes only. Then it holds that

$$\dim_{\mathrm{H}}^{\mathrm{S}} S_Z = \delta_Z$$

(2) Let $\mu_{\vec{p}}$ be a computable Bernoulli measure on 2^ω with bias sequence $(p_0, p_1, \dots)$ such that, for all i, $p_i \in (0,1)$ and $\lim_n p_n = p$. Then it holds that for any Schnorr μ-random sequence B,

$$\dim_{\mathrm{H}}^{\mathrm{S}} B = -[p \log p + (1-p)\log(1-p)]$$

Part (1) of the theorem is straightforward (using for instance the martingale characterization of Schnorr Hausdorff dimension), part (2) is an easy adaption of the corresponding theorem for effective (i.e. Martin-Löf style) dimension (as for example in [**?**]).

It is not hard to see that for the examples given in Theorem 17.9.1, Schnorr Hausdorff dimension and Schnorr packing dimension coincide, so they describe Schnorr regular sequences. In Section 17.10.1 we will see that there are highly irregular c.e. sets of natural numbers: While all c.e. sets have Schnorr Hausdorff dimension 0, there are c.e. sets of Schnorr packing dimension 1.

17.10 A machine characterization of Schnorr dimension

One of the fundamantal aspects of the theory of 1-random reals is Schnorr's result that Martin-Löf's randomness coincidences with the the collection of reals that are incompressible in terms of (prefix-free) Kolmogorov complexity. Furthermore, there exists a fundamental correspondence between effective Hausdorff and packing dimension, $\dim_{\mathrm{H}}^1$ and $\dim_{\mathrm{P}}^1$, respectively, and Kolmogorov complexity: For any real A we have the following:

$$\dim_{\mathrm{H}}^1 A = \liminf_{n\to\infty} \frac{\mathrm{K}(A\restriction_n)}{n} \quad \text{and } \dim_{\mathrm{P}}^1 A = \limsup_{n\to\infty} \frac{\mathrm{K}(A\restriction_n)}{n}.$$

Note that in both equations one could replace prefix-free complexity K by plain Kolmogorov complexity C, since both complexities differ only by a logarithmic factor.

A comparably elegant characterization via machine complexity is not possible neither for Schnorr randomness nor Schnorr dimension. (This follows by the work in Chapter 15 and **??** on lowness where it is shown that no reasonable characterization of Schnorr randomness is possibly using standard prefix-free Kolmogorov complexity.) As we have seen in Chapter 10, to obtain a machine characterization of Schnorr dimension, we have to restrict the admissible machines to those with domains having computable measure in the same way as we did for Schnorr randomness. We recall from Chapter 10 that Downey and Griffiths charcaterized Schnorr randomness in terms of *computable machines*, that is, prefix-free machines whose domain had computable measure. Of course we can assume that the measure of the domain of a computable machine is 1. This can be justified, as in the case of Schnorr tests, by adding superfluous strings to the domain. We recall that Downey and Griffiths proved that a real A is Schnorr random if and only if for every computable machine M,

$$(\exists c)\,(\forall n)\ \mathrm{K}_M(A\restriction_n) \geqslant n - c.$$

Building on this characterization, we can go on to describe Schnorr dimension as asymptotic entropy with respect to computable machines.

Theorem 17.10.1 (Downey, Merkle and Reimann [91], also Hitchcock [**?**])**.** *For any real A the following holds:*

$$\dim_{\mathrm{H}}^{\mathrm{S}} A = \inf_M \underline{\mathrm{K}}_M(A) \quad \textit{where } \underline{\mathrm{K}}_M(A) := \liminf_{n\to\infty} \frac{\mathrm{K}_M(A\restriction_n)}{n},$$

where the infimum is taken over all computable prefix-free machines M.

Proof. ($\geqslant$) Let $s > \dim_{\mathrm{H}}^{\mathrm{S}} A$. We show that this implies $s \geqslant \underline{\mathrm{K}}_M(A)$ for some computable machine M, which yields $\dim_{\mathrm{H}}^{\mathrm{S}} A \geqslant \inf_M \underline{\mathrm{K}}_M(A)$.

As $s > \dim_{\mathrm{H}}^{\mathrm{S}} A$, there exists a Schnorr s-test $\{U_i\}$ such that $A \in \bigcap_i [\![U_i]\!]$. Assume each set in the test is given as $U_n = \{\sigma_{n,1}, \sigma_{n,2}, \dots\}$. Note that the Kraft-Chaitin Theorem is applicable to the set of axioms

$$\langle \lceil s|\sigma_{n,i}|\rceil - 1, \sigma_{n,i}\rangle \quad (n \geqslant 2,\ i \geqslant 1).$$

Hence there exists a prefix-free machine M such that for $n \geqslant 2$ and all i, $\mathrm{K}_M(\sigma_{n,i}) = \lceil s|\sigma_{n,i}|\rceil - 1$. Furthermore, M is computable since $\sum 2^{-\lceil s|\sigma_{n,i_n}|\rceil - 1}$ is computable.

We know that for all n there is an i_n such that $\sigma_{n,i_n} \sqsubset A$, and it is easy to see that the length of these σ_{n,i_n} goes to infinity. Hence there must be infinitely many n such that

$$\mathrm{K}_M(A\upharpoonright_n) \leqslant \lceil s|\sigma_{n,i}|\rceil - 1 \leqslant sn,$$

which in turn implies that

$$\liminf_{n\to\infty} \frac{\mathrm{K}_M(A\upharpoonright_n)}{n} \leqslant s.$$

($\leqslant$) Suppose $s > \inf_M \underline{\mathrm{K}}_M(A)$. So there exists a computable prefix-free machine M such that $s > \underline{\mathrm{K}}_M(A)$. Define the set

$$S_M = \{w \in 2^{<\omega} : \mathrm{K}_M(w) < |w|s\}.$$

We claim that this is a total Solovay s-cover for A. It is obvious that the set covers A infinitely often, so it remains to show that

$$\sum_{w\in S_M} 2^{-|w|s}$$

is a finite, computable real number. The finiteness follows automatically from

$$\sum_{w\in S_M} 2^{-|w|s} < \sum_{w\in S_M} 2^{-K_M(w)} \leqslant 1,$$

by Kraft's inequality and the fact that M is a prefix-free machine. To show computability, given ε compute the measure induced by $\mathrm{dom}(M)$ up to precision ε, so all strings not enumerated by that stage (call it s) will add in total at most ε to the measure of $\mathrm{dom}(M)$, which means they will also add at most ε to $\sum_{S_M} 2^{-|w|s}$, hence

$$\sum_{w\in S_{M_s}} 2^{-|w|s} \leqslant \sum_{w\in S_M} 2^{-|w|s} \leqslant \sum_{w\in S_{M_s}} 2^{-|w|s} + \varepsilon,$$

since a v contributes to S_M only if $\mathrm{K}(v) < |v|s$. But obviously, this only happens if $v \in \mathrm{dom}(M)$. □

We can use an analogous argument to obtain a machine characterization of Schnorr packing dimension.

Theorem 17.10.2. *For any real A the following holds:*

$$\dim_{\mathrm{P}}^{\mathrm{S}} A = \inf_M \overline{\mathrm{K}}_M(A) := \inf_M \left\{\limsup_{n\to\infty} \frac{K_M(A\upharpoonright_n)}{n}\right\},$$

where the infimum is taken over all computable prefix-free machines M.

The proof of 17.10.2 is an analogue to the proof of Theorem 17.10.1, if one uses the correspondence between martingales and tests shown in Theorem 17.8.7.

17.10.1 Schnorr Dimension and Computable Enumerability

Usually, when studying algorithmic randomness, interest focuses on *left c.e. real numbers* (also known as *c.e. reals*) rather than on *c.e. sets* (of natural numbers). The reason is that c.e. sets exhibit a trivial behavior with respect to most randomness notions, while there are c.e. reals which are random, such as Chaitin's Ω.

As regards left c.e. reals, with respect to computability, so far all notions of effective dimension show mostly the same behavior as the corresponding notions of randomness. For instance, as we have seen, every left c.e. real of positive effective dimension is Turing-complete, a result that was known before to hold for left c.e. Martin-Löf random reals. For Schnorr dimension, a straightforward generalization of Theorem 13.13.5 shoes that if A is left c.e. and $\dim_{\mathrm{H}}^{\mathrm{S}} A > 0$, then $S' \equiv_{T} 0''$.

As regards *computably enumerable sets* (of natural numbers), they are usually, in the context of algorithmic randomness, of marginal interest, since they expose a rather non-random behavior. For instance, it is easy to see that no computably enumerable set can be Schnorr random.

Proposition 17.10.3 (Folklore)**.** *No computably enumerable set is Schnorr random.*

Proof. Every infinite c.e. set contains an infinite computable subset. So, given an infinite c.e. set $A \subseteq \mathbb{N}$, choose some computable infinite subset B. Assume $B = \{b_1, b_2, \dots\}$, with $b_i < b_{i+1}$.

Define a Schnorr test $\{V_n\}$ for A as follows: At level n, put all those strings v of length $b_n + 1$ into V_n for which

$$v(b_i) = 1 \quad \text{for all } i \leqslant n+1.$$

Then surely $A \in [\![V_n]\!]$ for all n, and $\lambda[\![V_n]\!] = 2^{-n}$. □

It does not seem clear how to improve the preceding result to Schnorr dimension zero. Indeed, defining coverings from the enumeration of a set directly might not work, because due to the dimension factor in Hausdorff measures, longer strings will be weighted higher. Depending on how the enumeration is distributed, this might not lead to a Schnorr s-covering at all.

However, we can exploit the somewhat predictable nature of a c.e. set to define a computable martingale which is, for any $s > 0$, s-successful on the characteristic sequence of the enumerable set, thereby ensuring that each c.e. set has computable dimension 0.

Theorem 17.10.4 (Downey, Merkle, Reimann [91])**.** *Every computably enumerable set $A \subseteq \mathbb{N}$ has Schnorr dimension zero.*

Proof. Given rational $s > 0$, we show that there exists a computable martingale d such that d is s-successful on A.

First, partition the natural numbers into effectively given, disjoint intervals I_n such that $|I_n| \ll |I_{n+1}|$, for instance, $|I_n| = 2^{|I_0|+\cdots+|I_{n-1}|}$. Set $i_n = |I_n|$ and $j_n = i_0 + i_1 + \dots i_n$.

Denote by δ the upper density of A on I_n, i.e.

$$\delta = \limsup_{n\to\infty} \frac{|A \cap I_n|}{i_n}.$$

W.l.o.g. we may assume that $\delta > 0$. For any $\varepsilon > 0$ with $\varepsilon < \delta$ there is a rational number r such that $\delta - \varepsilon < r < \delta$. Given such an r, there must be infinitely many n_k for which

$$|A \cap I_{n_k}| > r i_{n_k}.$$

Define a computable martingale d by describing an accordant betting strategy as follows. At stage 0, initialize with $d(\varepsilon) = 1$. At stage $k+1$, assume d is defined for all τ with $|\tau| \leqslant l_k$ for some $l_k \in \mathbb{N}$. Enumerate A until we know that for some interval I_{n_k} with $j_{n_k-1} > l_k$ (i.e. I_{n_k} has not been bet on before),

$$|A \cap I_{n_k}| > r i_{n_k}.$$

For all strings σ with $l_k < |\sigma| \leqslant j_{n_k-1}$, bet nothing (i.e. d remains constant here). Fix a (rational) stake $\gamma > 2^{1-s} - 1$. On I_{n_k}, bet γ on the mth bit being 1 ($j_{n_k-1} < m \leqslant j_{n_k}$) if m has already been enumerated into A. Otherwise bet γ on the mth bit being 0. Set $l_{k+1} = j_{n_k}$.

When betting against A, obviously this strategy will lose at most $\lceil 2\varepsilon \rceil |I_{n_k}|$ times on I_{n_k}. Thus, for all sufficiently large n_k,

$$\begin{aligned} d(A\restriction_{l_{k+1}}) &\geqslant d(A\restriction_{l_k})(1+\gamma)^{i_{n_k} - \lceil 2\varepsilon\rceil |I_{n_k}|}(1-\gamma)^{\lceil 2\varepsilon\rceil |I_{n_k}|} \\ &= d(A\restriction_{l_k})(1+\gamma)^{i_{n_k}} \left(\frac{1-\gamma}{1+\gamma}\right)^{\lceil 2\varepsilon\rceil |I_{n_k}|} > 2^{(1-s)i_{n_k}} \left(\frac{1-\gamma}{1+\gamma}\right)^{\lceil 2\varepsilon\rceil |I_{n_k}|}. \end{aligned}$$

Choosing ε small and n large enough we see that d is s-successful on A. $\square$

On the other hand, it is not hard to see that for every Schnorr 1-test there is a c.e. set which is not covered by it. This means that the class of all c.e. sets has Schnorr Hausdorff dimension 1. For effective Hausdorff dimension, we have seen that Lutz [194] showed that for any class $\mathcal{X} \subseteq 2^\omega$,

$$\dim_{\mathrm{H}}^1 \mathcal{X} = \sup\{\dim_{\mathrm{H}}^1 A : A \in \mathcal{X}\}.$$

This means that effective dimension has a strong *stability* property. Thus, as observed by Downey, Merkle and Reimann [91], *class of c.e. sets yields an example that stability fails for Schnorr dimension.*

In contrast to Theorem 17.10.4, perhaps somewhat surprisingly, the upper Schnorr entropy of c.e. sets can be as high as possible, namely, there exist c.e. sets with Schnorr packing dimension 1. This stands in sharp contrast to the case of effective dimension, where Barzdins' Lemma, Theorem 19.1.1, shows that all c.e. sets have effective packing dimension 0. Namely, Barzdins showed that if A is a c.e. set, then there exists a c such that for all n, $\mathrm{C}(A\restriction_n) \leqslant 2\log n + c$.

The result will follow as a special case of a more general result: Every hyperimmune degree contains a set of Schnorr packing dimension 1. As the proof of the theorem shows, this holds mainly because of the requirement that all machines involved in the determination of Schnorr dimension are total.

Downey, Merkle and Reimann remark that a straightforward forcing construction show the existence of e degrees which do not contain any set sequence of high Schnorr packing dimension.

Theorem 17.10.5. *For any hyperimmune set B there exists a set $A \equiv_{\mathcal{T}} B$ such that*

$$\dim_{\mathrm{P}}^{\mathrm{S}} A = 1.$$

Furthermore, if the set B is c.e., then A can be chosen to be c.e., too.

Proof. For given B, it suffices to construct a set $C \leqslant_{\mathcal{T}} B$ such that $\dim_{\mathrm{P}}^{\mathrm{S}} C = 1$ and to let, for some set of places Z of sublinear density, the set A be a join of B and C where B is coded into the places in Z in the sense that

$$A\restriction_Z = B \quad \text{and} \quad A\restriction_{\overline{Z}} = C;$$

a similar argument works for the case of c.e. sets.

So fix any hyperimmune set B. Then there is a function g computable in B such that for any computable function f there are infinitely many n such that $f(n) < g(n)$. Partition the natural numbers into effectively given, disjoint intervals

$$\mathbb{N} = I_0 \cup I_1 \cup I_2 \cup \dots$$

such that $|I_0| + \ldots + |I_n| \ll |I_{n+1}|$ for all n, for instance, choose I_n such that $|I_{n+1}| = 2^{|I_0|+\cdots+|I_n|}$, and let $i_n = |I_n|$. Furthermore, let $M_0, M_1, \ldots$ be a standard enumeration of all prefix-free (not necessarily computable) Turing machines with uniformly computable approximations $M_e[s]$.

For any pair of indices e and n, let C have an empty intersection with the interval $I_{\langle e,n\rangle}$ in case

$$\sum_{M_e[g(n)](w)\downarrow} 2^{-|w|} \leqslant 1 - 2^{-i_{\langle e,n\rangle}}. \tag{17.12}$$

Otherwise, in case (17.12) is false, any string of length $i_{\langle e,n\rangle}$ not output by M_e at stage $g(n)$ on a code of length at most $i_{\langle e,n\rangle}$ is M_e-incompressible in the sense that the string has M_e-complexity of at least $i_{\langle e,n\rangle}$; pick such a string σ and let $A\restriction_{I_{\langle e,n\rangle}} = \sigma$ (in case such a string does not exist, the domain of the prefix-free machine M_e contains exactly the finitely many strings of length $i_{\langle e,n\rangle}$ and we don't have to worry about M_e). Observe that $A \leqslant_{\mathcal{T}} B$ because g is computable in B.

For any M_e with domain of measure one, the function f_e that maps n to the first stage such that (17.12) is false is total and in fact computable; hence there are infinitely many n such that $f_e(n) < g(n)$ and for all these n, the restriction of A to $I_{\langle e,n\rangle}$ is M_e-incompressible. To see that this ensures Schnorr packing dimension 1, suppose

$$\dim_{\mathrm{P}}^{\mathrm{S}} C < 1.$$

Then there exists a computable machine M, an $\varepsilon > 0$ and some $n_\varepsilon \in \mathbb{N}$ such that

$$(\forall n \geqslant n_\varepsilon)\,[\mathrm{K}_M(C\restriction_n) \leqslant (1-\varepsilon)n].$$

We define another total machine $\widetilde{M}$ with the same domain as M: Given x compute $M(x)$. If $M(x)\downarrow$, check whether $|M(x)| = i_0 + i_1 + \cdots + i_k$ for some k. If so, output the last i_k bits, otherwise output 0. Let e be an index of $\widetilde{M}$. By choice of the i_k, for all sufficiently large n, the $\widetilde{M}$-complexity of $C\restriction_{I_{\langle e,n\rangle}}$ can be bounded as follows

$$\mathrm{K}_{\widetilde{M}}(C\restriction_{I_{\langle e,n\rangle}}) \leqslant \mathrm{K}_M(C\restriction_{I_{\langle e,0\rangle}\cup\ldots\cup I_{\langle e,n\rangle}}) \leqslant (1-\varepsilon)(i_{\langle e,0\rangle}+\ldots+i_{\langle e,n\rangle}), \leqslant (1-\frac{\varepsilon}{2})i_{\langle e,n\rangle},$$

which contradicts the fact that by construction there are infinitely many n such that the restriction of C to the interval $I_{\langle e,n\rangle}$ is M_e-incompressible, that is, $\widetilde{M}$-incompressible.

In the case of a noncomputable c.e. set B, it is not hard to see that we obtain a function g as above if we let $g(n)$ be equal to the least stage such that some fixed effective approximation to B agrees with B at place n. Using this function g in the construction above, the set C becomes c.e. because for any index e and for all n, in case n is not in B the restriction of C to the interval $I_{\langle e,n\rangle}$ is empty, whereas otherwise it suffices to wait for the stage $g(n)$ such that n enters B and to compute from $g(n)$ the restriction of C to the interval $I_{\langle e,n\rangle}$, then enumerating all the elements of C in this interval. □

17.11 Kolmogorov complexity and the dimensions of individual strings

In this last section, we will look at Lutz' recent work assigning a dimension to individual *strings*, and a new characterization of prefix-complexity using such dimensions.

Mayordomo's characterization of the dimension of an individual real says considers the liminf of $\frac{K(\alpha\restriction n)}{n}$, and, equivalently, the infimum over all s of the values of $d^s(\alpha \restriction n)$ where d^s is the universal Σ^0_1 s-supergale. To discreteize this characterization, Lutz used three devices:

(i) He replaced supergales by *termgales*, which resemble supergales, yet have modifications to deal with the terminations of strings. This is done first via s-termgales and then later by termgales, which are uniform families of s-termgales.

(ii) He replaced $\to \infty$ by a finite threshold.

(iii) He replaced optimal s-supergale by and optimal termgale.

His idea is to introduce a new symbol $\Box$ to mark the end of a string. Thus a *terminated binary string* is $\sigma\Box$ with $\sigma \in 2^{<\omega}$.

Definition 17.11.1 (s-Termgale-Lutz [193])**.** For $s \in [0, \infty)$, an *s-termgale* is a function d from the collection of terminated strings T to $\mathbb{R}^+ \cup \{0\}$, such that $d(\lambda) \leqslant 1$, and

$$d(\sigma) \geqslant 2^{-s}[d(\sigma 0) + d(\sigma 1) + d(\sigma\Box)].$$

For $s = 1$ this is akin to the usual supermartingale condition:

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

However, as noted by Lutz, if each of $0, 1, \Box$ are equally likely, independent of previous bits, then the conditional expected capital on the bet is

$$\frac{d(\sigma 0) + d(\sigma 1) + d(\sigma\Box)}{3} = \frac{2}{3}d(\sigma).$$

However, the assumption that all bits are equally likely is the cause of this problem. The termination symbol $\Box$ should be regarded as having vanishingly small probability.

The 1-termgale payoff condition $d(\sigma) \geqslant \frac{d(\sigma 0)+d(\sigma 1)+d(\sigma\Box)}{2}$ is the same as the corresponding supermartingale consition, *except that the 1-termgale must divert some of its capital on the possibility of $\Box$, and this deversion is without compensation.* However, $\Box$ can occur at most once and we can make this impact small. Lutz [193] provided the following example:

Define $d(\lambda) = 1$ and $d(\sigma 0) = \frac{3}{2}d(\sigma)$, with $d(\sigma 1) = d(\sigma\Box) = \frac{1}{4}d(\sigma)$. Then d is a 1-termgale and if $\sigma\Box$ is a string with n_0 0's and n_1 1's,

$$d(\sigma\Box) = (\frac{3}{2})^{n_0}(\frac{1}{4})^{n_1+1} = 2^{n_0(1+\log 3)-2(n+1)}.$$

Thus if $n_0 >> \frac{2}{1+\log 3}(n+1) \approx 0.7737(n+1)$, then $d(\sigma\Box)$ is significantly greater than $d(\lambda)$ even though d loses three quarters of its capital when $\Box$ occurs.

The following is straightforward.

Lemma 17.11.2. *(i) Suppose that d, d' are functions taking terminated strings to $\mathbb{R}^+ \cup \{0\}$, and*

$$2^{-s|\sigma|}d(\sigma) = 2^{-s'|\sigma|}d'(\sigma).$$

Then d is an s-termgale iff d' is an s'-termgale.

(ii) *In particular, if d is a 0-termgale, then d' defined as*

$$d'(\sigma) = 2^{s|\sigma|}d(\sigma)$$

is an s-termgale, and each s-termgale can be ontained in this way from a 0-termgale.

We need the following technical lemma.

Lemma 17.11.3 (Lutz [193]). *Suppose that $s \in [0, \infty)$, and d is an s-termgale. Then*

$$\sum_{\sigma \in 2^{<\omega}} 2^{-s|\sigma|} d(\tau\sigma\Box) \leqslant 2^s d(\tau).$$

Proof. The first part of the proof assumes that d is a 0-termgale. Simple induction shows that

$$\sum_{\sigma \in 2^{<m}} d(\tau\sigma\Box) + \sum_{\sigma \in 2^m} d(\tau\sigma) \leqslant d(\tau).$$

Thus $\sum_{\sigma \in 2^{\leqslant m}} d(\tau\sigma\Box) \leqslant d(\tau)$, and hence

$$\sum_{\sigma \in 2^{<\omega}} d(\tau\sigma\Box) \leqslant d(\tau).$$

Now, if d is an s-termgale, then if we define $d'(\sigma) = 2^{-s}d(\sigma)$, we have that d' is a 0-termgale, and hence

$$\sigma_{\sigma \in 2^{<\omega}} 2^{-s|\sigma|} d(\tau\sigma\Box) = 2^{s|\tau\Box|} \sum_{\sigma \in 2^{<\omega}} d'(\tau\sigma\Box) \leqslant 2^s d(\tau).$$

$\Box$

We are now ready to define (optimal) termgales.

Definition 17.11.4 (Lutz [193]). (i) A *termgale* is a family $d = \{d^s : s \in [0, \infty)\}$ of s-termgales such that

$$2^{-s|\sigma|} d^s(\sigma) = 2^{-s'|\sigma|} d'(\sigma),$$

for all s, s' and $\sigma \in 2^{<\omega}$.

(ii) We say that a termgale is *constructive* or Σ_1^0, if d^0 is a Σ_1^0 function.

Definition 17.11.5 (Optimal Termgale-Lutz [193]). A σ_1^0 le $\tilde{d}$ is *optimal* iff for each constructive termgale d there is a constant $c > 0$ such that for all s and σ,

$$\tilde{d}^s(\sigma\Box) \geqslant cd^s(\sigma\Box).$$

In the same way that we connected discrete semimeasures with martingales, we have that the *termgale induced by* a Σ_1^0 discrete semimeasure m is

$$d[m]^s(\tau) = 2^{s|\sigma|} \sum_{\sigma \in 2^{<\omega} \wedge \tau \prec \sigma\Box} m(\tau).$$

Theorem 17.11.6 (Lutz [193])**.** *(i) If m is a universal effective semimeasure then $d[m]$ is an optimal constructivbe termgale.*

(ii) Hence there is a multiplicatively optimal constructive termgale.

Proof. Evidently, $d[m]$ is a termgale. Let $d = \{d^s : s \in [0,\infty)\}$ be a termgale. Define $\widehat{m}(\sigma) = d^0(\sigma\Box)$. By Lemma 17.11.3, $\widehat{m}$ is a Σ^0_1 discrete semimeasure. Since m is multiplicatively optimal, for some c we have $m(\sigma) \geqslant c\widehat{m}(\sigma)$, for all $\sigma \in 2^{<\omega}$. Hence

$$d[m]^s(\sigma\Box) = 2^{s|\sigma\Box|}m(\sigma) \geqslant 2^{s|\sigma\Box|}c\widehat{m}(\sigma) = cd^s(\sigma\Box).$$

$\Box$

Definition 17.11.7 (Lutz [193])**.** If d is a termgale, $\ell \in \mathbb{Z}^+$, and $\sigma \in 2^{<\omega}$, then *the dimension of σ relative to d at significance level ℓ* is

$$\dim^\ell_d(\sigma) = \inf\{s \in [0,\infty) : d^s(\sigma\Box) > \ell\}.$$

Now we characterize this dimension in terms of the optimal termgale.

Theorem 17.11.8 (Lutz [193])**.** *(i) Let $\tilde{d}$ denote the optimal termgale, and suppose that d is a Σ^0_1 termgale. Then there is a constant $c > 0$ such that, for all $\sigma \in 2^{<\omega}$,*

$$dim^\ell_{\tilde{d}}(\sigma) \leqslant \ dim^1_d(\sigma) + \frac{c}{1+|\sigma|}.$$

(ii) Hence, if $\tilde{d}_1$ and $\tilde{d}_2$ are both optimal Σ^0_1 termgales, $\ell_1, \ell_2 \in \mathbb{Z}^+$, there there is a constant $c > 0$ such that for all $\sigma \in 2^{<\omega}$,

$$dim^{\ell_1}_{\tilde{d}_1}(\sigma) - \ dim^{\ell_2}_{\tilde{d}_2}(\sigma) \leqslant \frac{c}{1+|\sigma|}.$$

Proof. Suppose the hypotheses of the Theorem are satisfied. By the optimality of $\tilde{d}$, there is a constant $b > 0$ such that for all s, $\tilde{d}^s(\sigma\Box) \geqslant bd^s(\sigma\Box)$. Now let $c = \log \ell - \log b$, and note that $c > 0$. Let $s > \dim^1_d(\sigma) + \frac{c}{1+|\sigma|}$. Let $s_1 = s - fracc1 + |\sigma|$. The $s_1 > \ \dim^1_d(\sigma)$. Hence,

$$\tilde{d}^s(\sigma\Box) \geqslant bd^s(\sigma\Box) = b2^{(s-s_1)|\sigma\Box|}d^{s_1}(\sigma\Box))$$

$$= b2^c d^{s_1}(\sigma\Box) > b2^c = \ell,$$

giving the result. $\Box$

Theorem 17.11.8 is the key to Lutz' definition of the dimension of a string. As observed by Lutz, it says that if we base our definition on a particular optimal termgale $\tilde{d}$ and significance level ℓ, then this choice will have little impact on $\dim^\ell_{\tilde{d}}(\sigma)$. Therefore we fix a particular optimal conctructive termgale, $d_\Box$ in Lutz' notation, and now define:

Definition 17.11.9 (Lutz [193])**.** For $\sigma \in 2^{<\omega}$,

$$\dim(\sigma) = \ \dim_{d_\Box}(\sigma).$$

We lose some of the finer points of Hausdorff dimension. It is no longer true that the dimesnion has 1 as a bound, but we do have an analog.

Theorem 17.11.10 (Lutz [193]). *There is a constant $b > 0$ such that for all $\sigma \in 2^{<\omega}$, $dim(\sigma) \leqslant b$.*

Proof. For each σ and s define $d^s(\sigma) = 2^{(s-2)|\sigma|}$ and $d^s(\sigma\Box) = 2^{(s-2)|\sigma|+1}$. Then $d = \{d^s : s \geqslant 0\}$ is a termgale and $d^2(\sigma\Box) = 2$ for all σ. Thus by Theorem 17.11.8, there is a constant c such that

$$\dim(\sigma) \leqslant \frac{c}{1+|\sigma|} \leqslant 2 + c.$$

Thus we can choose $b = 2 + \lceil b \rceil$. $\square$

Fianlly, we have an analog of The Lutz-Mayordomo characterization of effective Hausdorff dimension.

Theorem 17.11.11 (Lutz [193]). *There is a constant $c \in \mathbb{N}$ such that for all $\sigma \in 2^{<\omega}$,*

$$|K(\sigma) - |\sigma|\ dim(\sigma)| \leqslant c.$$

Proof. Let m be the optimal effective semimeasure. Then we have that for all σ, $d[m]^s(\sigma\Box) > 1$ is equivalent to saying $2^{s|\sigma\Box|}m(\sigma) > 1$, and hence equivalently,

$$s > \frac{1}{1+|\sigma|} \log \frac{1}{m(\sigma)}.$$

Consequently, $\dim_{d[m]}(\sigma) = \frac{1}{1+|\sigma|} \log \frac{1}{m(\sigma)}$, giving

$$\log \frac{1}{m(\sigma)} = (1+|\sigma|) \dim_{d[m]}(\sigma).$$

Lutz completes the proof by fixing constants c_0, c_1, c_2 such that

$$K(\sigma) - \log \frac{1}{m(\sigma)}| \leqslant c_0,$$

using the Coding Theorem, Theroem 6.9.2,

$$\dim_{d[m]}(\sigma) - \dim(\sigma)| \leqslant \frac{c_1}{1+|\sigma|},$$

by Theorem 17.11.8, and

$$\dim(\sigma) \leqslant c_2,$$

by Theorem 17.11.10. Then, if we put $c = c_0 + c_1 + c_2$, we have

$$|\log \frac{1}{m(\sigma)} - (1+|\sigma|) \dim(\sigma)| \leqslant c_1,$$

by the first two, and also

$$|(1+|\sigma|) \dim(\sigma) - |\sigma| \dim(\sigma)| \leqslant c_2,$$

giving, using the triangle inequality,

$$|K(\sigma) - |\sigma| \dim(\sigma)| \leqslant c.$$

$\square$

This result is yet another natural characterization of Prefix-free Kolmogorov complexity. Additionally, we can use known bound on Kolmogorov complexity to establish bound on the dimensions of finite strings.

Corollary 17.11.12 (Lutz [193])**.** *There exist constants* $c_1, c_2 \in \mathbb{N}$ *such that for all* $\sigma \in 2^{<\omega}$,

(i) $dim(\sigma) \leqslant 1 + \frac{\log |\sigma|}{|\sigma|} + \frac{c_1}{|\sigma|}$.

(ii) For $n \in \mathbb{Z}^+$*, and* $r \in \mathbb{N}$, $\mu(\cup\{[\sigma] : \ dim(\sigma) > 1 + \frac{\log|\sigma|}{|\sigma|} \ dim(|\sigma|) - \frac{r}{|\sigma|}\} > 1 - 2^{c_2 - r}$.

The proofs are to apply the K-bounds from Chapter 6.

We remark that Lutz proved a number of things in [193] such as showing that you can obtain Theorem 17.3.2 by using Theorem 17.11.11. We refer the reader there for more on the dimensions of finite strings.

Part IV

Further Topics

18
Ω as an operator

18.1 Introduction

We have already seen that Chaitin's Ω is a natural example of a 1-random real. We have also pushed the analogy that in effective randomness, prefix-free machines are the analogs of partial computable functions, and the measures of the domains of prefix-free machines, that is computably enumerable reals, take the role of computably enumerable sets. In more detail, we have the following.

1. The domains of partial computable functions are exactly the c.e. sets, while the measures of the domains of prefix-free machines are exactly the c.e. reals.
2. The canonical example of a non-computable set is the halting problem $\emptyset'$, i.e., the domain of a universal partial computable function. The canonical example of 1-random real is Ω, the halting probability of a universal prefix-free machine.
3. $\emptyset'$ is well-defined up to computable permutation, while Ω is well-defined up to Solovay equivalence.

So far in this book we have dodged the "relativization bullet" in our discussions in, say, the chapter on randomness in Cantor space, and in the chapter on the quantity of K-degrees. We did this by choosing a *fixed* standard universal machine when looking at relativized halting probabilities. In this chapter we will look at results of Downey, Hirschfeldt, Miller and

Nies [80]. These results grapple with Ω as a *class* of operators. All of the results and proofs in this section are taken from [80].

Relativizing the definition of $\emptyset'$ gives the jump operator. If $A \in 2^\omega$, then A' is the domain of a universal A-computable machine. Myhill's theorem relativizes, so A' is well-defined up to computable permutation. Furthermore, if $A \equiv_T B$, then A' and B' differ by a computable permutation. *A fortiori*, the jump is well-defined on the Turing degrees. The jump operator plays an important role in computability theory; it gives a natural, uniform and degree invariant way to produce, for each $A \in 2^\omega$, a set A' with Turing degree strictly above A.

What happens, on the other hand, when the definition of Ω is relativized? For any oracle $A \in 2^\omega$ there is an A-computable prefix-free machine which is universal with respect to all such machines. The use of such a machine will give us an operator $U^A \colon 2^{<\omega} \to 2^{<\omega}$. We will soon show that the Kučera-Slaman Theorem, Theorem 13.3.4, and the Calude, Khoussainov, Hertling, Wang Theorem, Theorem 13.3.3, both relativize (with some care, as we see below).

However, note that the jump operator has the property that $A \equiv_T B$ implies that $A' \equiv_T B'$. (Indeed $A' \equiv_m B'$.) If we wish to look at Ω as an analog of the jump, then we'd hope that Ω_U^A is well-defined, not just up to A-Solovay equivalence but even up to Turing degree. Similarly, we might hope for Ω_U to be a degree invariant operator: in other words, if $A \equiv_T B$ then $\Omega_U^A \equiv_T \Omega_U^B$. Were this the case, Ω_U would provide a counterexample to a longstanding conjecture of Martin; it would induce an operator on the Turing degrees which is neither increasing nor constant on any cone. But as we show in Theorem 18.5.7, there are oracles $A =^* B$ (i.e., A and B agree except on a finite set) such that Ω_U^A and Ω_U^B are not only Turing incomparible, but *relatively random*!. In particular, we can ensure that Ω_U^A is a c.e. real while making Ω_U^B as random as we like. It follows easily that the Turing degree of Ω_U^A generally depends on the choice of U, and in fact, that the degree of randomness of Ω_U^A can vary drastically with this choice.

If U is a universal prefix-free oracle machine, then we call $\Omega_U \colon 2^\omega \to [0,1]$ an *Omega operator*. In spite of their failure to be Turing degree invariant. it turns out the Omega-operators are extremely interesting in their own right, and provide the first natural example of an interesting c.e. operator that is not CEA. For example, in Section 18.4, we show that the range of an Omega operator has positive measure and that every 2-random real is in the range of *some* Omega operator. This is not true for every 1-random real. In Section 15.6, we prove that $A \in 2^\omega$ is mapped to a c.e. real by *some* Omega operator iff Ω is A-random. Such an A is called *low for* Ω and we study them further in Chapter 15.

In the final section, we consider the analytic behavior of Omega operators. We prove that Omega operators are lower semicontinuous but not continuous, and moreover, that they are continuous exactly at the 1-generic reals. We also produce an Omega operator which does not have a closed range. On the other hand, we prove that every non-2-random in the closure of the range of an Omega operator is actually in the range. As a consequence, there is an $A \in 2^\omega$ such that $\Omega_U^A = \sup(\mathrm{range}\,\Omega_U)$.

We remark that repeatedly use van Lambalgen's Theorem, Theorem 11.6.5 from Chapter 11 which the reader should recall states

1. $A \oplus B$ is 1-random iff $A \in 2^\omega$ is 1-random and $B \in 2^\omega$ is A-random.
2. If $A \in 2^\omega$ is 1-random and $B \in 2^\omega$ is A-random, then A is B-random.

We also require a few important theorems from classical measure theory from Chapter 4. They include *the Lebesgue density theorem*, Theorem 4.2.3, which states that if $\mathcal{S} \subseteq 2^\omega$ is measurable, then for almost every $A \in \mathcal{S}$,

$$\lim_{n\to\infty} 2^n \mu([A \upharpoonright n] \cap \mathcal{S}) = 1.$$

Actually, not need the full strength of Lebesgue's theorem. Instead, we use the following corollary which says that if a class has positive measure then there is a neighborhood in which the local measure is arbitrarily close to one.

Corollary 18.1.1. *Let $\mathcal{S} \subseteq 2^\omega$ have positive measure. For every $\varepsilon > 0$, there is a $\sigma \in 2^{<\omega}$ such that $2^{|\sigma|}\mu([\sigma] \cap \mathcal{S}) \geqslant 1 - \varepsilon$.*

The other theorem from Chapter 4 we will need is *Kolmogorov's* 0-1 Law: The reader should recall that this states that if $\mathcal{S} \subseteq 2^\omega$ is a measurable class closed under $=^*$, then $\mu(\mathcal{S})$ is either zero or one.

18.2 Omega operators

Definition 18.2.1. A partial computable oracle function $M^A\colon 2^{<\omega} \to 2^{<\omega}$ is a *prefix-free oracle machine* if M^A is prefix-free for every $A \in 2^\omega$. A prefix-free oracle machine U is *universal* if for every prefix-free oracle machine M there is a prefix $\rho_M \in 2^{<\omega}$ such that

$$(\forall A \in 2^\omega)(\forall \sigma \in 2^{<\omega})\; U^A(\rho_M \sigma) = M^A(\sigma).$$

The astute reader will note that the definition of universal prefix free machine is much stronger than the requirement that U^A is a universal A-computable prefix-free machine for all $A \in 2^\omega$. The standard machine is of course an example of a universal prefix-free machine. For a prefix-free oracle machine M, let Ω_M^A be the halting probability of M^A. Formally, $\Omega_M^A = \sum_{M^A(\sigma)\downarrow} 2^{-|\sigma|}$. This defines an operator $\Omega_M\colon 2^\omega \to [0,1]$.

Definition 18.2.2 (Downey, Hirschfeldt, Miller, Nies [80])**.** If U is universal, then we call Ω_U an *Omega operator.*

It is of course a straightforward relativization of Schnorr's proof that Ω_U^A is 1-A-random if U is an Omega operator because $K_U^A(\Omega_U^A \upharpoonright n) \geqslant n - \mathcal{O}(1)$

for all n. However, it is also easy to show that the the constant $\mathcal{O}(1)$ can be chosen independent of A. To wit, we have the following.

Proposition 18.2.3 (Downey, Hirschdeldt, Miller, Nies [80])**.** *There is a constant $b \in \omega$ such that, for each $A \in 2^\omega$, Ω_U^A is A-random with constant b, namely, $(\forall n)\ \ K^A(\Omega_U^A \upharpoonright n) \geqslant n - b$.*

Proof. We define a prefix-free oracle machine M as follows. For any $A \in 2^\omega$ and $\sigma \in 2^{<\omega}$, first calculate $\tau = U^A(\sigma)$. Then wait for a stage s such that $\Omega_U^A[s] \geqslant \tau - 2^{-|\tau|}$. If such an s is found, then let $M^A(\sigma) = s+1$. Because $M^A(\sigma) > s$, the convergence of $M^A(\sigma)$ can not already be taken into account in the calculation of $\Omega_U^A[s]$ (this is a standard assumption on the stage by stage approximation of U). Now assume that U simulates M by the prefix $\rho \in 2^{<\omega}$. So, either $\Omega_U^A < \tau - 2^{-|\tau|}$ or $\Omega_U^A \geqslant \Omega_U^A[s] + 2^{-|\rho\sigma|} \geqslant \tau - 2^{-|\tau|} + 2^{-|\rho\sigma|}$. Assume, for a contradiction, that there is an $n \in \omega$ such that $K^A(\Omega_U^A \upharpoonright n) < n - |\rho| - 1$. Letting σ be a minimal program for $\Omega_U^A \upharpoonright n$, we have proved that either $\Omega_U^A - (\Omega_U^A \upharpoonright n) < -2^{-n}$, which is absurd, or $\Omega_U^A - (\Omega_U^A \upharpoonright n) \geqslant -2^{-n} + 2^{-|\rho\sigma|} > -2^{-n} + 2^{-n+1} = 2^{-n}$, which is also impossible. This is a contradiction, so $(\forall n)\ \ K^A(\Omega_U^A \upharpoonright n) \geqslant n - |\rho| - 1$. $\square$

Remark 18.2.4. It is clear that $(\forall A \in 2^\omega)(\forall \sigma \in 2^{<\omega})\ K(\sigma) \geqslant K^A(\sigma) - c$, for some $c \in \omega$. This proves that all reals in the range of Ω_U are 1-random with constant $b + c$. In other words, the range of Ω_U is contained in the closed set $\{X \mid (\forall n)\ K(X \upharpoonright n) \geqslant n - b - c\}$. In particular, every real in $\overline{\text{range}(\Omega_U)}$, the closure of the range of Ω_U, is 1-random. We will discuss the range of Ω_U and its closure in more depth in Section 18.6.

Of course, Ω_U^A is an A-c.e. real, and every A-c.e. real is computable from A', hence $\Omega_U^A \leqslant_T A'$. Note that it is not usually the case that $\Omega_U^A \equiv_T A'$. To see this, let A be 1-random. By van Lambalgen's theorem, A is Ω_U^A-random. Hence $A \nleqslant_T \Omega_U^A$. Therefore, $\Omega_U^A \equiv_T A'$ only on a set of measure zero[1] On the other hand, the fact that $\Omega \equiv_T \emptyset'$ has a natural relativization in the following simple result, already seen in Corollary 11.10.5, but now in relativized form.

Proposition 18.2.5 (Kurtz [165])**.** *$\Omega_U^A \oplus A \equiv_T A'$, for every $A \in 2^\omega$.*

Proof. It is clear that $\Omega_U^A \oplus A \leqslant_T A'$. For the other direction, define a prefix-free oracle machine M such that $M^A(0^n 1)\downarrow$ iff $n \in A'$, for all $A \in 2^\omega$ and $n \in \omega$. Assume that U simulates M by the prefix $\tau \in 2^{<\omega}$. To determine if $n \in A'$, search for a stage s such that $\Omega_U^A - \Omega_U^A[s] < 2^{-(|\tau|+n+1)}$. This can be done computably in $\Omega_U^A \oplus A$. Note that U^A cannot converge on a string of length $|\tau| + n + 1$ after stage s, so

$$n \in A' \text{ if and only if } M^A(0^n1)\downarrow \text{ if and only if } U^A(\tau 0^n 1)\downarrow \text{ if and only if } U^A(\tau 0^n 1)[s]\downarrow.$$

[1] This result is strengthened in Chapter 15, in Theorem 15.8.2 where it is proven that $\Omega_U^A \equiv_T A'$ iff A is K-trivial, thus only for countably many choices of $A \in 2^\omega$.

Therefore, $A' \leqslant_T \Omega_U^A \oplus A$. □

Recall that $B \in 2^\omega$ is called *generalized low* (GL_1) if $B' \leqslant_T B \oplus \emptyset'$.

Theorem 18.2.6 (Nies and Stephan, published in [80])**.** *If a Δ_2^0 set $A \in 2^\omega$ is B-random, then B is GL_1.*

Proof. Let $f(n) = (\mu s)[(\forall t \geqslant s) \quad A_t \restriction n = A_s \restriction n]$, so that $f \leqslant_T \emptyset'$. Let $\widehat{\mathcal{R}}_e$ be the interval $[A_s \restriction e+1]$ when $\{e\}^B(e)$ converges at s. Clearly, if $\mathcal{R}_i = \bigcup_{e \geqslant i} \widehat{\mathcal{R}}_e$, then $\{\mathcal{R}_i\}_{i \in \omega}$ is a Martin-Löf test relative to B. Since $A \notin \bigcap_r \mathcal{R}_i$, A is only in finitely many $\widehat{\mathcal{R}}_e$'s. So, for almost all e such that $\{e\}^B(e)$ converges, $f(e) \geqslant (\mu s)\ \{e\}^B_s(e)\downarrow$. Hence $B' \leqslant_T B \oplus \emptyset'$. □

Theorem 18.2.6 implies that the class of low 1-random reals is closed under the action of an Omega operator.

Corollary 18.2.7. *For each Δ_2^0 1-random real $A \in 2^\omega$, Ω_U^A is generalized low. If A is a low 1-random, then Ω_U^A is low.*

Proof. Let $B = \Omega_U^A$. Clearly B is A-random, so by van Lambalgen's theorem, A is B-random and Theorem 18.2.6 applies. If in addition A is low, then Ω_U^A is Δ_2^0, hence low. □

18.3 On A-random A-c.e. reals

We can relativize Solovay reducibility as follows. For $A, X, Y \in 2^\omega$, we write $Y \leqslant_S^A X$ to mean that there is a $c \in \omega$ and a partial A-computable $\varphi\colon 2^{<\omega} \to 2^{<\omega}$ such that if $q < X$, then $\varphi(q)\downarrow < Y$ and $Y - \varphi(q) < c(X-q)$. We say that $X \in 2^\omega$ is *A-Solovay complete* if $Y \leqslant_S^A X$, for every A-c.e. real $Y \in 2^\omega$.

Some basic facts about Solovay reducibility relativize easily. For example:

Proposition 18.3.1. *A-randomness is closed upward under $\leqslant_S^A$. In other words, if Y is A-random and $Y \leqslant_S^A X$, then X is also A-random.*

The relativization of the Kučera-Slaman Theorem is also straightforward.

Theorem 18.3.2. *If X is an A-random A-c.e. real, then X is A-Solovay complete.*

On the other hand, a satisfactory relativization of The Calude, Khoussainov, Hertling, Wang Theorem, Theorem 13.3.3, which states that each 1-random c.e. real is a halting ptobability, presents some difficulty. The direct relativization states that if $X \in 2^\omega$ is an A-c.e. real and A-Solovay complete, then there is an oracle machine M such that M^A is universal for A-computable prefix-free machines and $X = \Omega_M^A$. The reader should note that it is quite possible that a function can be total for some oracles and partial for others. Thus is by no means clear that we should be able to relativize Theorem 13.3.3 to give a machine that is universal in our strong sense. Nevertheless the theorem is true.

Theorem 18.3.3 (Downey, Hirschfeldt, Miller, Nies [80])**.** *Suppose that X is an A-c.e. real and A-Solovay complete. Then there is a universal prefix-free oracle machine U such that $X = \Omega_U^A$.*

Proof. Let V be a universal prefix-free oracle machine. Because Ω_V^A is an A-c.e. real, we have $\Omega_V^A \leqslant_S^A X$. Choose $n \in \omega$ and a partial oracle-computable function $\varphi^B : 2^{<\omega} \to 2^{<\omega}$ such that 2^n and φ^A witness this Solovay reduction. In other words, if $q < \Omega_V^A$ is a binary rational, then $\varphi^A(q) \downarrow < \Omega_V^A$ and

$$\Omega_V^A - \varphi^A(q) < 2^n(X - q). \tag{18.1}$$

We also require n to be large enough that $2^{-n} \leqslant X \leqslant 1 - 2^{-n}$ (clearly, no computable real can be A-Solovay complete, so $X \neq 0, 1$).

We now define another universal prefix-free oracle machine U. To make U universal, let $U^B(0^n\sigma) = V^B(\sigma)$, for all $\sigma \in 2^{<\omega}$ and oracles $B \in 2^\omega$. For convenience, we preserve the stage of convergence; i.e., $U^B(0^n\sigma)[t]\downarrow$ iff $V^B(\sigma)[t]\downarrow$. The other strings in the domain of U are used to ensure that $\Omega_U^A = X$. Let $\psi^B : \omega \to 2^{<\omega}$ be a partial oracle-computable function such that $\{\psi^A(s)\}_{s\in\omega}$ is a nondecreasing sequence with limit X. Fix an oracle B. We add strings not extending 0^n to the domain of U in stages. For each s,

1. Compute $q_s = \psi^B(s)$.
2. Compute $r_s = \varphi^B(q_s)$.
3. Search for a t_s such that $\Omega_V^B[t_s] \geqslant r_s$.
4. If possible, add enough strings (not extending 0^n) to the domain of U at stage t_s to make $\Omega_U^B[t_s] = q_s$.

Note that (if $B \neq A$) this procedure may get stuck in any of the first three steps. In this case, U^B will converge on only finitely many strings not extending 0^n. This completes the construction of U, which is clearly a universal prefix-free oracle machine.

It remains to verify that $\Omega_U^A = X$. By the definition of ψ, we have $q_s = \psi^A(s)\downarrow < X$, for each s. Therefore, $r_s = \varphi^A(q_S)\downarrow < \Omega_V^A$. So, there is a stage t_s such that $\Omega_V^A[t_s] \geqslant r_s$. Because $q_s < X \leqslant 1-2^{-n}$, there are enough strings available in Step (iv) to ensure that $\Omega_U^A[t_s] \geqslant q_s$. But $\lim_s q_s = X$, so $\Omega_U^A \geqslant X$. Now assume, for a contradiction, that $\Omega_U^A > X$. Because the strings extending 0^n add at most $2^{-n} \leqslant X$ to Ω_U^A, there must be some s which causes too many strings to be added to the domain of U in Step (iv). In other words, there is an s such that $\Omega_U^A[t_s] = q_s$ and

$$\Omega_U^A[t_s] + 2^{-n}(\Omega_V^A - \Omega_V^A[t_s]) > X.$$

So, $\Omega_V^A - \Omega_V^A[t_s] > 2^n(X - q_s)$. But in Step (iii), we ensured that $\Omega_V^A[t_s] \geqslant r_s = \varphi^A(q_s)$. Therefore, $\Omega_V^A - \varphi^A(q_s) > 2^n(X - q_s)$, contradicting (18.1). This proves that $\Omega_U^A = X$, which completes the theorem. $\square$

Combining Propositions 18.2.3 and 18.3.1 with Theorems 18.3.2 and 18.3.3, we get the following corollary.

Corollary 18.3.4 (Downey, Hirschfeldt, Miller, Nies [80]). *For $A, X \in 2^\omega$, the following are equivalent:*

1. *X is an A-c.e. real and A-random.*
2. *X is an A-c.e. real and A-Solovay complete.*
3. *$X = \Omega_U^A$ for some universal prefix-free oracle machine U.*

18.4 Reals in the range of some Omega operator

We proved in the last section that $X \in 2^\omega$ is in the range of *some* Omega operator iff there is an $A \in 2^\omega$ such that X is both A-random and an A-c.e. real. What restriction does this place on X?

The impression we have is that somehow Ω is a *very special* 1-random real, and results such as Stephan's result that most 1-random reals are computationally feeble (Theorem 11.5.2) would suggest that most 1-random reals don't resemble Ω at all. *But* in relativized form we see that this is not the case. In this section, we show that every 2-random real is an A-random A-c.e. real for some $A \in 2^\omega$, but that not every 1-random real has this property. Furthermore, we prove that the range of every Omega operator has positive measure.

Theorem 18.4.1 (Downey, Hirschfeldt, Miller, Nies [80]). *If $X \in 2^\omega$ is 2-random, then X is an A-random A-c.e. real for some $A \in 2^\omega$.*

Proof. Let $A = (1 - X + \Omega)/2$. Then $X = 1 - 2A + \Omega$ is an A-c.e. real. In particular, take a nondecreasing computable sequence $\{\Omega_s\}_{s\in\omega}$ of rationals limiting to Ω. Then X is the limit of $\{1 - 2(A \upharpoonright s) + \Omega_s\}_{s\in\omega}$, a nondecreasing A-computable sequence of rationals. It remains to prove that X is A-random. Because X is 2-random it is Ω-random. Hence, by van Lambalgen's theorem, Ω is X-random. But then $A = (1 - X + \Omega)/2$ is X-random (because clearly, $\Omega \equiv_S^X (1 - X + \Omega)/2$). Therefore, applying van Lambalgen's theorem again, X is A-random. $\square$

As was mentioned above, the previous theorem cannot be proved if X is only assumed to be 1-random.

Example 18.4.2. $X = 1 - \Omega$ is not in the range of any Omega operator.

Proof. The 1-random real $X = 1 - \Omega$ is a co-c.e. real, i.e., the limit of a decreasing computable sequence of rationals. Assume that X is an A-c.e. real for some $A \in 2^\omega$. Then A computes sequences limiting to X from both sides; hence $X \leqslant_T A$. Therefore, X is not an A-random A-c.e. real for any $A \in 2^\omega$. $\square$

It would not be difficult to prove that $1-\Omega$ cannot even be in the *closure* of the range of an Omega operator. In fact, a direct proof is unnecessary because this follows from Theorem 18.6.4 below.

Now we consider a *specific* Omega operator. Let U be an arbitrary universal prefix-free oracle machine. Recall that analytic sets are measurable and that the image of an analytic set under any Borel operator—for example, Ω_U—is also analytic.

Theorem 18.4.3. *The range of Ω_U has positive measure. In fact, if $\mathcal{S} \subseteq 2^\omega$ is any analytic set whose downward closure under $\leqslant_T$ is 2^ω, then $\mu(\Omega_U[\mathcal{S}]) > 0$.*

Proof. Let $\mathcal{R} = \Omega_U[\mathcal{S}]$. Note that $\mathcal{R}$ is an analytic subset of 2^ω. Hence $\mu(\mathcal{R})$ is defined. Assume, for a contradiction, that $\mu(\mathcal{R}) = 0$. In particular, the outer measure of $\mathcal{R}$ is zero. This means that there is a nested sequence $\mathcal{U}_0 \supseteq \mathcal{U}_1 \supseteq \mathcal{U}_2 \supseteq \cdots$ of open subsets of 2^ω such that $\mathcal{R} \subseteq \mathcal{U}_n$ and $\mu(\mathcal{U}_n) \leqslant 2^{-n}$, for each $n \in \omega$. Take a set $B \in \mathcal{S}$ which codes $\{\mathcal{U}_n\}_{n\in\omega}$ in some effective way. Then $\{\mathcal{U}_n\}_{n\in\omega}$ is a B-Martin-Löf test, which implies that $\Omega_U^B \notin \bigcap_n \mathcal{U}_n$. But $\mathcal{R} \subseteq \bigcap_n \mathcal{U}_n$, so $\Omega_U^B \notin \mathcal{R} = \Omega_U[\mathcal{S}]$. This is a contradiction, so $\mu(\mathcal{R}) > 0$. $\square$

The theorem implies that many null classes have Ω_U-images with positive measure, for example $\mathcal{S} = \{A \mid (\forall n)\ 2n \notin A\}$.

We finish with a simple consequence of Theorem 18.4.3.

Corollary 18.4.4. *For almost every $X \in 2^\omega$, there is an $A \in 2^\omega$ such that $X =^* \Omega_U^A$.*

Proof. Let $\mathcal{S} = \{X \mid (\exists A \in 2^\omega)\ X =^* \Omega_U^A\}$. Then $\mathcal{S}$ is Σ^1_1—hence measurable by Lusin's theorem—and closed under $=^*$. But $\mu(\mathcal{S}) \geqslant \mu(\operatorname{range} \Omega_U) > 0$. It follows from Kolmogorov's 0–1 law that $\mu(\mathcal{S}) = 1$. $\square$

18.5 When Ω^A is a c.e. real

In this key section, we consider reals $A \in 2^\omega$ for which Ω_U^A is a c.e. real. Far from being a rare property, we will show that $\mu\{A \mid \Omega_U^A \text{ is a c.e. real}\} > 0$ for any fixed universal prefix-free oracle machine U. On the other hand, only a c.e. real can have an Ω_U-preimage with positive measure. So c.e. reals clearly play an important role in understanding Ω_U. Their main application here is in our proof that no Omega operator is degree invariant. Recall that we want to obtain reals $A =^* B$ such that Ω_U^A is a c.e. real while Ω_U^B is random relative to a given (arbitrarily complex) Z. We show that each of these outcomes occurs with positive measure in Propositions 18.5.4 and 18.5.5, respectively. Proposition 18.5.5 has no obvious connection to the c.e. reals, but in fact, Proposition 18.5.4—applied to a modification of the universal machine U—is used to prove it.

Theorem 18.5.1 (Downey, Hirschfeldt, Miller, Nies [80]). *Let M be a prefix-free oracle machine. If $\mathcal{P} \subseteq 2^\omega$ is a nonempty Π^0_1 class, then there is a $\emptyset'$-c.e. real $A \in \mathcal{P}$ such that $\Omega^A_M = \inf\{\Omega^A_M \mid A \in \mathcal{P}\}$, which is a c.e. real.*

Proof. Let $\mathcal{P} \subseteq 2^\omega$ be a nonempty Π^0_1 class and let $X = \inf\{\Omega^A_M \mid A \in \mathcal{P}\}$. Note that X is a c.e. real because it is the limit of the nondecreasing computable sequence $X_s = \inf\{\Omega^A_M[s] \mid A \in \mathcal{P}_s\}$. We will prove that there is a $A \in \mathcal{P}$ such that $\Omega^A_M = X$. Choose a sequence $\{B_n\}_{n\in\omega}$ such that $B_n \in \mathcal{P}$ and $\Omega^{B_n}_M - X \leqslant 2^{-n}$, for each $n \in \omega$. By compactness, $\{B_n\}_{n\in\omega}$ has a convergent subsequence $\{A_n\}_{n\in\omega}$. Note that $\Omega^{A_n}_M - X \leqslant 2^{-n}$. Let $A = \lim A_n$. Because $\mathcal{P}$ is closed, $A \in \mathcal{P}$. Therefore, $\Omega^A_M \geqslant X$. Assume, for a contradiction, that Ω^A_M is strictly greater than X. Take $m \in \omega$ such that $\Omega^A_M - X > 2^{-m}$. For some $s \in \omega$, $\Omega^A_M[s] - X > 2^{-m}$. Let k be the use of $\Omega^A_M[s]$ (under the usual assumptions on the use of computations, we can take $k = s$). In particular, if $B \upharpoonright k = A \upharpoonright k$, then $\Omega^A_M[s] = \Omega^B_M[s]$. Now take $n > m$ large enough that $A_n \upharpoonright k = A \upharpoonright k$. Then

$$2^{-n} \geqslant \Omega^{A_n}_M - X \geqslant \Omega^{A_n}_M[s] - X = \Omega^A_M[s] - X > 2^{-m} \geqslant 2^{-n}.$$

This is a contradiction, proving that $\Omega^A_M = X$.

Finally, we must prove that A can be a $\emptyset'$-c.e. real. Let $\mathcal{S} = \{A \in \mathcal{P} \mid \Omega^A_M = X\}$. Note that $\mathcal{S} = \{A \in 2^\omega \mid (\forall s)\ A \in \mathcal{P}_s \text{ and } \Omega^A_M[s] \leqslant X\}$. The fact that $X \leqslant_T \emptyset'$ makes $\mathcal{S}$ a $\Pi^0_1[\emptyset']$ class. We proved above that $\mathcal{S}$ is nonempty, so $A = \min(\mathcal{S})$ is a $\emptyset'$-c.e. real satisfying the theorem. □

We now consider reals $X \in 2^\omega$ such that $\Omega^{-1}_U[X]$ has positive measure.

Lemma 18.5.2 (Downey, Hirschfeldt, Miller, Nies [80]). *Let M be a prefix-free oracle machine. If $X \in 2^\omega$ is such that $\mu\{A \mid \Omega^A_M = X\} > 0$, then X is a c.e. real.*

Proof. By the Lebesgue density theorem, there is an $\sigma \in 2^{<\omega}$ such that $\mu\{A \succ \sigma \mid \Omega^A_M = X\} > 2^{-|\sigma|-1}$. In other words, Ω_M maps more than half of the extensions of σ to X. So, X is the limit of the nondecreasing computable sequence $\{X_s\}_{s\in\omega}$, where for each $s \in \omega$, we let X_s be the largest rational such that $\mu\{A \succ \sigma \mid \Omega^A_M[s] \geqslant X_s\} > 2^{-|\sigma|-1}$. □

For $X \in 2^\omega$, let $m_U(X) = \mu\{A \mid \Omega^A_U = X\}$. Define the *spectrum* of Ω_U to be $\mathrm{Spec}(\Omega_U) = \{X \mid m_U(X) > 0\}$. By the lemma, the spectrum is a set of 1-random c.e. reals. We prove that it is nonempty.

Kurtz [165] defined $Z \in 2^\omega$ to be *weakly n-random* if it is not contained in a Π^0_n class which has measure zero. He proved that this randomness notion lies strictly between n-randomness and $(n-1)$-randomness. In particular, an n-random real cannot be contained in a null Π^0_n class. We use this fact below.

Lemma 18.5.3. *For each c.e. real $X \in 2^\omega$, $m_U(X) > 0$ iff there is a 1-random $A \in 2^\omega$ such that $\Omega^A_U = X$.*

Proof. If $m_U(X) > 0$, then there is clearly a 1-random $A \in 2^\omega$ such that $\Omega^A_U = X$, as the 1-random reals form a class of measure one. For the other direction, assume that $A \in 2^\omega$ is a 1-random real such that $\Omega^A_U = X$. By van Lambalgen's theorem, the fact that X is A-random implies that A is X-random. But $X \equiv_T \emptyset'$, because X is a 1-random c.e. real, so A is 2-random. Note that $\{B \mid \Omega^B_U = X\}$ is a Π^0_2 class containing this 2-random. Hence $m_U(X) > 0$. ☐

Proposition 18.5.4. $\mathrm{Spec}(\Omega_U) \neq \emptyset$.

Proof. Apply Theorem 18.5.1 to a nonempty Π^0_1 class containing only 1-random reals. This gives a 1-random $A \in 2^\omega$ such that $X = \Omega^A_U$ is a c.e. real. Hence by Lemma 18.5.3, $X \in \mathrm{Spec}(\Omega_U)$. ☐

We have proved that Ω_U maps a set of positive measure to the c.e. reals. One might speculate that almost every real is mapped to a c.e. real. We now prove that this is not the case. (However, in the next section we will see that almost every real can be mapped to a c.e. real by *some* Omega operator.)

Proposition 18.5.5. *There is an $\varepsilon > 0$ such that*

$$(\forall Z \in 2^\omega)\ \mu\{B \mid \Omega^B_U \textit{ is } Z\textit{-random}\} \geqslant \varepsilon.$$

Proof. Let M be a prefix-free oracle machine such that $\Omega^B_M = B$, for every $B \in \omega$. Define a universal prefix-free oracle machine V by $V^B(0\sigma) = U^B(\sigma)$ and $V^B(1\sigma) = M^B(\sigma)$, for all $\sigma \in 2^{<\omega}$. Then $\Omega^B_V = (\Omega^B_U + B)/2$. Apply Proposition 18.5.4 to V to get a c.e. real $X \in 2^\omega$ such that $\mathcal{S} = \{B \mid \Omega^B_V = X\}$ has positive measure. Let $\varepsilon = \mu\mathcal{S}$.

Now take $Z \in 2^\omega$. We can assume, without loss of generality, that $Z \geqslant_T \emptyset'$. Let $B \in \mathcal{S}$ be Z-random. Then $\Omega^B_U = 2\Omega^B_V - B = 2X - B$ must also be Z-random, because $X \leqslant_T Z$. Therefore,

$$\mu\{B \in \mathcal{S} \mid \Omega^B_U \text{ is } Z\text{-random}\} \geqslant \mu\{B \in \mathcal{S} \mid B \text{ is } Z\text{-random}\} = \mu\mathcal{S} = \varepsilon,$$

since the Z-random reals have measure 1.[2] ☐

These results tell us that the Σ^0_3 class of reals A such that Ω^A_U is c.e. has intermediate measure.

[2]This simple construction shows more. Because $\Omega^B_U = 2X - B$ for $B \in \mathcal{S}$, we know that $\mu\{\Omega^B_U \mid B \in \mathcal{S}\} = \mu\{2X - B \mid B \in \mathcal{S}\} = \mu\mathcal{S} > 0$. Therefore, the range of Ω_U has a subset with positive measure. While this follows from the most basic case of Theorem 18.4.3, the new proof does not resort to Lusin's theorem on the measurability of analytic sets.

Corollary 18.5.6 (Downey, Hirschfeldt, Miller, Nies [80]). $0 < \mu\{A \mid \Omega_U^A$ *is a c.e. real*$\} < 1$.

The most important consequence of the work in this section is the following resoundingly negative answer to the question of whether Ω_U is degree invariant.

Theorem 18.5.7 (Downey, Hirschfeldt, Miller, Nies [80]).

1. *For all $Z \in 2^\omega$, there are $A, B \in 2^\omega$ such that $A =^* B$, Ω_U^A is a c.e. real and Ω_U^B is Z-random.*
2. *There are $A, B \in 2^\omega$ such that $A =^* B$ and $\Omega_U^A \mid_T \Omega_U^B$ (and in fact, Ω_U^A and Ω_U^B are 1-random relative to each other).*

Proof. (i) Let $\mathcal{S} = \{A \mid \Omega_U^A$ is a c.e. real$\}$ and $\mathcal{R} = \{B \mid \Omega_U^B$ is Z-random$\}$. By Propositions 18.5.4 and 18.5.5, respectively, both classes have positive measure. Let $\widehat{\mathcal{R}} = \{A \mid (\exists B \in \mathcal{R})\ A =^* B\}$. By Kolmogorov's 0–1 law, $\mu\widehat{\mathcal{R}} = 1$. Hence, there is an $A \in \mathcal{S} \cap \widehat{\mathcal{R}}$, completing the proof.

(ii) By part (i), there are $A, B \in 2^\omega$ such that $A =^* B$, Ω_U^A is a c.e. real and Ω_U^B is 2-random. Hence Ω_U^B is Ω_U^A-random and, by van Lambalgen's theorem, Ω_U^A is Ω_U^B-random. This implies that $\Omega_U^A \mid_T \Omega_U^B$. $\square$

We close the section with two further observations on the spectrum.

Proposition 18.5.8 (Downey, Hirschfeldt, Miller, Nies [80]). $\sup(\operatorname{range}\Omega_U) = \sup\{\Omega_U^A \mid A$ *is 1-random*$\} = \sup \operatorname{Spec}(\Omega_U)$.

Proof. Let $X = \sup(\operatorname{range}\Omega_U)$. Given a rational $q < X$, choose σ such that $\Omega_U^\sigma \geqslant q$. By the same proof as Proposition 18.5.4, there is a 1-random $A \succ \sigma$ such that Ω_U^A is a c.e. real. $\square$

Proposition 18.5.9 (Downey, Hirschfeldt, Miller, Nies [80]). *If $p < q$ are rationals and $\mathcal{C} = \{A \in 2^\omega \mid \Omega_U^A \in [p, q]\}$ has positive measure, then* $\operatorname{Spec}(\Omega_U) \cap [p, q] \neq \emptyset$.

Proof. Note that $\mathcal{C}$ is the countable union of $[\sigma] \cap \mathcal{C}$ for every $\sigma \in 2^{<\omega}$ such that $\Omega^\sigma \geqslant p$. Because $\mu\mathcal{C} > 0$, for some such σ we have $\mu([\sigma] \cap \mathcal{C}) > 0$. But $[\sigma] \cap \mathcal{C} = \{A \succ \sigma \mid \Omega^A \leqslant q\}$ is a Π_1^0 class. Let $\mathcal{R} \subset 2^\omega$ be a Π_1^0 class containing only 1-randoms with $\mu\mathcal{R} > 1 - \mu([\sigma] \cap \mathcal{C})$. Then $\mathcal{R} \cap [\sigma] \cap \mathcal{C}$ is a nonempty Π_1^0 class containing only 1-randoms. Applying Theorem 18.5.1 to this class, there is a 1-random real $A \in C$ such that $X = \Omega_U^A$ is a c.e. real. Then $X \in \operatorname{Spec}(\Omega_U) \cap [p, q]$, by Lemma 18.5.3 and the definition of $\mathcal{C}$. $\square$

We finish with another corollary of Theorem 18.5.1.

Corollary 18.5.10 (Downey, Hirschfeldt, Miller, Nies [80]). *There is a properly Σ_2^0 set $A \in 2^\omega$ such that Ω_U^A is a c.e. real.*

18.6 Analytic behavior of Omega operators

In this section, we examine Omega operators from the perspective of analysis. Given a universal prefix-free oracle machine $U \colon 2^{<\omega} \to 2^{<\omega}$, we consider two questions:

1. To what extent is Ω_U continuous?
2. How complex is the range of Ω_U?

To answer the first question, we observe that Ω_U is lower semicontinuous but not continuous. Furthermore, we prove that it is continuous exactly at 1-generic reals. Together with the semicontinuity, this implies that Ω_U can only achieve its supremum at a 1-generic. But must Ω_U actually achieve its supremum? This relates to the second question. Theorem 18.6.4 states that any real in $\overline{\text{range}(\Omega_U)} \smallsetminus \text{range}(\Omega_U)$ must be 2-random. Because $X = \sup(\text{range}\,\Omega_U)$ is a c.e. real—hence not 2-random, there is an $A \in 2^\omega$ such that $\Omega_U^A = X$.

It is natural to ask if $\text{range}(\Omega_U)$ is closed. In other words, is Theorem 18.6.4 vacuous. Example 18.6.6 demonstrates that for *some* choice of U, the range of Ω_U is not closed, and indeed, that $\mu(\text{range}\,\Omega_U) < \mu(\overline{\text{range}\,\Omega_U})$. Whether this is the case for *all* universal prefix-free oracle machines is left open. Furthermore, we know of no nontrivial upper-bound on the complexity of $\text{range}(\Omega_U)$, but we do observe that $\overline{\text{range}(\Omega_U)}$ is a Π^0_3 class.

Recall that a function $f \colon \mathfrak{X} \to \mathbb{R}$ is *lower semicontinuous* if $\{x \in \mathfrak{X} \mid f(x) > a\}$ is an open set for every $a \in \mathbb{R}$. Here $\mathfrak{X}$ is an arbitrary topological space. We claim that for any prefix-free oracle machine M, the function Ω_M is lower semicontinuous. Note that for any $A \in 2^\omega$,

$$(\forall \delta > 0)(\exists m)\, \Omega_M^A - \Omega_M^{A \upharpoonright m} \leqslant \delta \tag{18.2}$$

and hence $(\forall X \succ A \upharpoonright m)\, \Omega_M^A - \Omega_M^X \leqslant \delta$.

Proposition 18.6.1 (Downey, Hirschfeldt, Miller, Nies [80])**.** *Ω_M is lower semicontinuous for every prefix-free oracle machine M.*

Proof. Take $a \in \mathbb{R}$ and assume that $\Omega_M^A > a$. Choose a real $\delta > 0$ such that $\Omega_M^A - \delta > a$. By the observation above, there is an $m \in \omega$ such that $X \succ A \upharpoonright m$ implies that $\Omega_M^A - \Omega_M^X \leqslant \delta$. Therefore, $\Omega_M^X \geqslant \Omega_M^A - \delta > a$. So $[A \upharpoonright m]$ is an open neighborhood of A contained in $\{X \mid \Omega_M^X > a\}$. But A was an arbitrary element of $\{X \mid \Omega_M^X > a\}$, proving that this set is open. □

Next we prove that Omega operators are not continuous and characterize their points of continuity. Recall that an open set $\mathcal{S} \subseteq 2^\omega$ is *dense* along $A \in 2^\omega$ if each initial segment of A has an extension in $\mathcal{S}$. We say that A is 1-*generic* if A is in every Σ^0_1 class $\mathcal{S}$ that is dense along A. We prove that Ω_U is continuous exactly on the 1-generics, for any universal prefix-free oracle machine U.

Theorem 18.6.2 (Downey, Hirschfeldt, Miller, Nies [80]). *The following are equivalent for a set $A \in 2^\omega$:*

1. *A is 1-generic.*
2. *If M is a prefix-free oracle machine, then Ω_M is continuous at A.*
3. *There is a universal prefix-free oracle machine U such that Ω_U is continuous at A.*

Proof. (i) $\Longrightarrow$ (ii). Let M be any prefix-free oracle machine. By (18.2), it suffices to show that

$$(\forall \varepsilon)(\exists n)(\forall X \succ A \upharpoonright n)\ \Omega_M^X \leqslant \Omega_M^A + \varepsilon.$$

Suppose this fails for a rational ε. Take a rational $r < \Omega_M^A$ such that $\Omega_M^A - r < \varepsilon$. The following Σ_1^0 class is dense along A:

$$\mathcal{S} = \{B \mid (\exists n)\ \Omega_M^B[n] \geqslant r + \varepsilon\}.$$

Thus $A \in S$. But this implies that $\Omega_M^A \geqslant r + \varepsilon > \Omega_M^A$, which is a contradiction.

(ii) $\Longrightarrow$ (iii) is trivial.

(iii) $\Longrightarrow$ (i). Fix a universal prefix-free oracle machine U. We assume that A is not 1-generic and show that there is an $\varepsilon > 0$ such that

$$(\forall n)(\exists B \succ A \upharpoonright n)\ \Omega_U^B \geqslant \Omega_U^A + \varepsilon. \tag{18.3}$$

Take a Σ_1^0 class $\mathcal{S}$ which dense along A but $A \notin \mathcal{S}$. Define a prefix-free oracle machine L^X as follows. When (some initial segment of) $X \in 2^\omega$ enters $\mathcal{S}$, then L^X converges on the empty string. Thus L^A is nowhere defined. Let $c \in \omega$ be the length of the coding prefix for L in U. We prove that $\varepsilon = 2^{-(c+1)}$ satisfies (18.3).

Choose m as in (18.2) for the given universal machine, where $\delta = 2^{-(c+1)}$. For each $n \geqslant m$, choose $B \succ A \upharpoonright n$ such that $B \in \mathcal{S}$. Since L^B converges on the empty string, $\Omega_U^B \geqslant \Omega_U^A - 2^{-(c+1)} + 2^c = \Omega_U^A + \varepsilon$. □

Let U be a universal prefix-free oracle machine.

Corollary 18.6.3 (Downey, Hirschfeldt, Miller, Nies [80]). *If $\Omega_U^A = \sup(\operatorname{range}\Omega_U)$, then A is 1-generic.*

Proof. By the previous theorem, it suffices to prove that Ω_U is continuous at A. But note that the lower semicontinuity of Ω_U implies that

$$\{X \mid |\Omega_U^A - \Omega_U^X| < \varepsilon\} = \{X \mid \Omega_U^X > \Omega_U^A - \varepsilon\}$$

is open, for every $\varepsilon > 0$. Thus, A is 1-generic. □

The corollary above does not guarantee that the supremum is achieved. Surprisingly, it is. In fact, we can prove quite a bit more. One way to view the proof of the following theorem is that we are trying to prevent any real which is not 2-random from being in the closure of the range of Ω_U.

If we fail for some $X \in 2^\omega$, then it will turn out that $X \in \operatorname{range}(\Omega_U)$. Note that this is a consequence of universality; it is easy to construct a prefix-free oracle machine $M\colon 2^{<\omega} \to 2^{<\omega}$ such that Ω_M does not achieve its supremum.

Theorem 18.6.4 (Downey, Hirschfeldt, Miller, Nies [80]). *If* $X \in \overline{\operatorname{range}(\Omega_U)} \setminus \operatorname{range}(\Omega_U)$*, then* X *is* 2*-random.*

Proof. Assume that $X \in \overline{\operatorname{range}(\Omega_U)}$ is not 2-random and let $\mathcal{R}_X = \Omega_U^{-1}[X] = \{A \mid \Omega_U^A = X\}$. For each rational $p \in [0,1]$, define $\mathcal{C}_p = \{A \mid \Omega_U^A \leqslant p\}$. Note that every $\mathcal{C}_p$ is closed (in fact, a Π^0_1 class). For every rational $q \in [0,1]$ such that $q < X$, we will define a closed set $\mathcal{B}_q \subseteq 2^\omega$ such that

$$\mathcal{R}_X = \bigcap_{q<X} \mathcal{B}_q \cap \bigcap_{p>X} \mathcal{C}_p, \tag{18.4}$$

where q and p range over the rationals. Furthermore, we will prove that every finite intersection of sets from $\{\mathcal{B}_q \mid q < X\}$ and $\{\mathcal{C}_p \mid p > X\}$ is nonempty. By compactness, this ensures that $\mathcal{R}_X$ is nonempty, and therefore, that $X \in \operatorname{range}(\Omega_U)$.

We would like to define $\mathcal{B}_q$ to be $\{A \mid \Omega_U^A \geqslant q\}$, which would obviously satisfy (18.4). The problem is that $\{A \mid \Omega_U^A \geqslant q\}$ is a Σ^0_1 class; $\mathcal{B}_q$ must be closed if we are to use compactness. The solution is to let $\mathcal{B}_q = \{A \mid \Omega_U^A[k] \geqslant q\}$ for some $k \in \omega$. Then $\mathcal{B}_q$ is closed (in fact, clopen) and, by choosing k appropriately, we will guarantee that Ω_U^A is bounded away from X for every $A \notin \mathcal{B}_q$.

For each rational $q \in [0,1]$, we build a prefix-free oracle machine M_q. For $A \in 2^\omega$ and $\sigma \in 2^{<\omega}$, define $M_q^A(\sigma)$ as follows.

1. Wait for a stage $s \in \omega$ such that $\Omega_U^A[s] \geqslant q$.
2. Compute $\tau = U^{\emptyset'_s}(\sigma)$.
3. Wait for a stage $t \geqslant s$ such that $\Omega_U^A[t] \geqslant \tau$.

The computation may get stuck in any one of the three steps, in which case $M_q^A(\sigma)\uparrow$. Otherwise, let $M_q^A(\sigma) = t+1$. The value to which $M_q^A(\sigma)$ converges is only relevant because it ensures that a U-simulation of M_q can not converge before stage $t+1$.

We are ready to define $\mathcal{B}_q \subseteq 2^\omega$ for a rational $q \in [0,1]$ such that $q < X$. Assume that U simulates M_q by the prefix $\rho \in 2^{<\omega}$. Choose $\sigma \in 2^{<\omega}$ such that $U^{\emptyset'}(\sigma) = \tau \prec X$ and $|\tau| > |\rho\sigma|$. Such a τ exists because X is not 2-random. Choose $k_q \in \omega$ large enough that $U^{\emptyset'_s}(\sigma) = \tau$, for all $s \geqslant k_q$. Let $\mathcal{B}_q = \{A \mid \Omega_U^A[k_q] \geqslant q\}$.

We claim that the definition of $\mathcal{B}_q$ ensures that Ω_U^A is bounded away from X for any $A \notin \mathcal{B}_q$. Let $l_q = \min\{q, \tau\}$ and $r_q = \tau + 2^{-|\rho\sigma|}$. Clearly $l_q < X$. To see that $r_q > X$, note that $X - \tau \leqslant 2^{-|\tau|} < 2^{-|\rho\sigma|}$. Now assume that $A \notin \mathcal{B}_q$ and that $\Omega_U^A \geqslant l_q$. Thus $\Omega_U^A \geqslant q$ but $\Omega_U^A[k_q] < q$. This implies that the s found in Step (i) of the definition of M_q is greater than k_q. Therefore, $U^{\emptyset'_s}(\sigma) = \tau$. But $\Omega_U^A \geqslant \tau$, so Step (iii) eventually produces a $t \geqslant s$ such that $\Omega_U^A[t] \geqslant \tau$. This means that $M_q^A(\sigma) \downarrow= t+1$, so $U^A(\rho\sigma)\downarrow$ sometime after stage t, which implies that $\Omega_U^A \geqslant \Omega_U^A[t] + 2^{-|\rho\sigma|} \geqslant \tau + 2^{-|\rho\sigma|} = r_q$. We have proved that

$$\Omega_U^A \in [l_q, r_q) \implies A \in \mathcal{B}_q. \tag{18.5}$$

Next we verify (18.4). Assume that $A \in \mathcal{R}_X$. We have just proved that $A \in \mathcal{B}_q$ for all rationals $q < X$. Also, it is clear that $A \in \mathcal{C}_p$ for all rationals $p > X$. Therefore, $\mathcal{R}_X \subseteq \bigcap_{q<X} \mathcal{B}_q \cap \bigcap_{p>X} \mathcal{C}_p$. For the other direction, assume that $A \in \bigcap_{q<X} \mathcal{B}_q \cap \bigcap_{p>X} \mathcal{C}_p$. Thus if $q < X$, then $\Omega_U^A \geqslant \Omega_U^A[k_q] \geqslant q$. Hence $\Omega_U^A \geqslant X$. On the other hand, if $p > X$, then $\Omega_U^A \leqslant p$. This implies that $\Omega_U^A \leqslant X$, and so $\Omega_U^A = X$. Therefore $A \in \mathcal{R}_X$, which proves (18.4).

It remains to prove that $\mathcal{R}_X$ is nonempty. Let Q be a finite set of rationals less than X and P a finite set of rationals greater than X. Define $l = \max\{l_q \mid q \in Q\}$ and $r = \min(P \cup \{r_q \mid q \in Q\})$. Note that $X \in (l, r)$. Because $X \in \overline{\operatorname{range}(\Omega_U)}$, there is an $A \in 2^\omega$ such that $\Omega_U^A \in (l, r)$. From (18.5) it follows that $A \in \mathcal{B}_q$ for all $q \in Q$. Clearly, $A \in \mathcal{C}_p$ for every $p \in P$. Hence $\bigcap_{q\in Q} \mathcal{B}_q \cap \bigcap_{p\in P} \mathcal{C}_p$ is nonempty. By compactness, $\mathcal{R}_X$ is nonempty. □

If $X \in \operatorname{range}(\Omega_U)$ is not 2-random, then an examination of the construction gives an upper-bound on the complexity of $\Omega_U^{-1}[X]$. The Π_1^0 classes $\mathcal{C}_p$ can be computed uniformly. The $\mathcal{B}_q$ are also Π_1^0 classes and can be found uniformly in $X \oplus \emptyset'$. Therefore, $\Omega_U^{-1}[X] = \bigcap_{q<X} \mathcal{B}_q \cap \bigcap_{p>X} \mathcal{C}_p$ is a nonempty $\Pi_1^0[X \oplus \emptyset']$ class.

The following corollary gives an interesting special case of Theorem 18.6.4. It is not hard to prove that there is an $A \in 2^\omega$ such that $\Omega_U^A = \inf(\operatorname{range}\Omega_U)$ (see Theorem 18.5.1). It is much less obvious that Ω_U achieves its supremum.

Corollary 18.6.5 (Downey, Hirschfeldt, Miller, Nies [80])**.** *There is an $A \in 2^\omega$ such that $\Omega_U^A = \sup(\operatorname{range}\Omega_U)$.*

Proof. Note that $\sup(\operatorname{range}\Omega_U)$ is a c.e. real, hence not 2-random. So, the corollary is immediate from Theorem 18.6.4. □

No 1-generic is 1-random, so $\mu\{A \mid \Omega_U^A = \sup(\operatorname{range}\Omega_U)\} = 0$. Therefore, $\sup(\operatorname{range}\Omega_U)$ is an example of a c.e. real in the range of Ω_U which is not in $\operatorname{Spec}(\Omega_U)$.

One might ask if Theorem 18.6.4 is vacuous. In other words, is the range of Ω_U actually closed. We can construct a specific universal prefix-free oracle machine such that it is not. The construction is somewhat similar to the proof of Theorem 18.4.3. In that case, we avoid a measure zero set by using an oracle which codes a relativized Martin-Löf test coving that set.

Now we will avoid a measure zero *closed* set by using a natural number to code a finite open cover with sufficiently small measure.

Example 18.6.6. There is a universal prefix-free oracle machine V such that

$$\mu(\operatorname{range}\Omega_V) < \mu(\overline{\operatorname{range}\Omega_V}).$$

Proof. Let U be a universal prefix-free oracle machine. Let M be a prefix-free oracle machine such that

$$\Omega_M^A = \begin{cases} 1, & \text{if } |A| > 1 \\ 0, & \text{otherwise.} \end{cases}$$

Define a universal prefix-free oracle machine V by $V^A(0\sigma) = U^A(\sigma)$ and $V^A(1\sigma) = M^A(\sigma)$, for all $\sigma \in 2^{<\omega}$. This definition ensures that $\Omega_V^A \leqslant 1/2$ iff $|A| \leqslant 1$. Therefore $\mu(\operatorname{range}(\Omega_V) \cap [0, 1/2]) = 0$. We will prove that $\mu(\overline{\operatorname{range}(\Omega_V)} \cap [0, 1/2]) > 0$.

Let $\{\mathcal{O}_i\}_{i\in\omega}$ be an effective enumeration of finite unions of open intervals with dyadic rational endpoints. We construct a prefix-free oracle machine N. By the Recursion Theorem for prefix-free oracle machines, we may assume in advance that we know the prefix ρ by which V simulates N. Given an oracle $A \in 2^\omega$, find the least $n \in \omega$ such that $A(n) = 1$. Intuitively, N^A will try to prevent Ω_V^A from being in $\mathcal{O}_n$. Whenever a stage $s \in \omega$ occurs such that $\Omega_V^A[s] \in \mathcal{O}_n$ and $(\forall \sigma \in 2^{<\omega})\ V^A(\rho\sigma)[s] = N^A(\sigma)[s]$, then N^A acts as follows. Let ε be the least number such that $\Omega_V^A[s]+\varepsilon \notin \mathcal{O}_n$ and note that ε is necessarily a dyadic rational. If possible, N^A converges on additional strings with total measure $2^{|\rho|}\varepsilon$. This would ensure that $\Omega_V^A \geqslant \Omega_V^A[s]+\varepsilon$. If $\mu\mathcal{O}_n \leqslant 2^{-|\rho|}$, then N^A cannot run out of room in its domain and we have $\Omega_V^A \notin \mathcal{O}_n$.

Assume, for the sake of contradiction, that $\mu(\overline{\operatorname{range}(\Omega_V)} \cap [0, 1/2]) = 0$. Then there is an open cover of $\overline{\operatorname{range}(\Omega_V)} \cap [0, 1/2]$ with measure less than $2^{-|\rho|}$. We may assume that all intervals in this cover have dyadic rational endpoints. Because $\overline{\operatorname{range}(\Omega_V)} \cap [0, 1/2]$ is compact, there is a finite subcover $\mathcal{O}_n$. But $\mu\mathcal{O}_n < 2^{-|\rho|}$ implies that $\Omega_V^{0^n10^\omega} \notin \mathcal{O}_n$. This is a contradiction, so $\mu(\overline{\operatorname{range}(\Omega_V)} \cap [0, 1/2]) > 0$. $\square$

Note that the proof above shows that if U is a universal prefix-free oracle machine and $\mathcal{A} = \{\Omega_U^{0^n10^\omega}\}_{n\in\omega}$, then $\overline{\mathcal{A}}$ has positive measure and $\overline{\mathcal{A}} \setminus \mathcal{A}$ contains only 2-randoms.

Proposition 18.6.7 (Downey, Hirschfeldt, Miller, Nies [80])**.** $\overline{\operatorname{range}(\Omega_U)}$ *is a* Π_3^0 *class.*

Proof. It is easy to verify that $a \in \overline{\operatorname{range}(\Omega_U)}$ iff

$$(\forall \varepsilon > 0)(\exists \sigma \in 2^{<\omega}) \left[\begin{array}{c} \Omega_U^\sigma[|\sigma|] > a - \varepsilon\ \wedge \\ (\forall n \geqslant |\sigma|)(\exists \tau \succ \sigma)\ |\tau| = n \wedge \Omega_U^\tau[n] < a + \varepsilon \end{array} \right],$$

where ε ranges over rational numbers. This is a Π^0_3 definition because the final existential quantifier is bounded. $\square$

19

Complexity of computably enumerable sets

19.1 Barzdins' Lemma and Kummer complex sets

In this section, we look at the initial segment complexity of c.e. *sets* as opposed to that of left-c.e. *reals*. In particular, we will examine some beautiful results of Kummer [163], who demonstrated a fascinating gap phenomenon for the initial segment complexity of c.e. sets. We begin with an old result of Barzdins [23].

Theorem 19.1.1 (Barzdins' Lemma [23]). *If A is a c.e. set then $C(A \upharpoonright n \mid n) \leqslant \log n + O(1)$ and $C(A \upharpoonright n) \leqslant 2 \log n + O(1)$.*

Proof. To describe $A \upharpoonright n$ given n, it suffices to supply the number k_n of elements $\leqslant n$ in A and an e such that $A = W_e$, since we can recover $A \upharpoonright n$ from this information by running the enumeration of W_e until k_n many elements $\leqslant n$ appear in W_e. Such a description can be given in $\log n + O(1)$ many bits.

For the second part of the lemma, we can encode k_n and n as two strings σ and τ, respectively, each of length $\log n$. We can recover σ and τ from $\sigma\tau$ because we know the length of each of these two strings is exactly half the length of $\sigma\tau$. Thus we can describe $A \upharpoonright n$ in $2 \log n + O(1)$ many bits. □

Barzdins also constructed an example of a c.e. set A with $C(A \upharpoonright n) \geqslant \log n$ for all n. Of course, if $C(A \upharpoonright n) \leqslant \log n + O(1)$ for all n then, by Theorem 6.4.2, A is computable. A longstanding open question was whether the $2 \log n$ is optimal in the second part of Theorem 19.1.1. The best we could hope for is a c.e. set A such that $C(A \upharpoonright n) \geqslant 2\log n - O(1)$ *infinitely often*, since the following is known.

Theorem 19.1.2 (Solovay (unpublished))**.** *There is no c.e. set A such that $C(A \upharpoonright n \mid n) \geqslant \log n - O(1)$ for all n. Similarly, there is no c.e. set A such that $C(A \upharpoonright n) \geqslant 2\log n - O(1)$ for all n.*

Proof. Let A be a c.e. set. Let f be such that $A \upharpoonright f(n)$ has exactly 2^n elements. Note that $\log f(n) \geqslant n$. To describe $A \upharpoonright f(n)$ given $f(n)$ it suffices to provide n and an e such that $A = W_e$ (as in the proof of Barzdins' Lemma), which can be done in $\log n + O(1)$ many bits. But then $C(A \upharpoonright f(n) \mid f(n)) \leqslant \log n + O(1)$, and if n is sufficiently large then $\log n$ is much smaller than $\log f(n) - O(1) \geqslant n - O(1)$. Similarly, $C(A \upharpoonright f(n)) \leqslant 2\log n + \log f(n) + O(1)$, and again if n is sufficiently large then $2\log n + \log f(n)$ is smaller than $2\log f(n) - O(1)$. □

Solovay explicitly asked whether there is a c.e. set A such that $C(A \upharpoonright n) \geqslant 2\log n - O(1)$ infinitely often. As we will see the answer is yes, and there is a precise characterization of the degrees that contain such sets.

Definition 19.1.3. A c.e. set A is *Kummer complex* if for each d there are infinitely many n such that $C(A \upharpoonright n) \geqslant 2\log n - d$.

Theorem 19.1.4 (Kummer [163])**.** *There is a Kummer complex c.e. set.*

Proof. Let $t_0 = 0$ and $t_{k+1} = 2^{t_k}$. Let $I_k = (t_k, t_{k+1}]$ and

$$f(k) = \sum_{i=t_k+1}^{t_{k+1}} (i - t_k + 1).$$

Note that $f(k)$ asymptotically approaches $\frac{t_{k+1}^2}{2}$, and hence $\log f(k) > 2\log t_{k+1} - 2$ for sufficiently large k. So it is enough to build a c.e. set A such that for each k there is an $n \in I_k$ with $C(A \upharpoonright n) \geqslant \log f(k)$.

Enumerate A as follows. At stage $s+1$, for each $k \leqslant s$, if $C_s(A_s \upharpoonright n) < \log f(k)$ for all $n \in I_k$ and $I_k \nsubseteq A_s$, then put the smallest element of $\overline{A_s} \cap I_k$ into A_{s+1}.

Now suppose that $C(A \upharpoonright n) < \log f(k)$ for all $n \in I_k$. Then there must be a stage s such that $A_s \upharpoonright n = A \upharpoonright n$ and $C_s(A_s \upharpoonright n) < \log f(k)$. We must have $I_k \nsubseteq A_s$, since otherwise the smallest element of $\overline{A_s} \cap I_k$ would enter A, contradicting the assumption that $A_s \upharpoonright n = A \upharpoonright n$. Thus, all of I_k is eventually put into A. So for each $n \in I_k$ there are stages $s_0 < s_1 < \cdots < s_{n-t_k}$ such that $A_{s_{i+1}} \upharpoonright n \neq A_{s_i} \upharpoonright n$ and $C_{s_i}(A_{s_i}) < \log f(k)$, and hence there are at least $n - t_k + 1$ many strings σ with $|\sigma| = n$ and $C(\sigma) < \log f(k)$. Thus, there are at least $f(k)$ many strings σ such that $C(\sigma) < \log f(k)$, which is a contradiction. □

Kummer also gave an exact characterization of the degrees containing Kummer complex c.e. sets, using the notion of array noncomputability discussed in Section 5.21.

Theorem 19.1.5 (Kummer's Gap Theorem [163])**.**

(i) A c.e. degree contains a Kummer complex set iff it is array noncomputable.

(ii) In addition, if A is c.e. and of array computable degree, then for every unbounded, nondecreasing, total computable function f,

$$C(A \upharpoonright n) \leqslant \log n + f(n) + O(1).$$

(iii) Hence the c.e. degrees exhibit the following gap phenomenon: for each c.e. degree $\mathbf{a}$, either

(a) there is a c.e. set $A \in \mathbf{a}$ such that $C(A \upharpoonright n) \geqslant 2 \log n - O(1)$ for infinitely many n, or

(b) there are no c.e. set $A \in \mathbf{a}$ and $\varepsilon > 0$ such that $C(A \upharpoonright n) \geqslant (1 + \varepsilon) \log n - O(1)$ for infinitely many n.

Proof. Part (iii) follows immediately from parts (i) and (ii), so we prove the latter.

Part (i): To make A Kummer complex, all we need is to have the construction from Theorem 19.1.4 work for infinitely many intervals. Let I_k and $f(k)$ be as in the proof of that theorem, and let $\mathcal{I}$ be the very strong array $\{I_k\}_{k \in \mathbb{N}}$. Consider a c.e. set A that is $\mathcal{I}$-a.n.c. and fix an enumeration $\{A_s\}_{s \in \mathbb{N}}$ of A.

Define a c.e. set W as follows. At stage $s+1$, for each $k \leqslant s$, if $C_s(\overline{A_s \upharpoonright n}) < \log f(k)$ for all $n \in I_k$ and $I_k \nsubseteq A_s$, then put the smallest element of $\overline{A_s} \cap I_k$ into W.

Since A is $\mathcal{I}$-a.n.c., there are infinitely many k such that $A \cap I_k = W \cap I_k$. A similar argument to that in the proof of Theorem 19.1.4 now shows that, for any such k, if $C(A \upharpoonright n) < \log f(k)$ for all $n \in I_k$ then $I_k \subset A$, and hence that for each $n \in I_k$, there are at least $n - t_k + 1$ many strings σ with $|\sigma| = n$ and $C(\sigma) < \log f(k)$, which leads to the same contradiction as before. Thus A is Kummer complex.

By Theorem 5.21.4, each array noncomputable c.e. degree contains an $\mathcal{I}$-a.n.c. set, so each such degree contains a Kummer complex set.

Part(ii): Let A be c.e. and of array computable degree, and let f be a nondecreasing, unbounded, total computable function. Let $m(n) = \max\{i : f(i) < n\}$. Note that m is also nondecreasing, unbounded, and computable, and is defined for almost all n. Let $g(n) = A \upharpoonright m(n)$. (If $m(n)$ is undefined then let $g(n) = 0$.) Since g is computable in A, Lemma 5.21.6 implies that there is a total computable approximation $\{g_s\}_{s \in \mathbb{N}}$ such that $\lim_s g_s(n) = g(n)$ and $|\{s : g_s(n) \neq g_{s+1}(n)\}| \leqslant n$ for all n. (Recall that this cardinality is known as the number of mind changes of g at n.)

Suppose that we are given n and, for $k_n = \min\{i : m(i) \geqslant n\}$, we are also given the exact number p_n of mind changes of g at k_n. Then we can compute $g(k_n) = A \upharpoonright m(k_n)$, and hence also compute $A \upharpoonright n$, since $m(k_n) \geqslant n$. In other words, we can describe $A \upharpoonright n$ given n and p_n, so

$$C(A \upharpoonright n) \leqslant \log n + 2 \log p_n + O(1).$$

By the definition of k_n, we have $m(k_n - 1) < n$, so by the definition of m, we have $k_n - 1 \leqslant f(n)$. Furthermore, $p_n \leqslant k_n$, so

$$C(A \upharpoonright n) \leqslant \log n + 2 \log f(n) + O(1) \leqslant \log n + f(n) + O(1),$$

as desired. □

It is natural to ask whether there is a classification of, say, all jump classes in terms of initial segment complexity.

19.2 On the entropy of computably enumerable sets

In this section we will consider the Chaitin's notion of entropy of a computably enumerable set in the spirit of the Coding Theorem. We begin by recalling the basic theorem, Theorem 11.7.1, about the enumeration probability. Recall that

$$P(A) = \mu(\{X : W_e^X = A\},$$

where e is a universal index. Then we recall that Theorem 11.7.1 stated the following.

Theorem 19.2.1 (de Leeuw, Moore, Shannon, and Shapiro [60]). *If $P(A) > 0$ then A is computably enumerable.*

Definition 19.2.2 (Chaitin). Define

(i) $H(A) = \lceil -\log P(A) \rceil$.

(ii) $I(A) = \min\{H(j) : A = W_j\}$.

Theorem 19.2.3 (Solovay [284]). $I(A) \leqslant 3H(A) + H(H(A)) + O(1)$.

Proof. Theorem 19.2.3 follows from the lemma below.

Lemma 19.2.4. *There is a computable function $h(n, m)$ such that if $H(A) \leqslant n$ then for some $z \leqslant d.2^{3n}$, $h(n, z)$ is a Gödel number of A as a computably enumerable set.*

□

We remark that recently Shen has shown that that tigher bounds can be extracted from Solovay's proof in the case that A is a computably enumerable *finite* set.

Theorem 19.2.5 (Shen [**?**])**.** *Suppose that A is a finite computably enumerable set, then $I(A) \leqslant 2H(A) + H(H(A)) + O(1)$.*

19.3 Dimension for computably enumerable sets

19.4 The collection of non-random strings

19.4.1 The plain complexity case

Probably the most natural computably enumerable set to associate with randomness is

$$\overline{R}_C = \{x : C(x) < |x|\}^1,$$

the collection of strings that a non-random relative to plain complexity. This set has been extensively studied by Muchnik (see [**?**]) and others. We have already seen in Chapter 6 that this set is a simple c.e. set and hence it is not m-complete nor even *btt*-complete. On the other hand it is evident Turing complete. (This is an exercise in Li-Vitanyi [185], Exercise 2.63.) In this section we will prove the remarkable fact that $\overline{R}_C$ is truth-table complete. This is by no means obvious. The problem, roughly speaking, is as follows. Suppose that we are attempting to define a reduction from the halting problem K to $\overline{R}_C$, showing, say $K \leqslant_{\mathrm{wtt}} \overline{R}_C$.

For a single x we will have some collection of numbers F_x such that the reduction would like these number not to, say, all be in $\overline{R}_C$, unless x enters K. Should x enter K it is likely that we will be able to lower the complexity of those numbers and force them into $\overline{R}_C$, but it is also within the opponent's power to lower the complexity of those elements. There is no problem, in some sense, if we are using a *wtt*-reduction since with, say, one length per x, we could argue that we could eventually get to a random string that won't be enumerated by the universal machine into $\overline{R}_C$. But, in this construction, in advance, we will need to specify the sets so that we can make this idea work. Note that, for instance, we cannot take *all* the strings of length x since we can't know which are random. Kummer's idea is that we can have *blocks* of strings which we can control. This would seem the crudest idea that would have a chance of working, but Kummer was able to find a method to allow the proof to go through. The idea is the basis for other arguments such as Muchnik's Theorem Theorem 19.4.11 showing that the the set $\{(x, y, n) : K(x|y) < n\}$ is creative. Not surprisingly, the Kummer's argument works by a clever counting method.

---

[1]Strictly speaking, we should be studying $\{x : C(x) < |x| + d\}$ for some fixed constant d where $\{x : C(x) < |x| + d\}$ is the collection of Kolmogorov random strings. For ease of notation 'we will suppress this constant.

Theorem 19.4.1 (Kummer [164]). *$\overline{R}_C$ is truth table complete.*

Proof. (Kummer [164]) The proof is quite interesting in that the argument is *nonuniform*, though th construction itself is uniform in d which can be known by the Recursion Theorem. (It is the i_0 below which is the source of nonuniformity.) We will construct a partial computable function g in stages ($g = \cup_s g_s$ with $\mathrm{dom}(g_s) \subseteq 1^s 02^{<\omega}$.Additionally we will define a c.e. sequence of sets $E_n \subseteq \{0,1\}^n$ such that $|E_n| \leqslant 2^{n-2s-2}$, for all $n \geqslant 2s+2$. This allows us to define a partial computable

$$g_s(1^s 0\{0,1\}^s) = E_n{}^2.$$

For the sake of the possible *tt*-reductions, we will additionally define for each $i \leqslant 2^{2s+2}$ a possibly infinite sequence of finite sets $\mathcal{S}_i = S_{i,0}, S_{i,1} \dots,$ with $\ell(i)$ the current length of the i-th sequence. At the end of the proof we will argue that there will be a greatest i such that the i-th sequence is infinite, and for the correct choice of d, and almost all x,

$$x \in K \text{ iff } S_{i,x} \subseteq \overline{R}_C.$$

This is then a *tt*-reduction (indeed a *conjunctive* tt-reduction) reducing the halting problem, K, to $\overline{R}_C$.

The definition of $\mathcal{S}_i$ has higher priority than that of $\mathcal{S}_j$ if $i > j$. If $S_{i,x}$ is defined then all of its elements will have the same length, denoted by $m_{i,x}$. At a later stage some higher priority $\mathcal{S}_j$ may assert control and may occupy this length, in which case we will declare that $m_{i,x} \uparrow$, at all stages henceforth. (The point here is that the opponent will have made more strings of this length nonrandom, and hence the next choice will be "more likely" to succeed. See (iii) below.) Finally, if $m_{i,x}$ is defined and x appears in K then we will let $E_{m_{i,x}} = S_{i,x}$, declaring $n = m_{i,x}$ as *used*. This entails n no longer available as a candidate for $j > i$. Define

$$M_{n,s} = \{\sigma \in \{0,1\}^n : C_s(\sigma) < n\}.$$

Construction, Stage $s+1$ See if there is an i with $0 \leqslant i < 2^{2d+2}$ and an $n \leqslant s$ such that

(i) n is unused and $n \geqslant 2d+2$,

(ii) $n \neq m_{j,x}$ for all j, x with $j \geqslant i$ (priority), and

(iii) $i2^{n-2d-2} \leqslant |M_{n,s}|$.

If so choose the largest i, and for this i the least n, and then perform the following actions.

(a) Declare $m_{j,x} \uparrow$ for all j, x with $m_{j,x} = n$.

[2] That is, we will be using the Recursion Theorem to *lower* the complexity of the members of E_n to make them nonrandom should we need to to code whether x is in $\emptyset'$.

(b) Let $S_{i,\ell(i)}$ be the set of least k elements in $\{0,1\}^n - M_{n,s}$, where $k = \min\{2^n - |M_{n,s}|, 2^{n-2d-2}\}$.

(c) Set $m_{i,\ell(i)} = n$.

(d) Set $\ell(i) = \ell(i) + 1$.

To complete this stage of the construction, for all j, x such that $x \in K_{s+1}$ and $m_{j,x} \downarrow$ and used, define $E_{m_{j,x}} = S_{j,x}$ and declare that $m_{j,x}$ as used.
End of Construction

We know that since universal machine is optimal, there is a constant d such that for all σ, $C(\sigma) \leqslant C^g(\sigma) + d$, where C^g denotes the plain complexity generated by the partial computable function g. We now argue below for this fixed d. (As we remarked above, this is not the source of the nonuniformity in the construction, since we could know this d by the Recursion Theorem. However, the next Lemma, is nonuniform.)

Lemma 19.4.2. *There is a largest i such that $\ell(i)$ is incremented infinitely often.*

Proof. We can always invoke the first part of the construction for $i = 0$ and $n = s$. Since there are only 2^{2d+2} many i, the result follows. □

Now using Lemma 19.4.2, we can fix the relevant $i = i_0$. We can also fix a s_0 such that at no stage $t \geqslant s_0$ do we choose $\mathcal{S}_j$, for any $j > i$. Notice that $S_{i_0,x}$ is defined for all x and moreover the strings in $S_{i_0,x}$ and $S_{i_0,y}$ are of different lengths for $x \neq y$.

Lemma 19.4.3. *$m_{i_0,x}$ is almost always defined.*

Proof. Since $\ell(i_0, s) \to \infty$, for all x there is a stage where $m_{i_0,x}$ is defined. After stage s_0 this definition is not initialized. □

Let x_0 be the least x such that for all $x \geqslant x_0$, $m_{i_0,x}$ is always defined.

Lemma 19.4.4. *For all $x \geqslant x_0$, if $x \in K$, then $E_{m_{i_0,x}} = S_{i_0,x}$. If $x \notin K$, then $E_{m_{i_0,x}} = \emptyset$.*

Proof. For $x \geqslant x_0$, there is a stage s where $m_{i_0,x} \downarrow$ at stage s, with $m_{i_0,x} = n$, say. Thus n is unused and $E_n = \emptyset$ at the beginning of stage s. Since $m_{i_0,x} \downarrow$ at all later stages, E_n remains as $\emptyset$ at all later stages unless x enters K in which case we set it to be $S_{i_0,x}$ in the last part of the construction. □

Lemma 19.4.5. *For almost all x, $x \in K$ iff $S_{i_0,x} \subseteq \overline{R}_C$.*

Proof. If $x \geqslant x_0$, and $x \in K$ then $E_{m_{i_0,x}} = S_{i_0,x}$. Choose x sufficiently large that $m_{i_0,x} \geqslant 2d - 2$. Let $n = m_{i_0,x}$. Then for each $z \in E_n$ there is a $z' \in \{0,1\}^{n-2d-2}$ such that $g(1^d 0 z') = z$. That is, the construction sets the g-plain complexity of z to less than n-d-1, and since d is the relevant coding constant, this sets $C(z) \leqslant C^g(z) + d < n$. Therefore $z \in \overline{R}_C$.

Now for the hard direction. Suppose that there are infinitely many x with $x \notin K$ and $S_{i_0,x} \subseteq \overline{R}_C$. Choose such an x for which $m_{i_0,x}$ is always defined. At the stage s_1+1, $m_{i_0,x}$ was defined we had

$$|M_{n,s_1}| \geqslant i_0 2^{n-2d-2} \wedge S_{i_0,x} \cap M_{n,s_1} = \emptyset.$$

By hypothesis, there is a stage $s_2 > \max\{s_0, s_1\}$ such that $S_{i_0,x} \subseteq M_{n,s_2}$. But then, by the definition of $S_{i_0,x}$, it will follow that

$$|M_{n,s_2}| \geqslant (i_0+1)2^{n-2d-2}.$$

(This follows since we get an additional $k = \min\{2^n - M_{n,s_1}, 2^{n-2d-2}\}$ elements into M_{n,s_2}, which either means that the remaining $2^n - M_{n,s_1}$ elements enter (impossible), or we get the required additional 2^{n-2d-2} elements.) Since $|R_C \cap \{0,1\}^n| \geqslant 1$, $i_0 + 1 < 2^{2d-2}$ and hence at stage s_1+1 we would choose $\mathcal{S}_j$ for some $j > i_0$, a contradiction. □

□

With a relatively straightforward modification of the previous proof (using suitable sets of strings L_n in place of $\{0,1\}^n$), Kummer proved the following which applies to sets like $\{\sigma : C(\sigma) \leqslant \log|\sigma|\}$.

Corollary 19.4.6 (Kummer [164])**.** *Suppose that f is computable with $f(x) \leqslant \lfloor \log(x+1) \rfloor$, and $\lim_{x\to\infty} f(x) = \infty$. Then the set $\{x : C(x) < f(x)\}$ is conjunctive tt-complete.*

One exciting arena of research here is the work looking at efficient reductions to $\overline{R}_C$, such as that of Allender, Buhrman, Koucký, van Melkebeek, and Ronneburger, (e.g. [1]), and others. They observe that the Kummer reduction, whilst having computable use, has exponential growth in the number of queries it uses. It is known that, for instance, a polynomial number of queries is not possible. They ask what sets *can* be, say, polynomial time reduced to $\overline{R}_C$, and time as space variants (not treated in this book). Amazingly this question seems to have impact on basic separation questions amongst complexity classes. For instance, they show that $PSPACE \subseteq P^{\overline{R}_C}$. The methods are highly nontrivial. This work is beyond the scope of the present book.

19.4.2 The prefix-free case

As with actually defining prefix-free complexity for strings, as we saw in Chapter 6.11 we have two possibilities for prefix-free complexity. They are

- $\overline{R}_{KS} = \{x : K(x) < |x| + K(|x|)(+O(1)\}$, (the set of non-strongly Chaitin random strings) and
- $\overline{R}_K = \{x : K(x) < |x|(+O(1)\}$ (the set of non-weakly Chaitin random strings).

It had been a longstanding open question, going back to the Solovay manuscript, whether $\overline{R}_{KS}$ was a computably enumerable set. This was finally solved in early 2005 by Joe Miller.

Theorem 19.4.7 (Miller [210])**.** *Fix $c \geqslant 0$ and let $B = \{v : K(v) < |v| + K(|v|) - c\}$. If A contains B and has property (*) below, then A is not a c.e. set.*

(*) For all n,

$$|A \cap 2^n| \leqslant 2^{n-1}.$$

Corollary 19.4.8. *For all c, $B = \{v : K(v) < |v| + K(|v|) - c\}$ is not Σ_1^0.*

Miller points out that this result gives a weak form of the result of Solovay that there are strings that are Kolmogorov random but not strongly Chaitin random. (Corollary 7.3.3) Whilst the result below is weaker, the proof is much easier!

Corollary 19.4.9. *Fix $k \geqslant 0$. There is no c such that all strings Kolmogorov random for constant k are strongly Chaitin random for constant c.*

Proof. Otherwise, the set of strings not strongly Chaitin random for constant c would be contained in the set not Kolmogorov random for constant k. But the latter is a Σ_1^0 set of strings with property (*). □

Proof. (of Theorem 19.4.7) Assume that A is Σ_1^0.

We define a prefix-free machine M that U simulates with a prefix of length k. By the Recursion Theorem, we can assume that M knows k. For any stage s and n such that $K_s(n) < K_{s-1}(n)$, M should find the first $2^{n-k-c-2}$ strings of length n that are not in A_s and give these strings descriptions of length $n + K_s(n) - k - c - 1$. That completes the definition of M. It is straightforward to check that M will not run out of room in its domain and that it will always have enough strings not in A_s to choose from.

The main point is that, if $K_{s-1}(n)$ is wrong, then M is going to make sure that U compresses at least $2^{n-k-c-2}$ strings of length n that are not in A_s by at least $c+1$ bits. These strings must eventually be added to A, so $|A \cap 2^n| \geqslant |A_s \cap 2^n| + 2^{n-k-c-2}$.

Let b be the greatest natural number such that, for infinitely many n,

$$|A \cap 2^n| \geqslant b2^{n-k-c-2}.$$

Define a partial computable function f as follows. If $|A_s \cap 2^n| \geqslant b2^{n-k-c-2}$, then $f(n) = K_{s-1}(n)$. Note that by the argument above, if $f(n)$ is defined, then $f(n) = K(n)$ (except perhaps finitely often). Otherwise, we would contradict the choice of b. Furthermore, the choice of b guarantees that f has an infinite domain. But such an f is impossible. Hence A is not Σ_1^0. □

Lets turn to the set of strings that are not weakly Chaitin random, $\overline{R}_K$. This set certainly is computably enumerable. Again it is clearly Turing complete. But is it *tt*-complete? An. A. Muchnik proved that the answer may be no, depending on the choice of universal machine. Muchnik considers the *overgraph* of K. That is, the set

$$M_K = \{(x, n) : K(x) < n\}.$$

Of course, Kummer's Theorem 19.4.1 shows that the overgraph of plain complexity

$$M_C\{(x, n) : C(x) < n\}$$

is always *tt*-complete. Muchnik proved that whether M_K is *tt*-complete depends upon the choice of universal machine for the definition of prefix-free Kolmogorov complexity. We will denote M_K^Q as the version of M_K relative to a universal machine Q, and similarly $\overline{R}_K^Q$.

Theorem 19.4.10 (Muchnik, An. A. [221]). *There exist universal prefix-free machines V and U such that*

(i) M_K^V is tt-complete.

(ii) M_K^U (and hence $\overline{R}_K^U$) is not tt-complete.

We remark that the proof, below, of (ii) has a number of interesting new ideas, especially casting complexity considerations in the context of games on finite directed graphs.

Proof. (i) Let W be any c.e. set and K be any prefix-free complexity. We modify K to get our new complexity K^V. On input z V first tries to parse z as $z = 0^x 1v$. If successful, and $x \notin W$, $K^V(z)$ will be $K(z) + 2$, if this is even, and $K(z) + 3$ if odd. If $z = 0^x 1v$ *and* $x \in W$, or $z = 0^y$, for some y, we let $K^V(z) = K(z) + 1$ if it is odd, and $K(z) + 2$ otherwise.

We show that $W \leqslant_{tt} \overline{R}_K^V$. Fix a large enough c and for each $p \leqslant cx$ ask if $(0^x 1, p) \in M_{K^V}$. Then $x \in W$ iff $K^V(0^x 1)$ is odd.

(ii) This argument is much more difficult. We will follow Muchnik's proof which uses strategies from finite games. This perhaps is an artifact within the proof, but nevertheless is interesting in its own right.

A finite game is played on (directed) graphs and is determined by

- a finite set of positions (vertices).
- two directed graphs on that seti, called the α-graph and the β-graph.
- Two complementary subsets of the positions, the α-set and the β-set, and
- The initial position d_0.

For this game, the union of the α- and the β-graphs will be *acyclic*.

As with pebble games, the game begins at position d_0, players α and β play in turns, and a move of the game is as follows. Suppose that player ν is at position d. Then they can either stay at position ν or move to position d' provided that (d, d') is in the ν-graph. Since the union of the two graphs is acyclic, the game must stabilize at some stage, and if that stable position is in the ν-graph, then player ν wins.

Clearly this game is determined, but there is an explicit winning strategy for one of the players.

To wit, Let D_ν be the set of all positions d such that (d, d_0) is in the ν-graph. Then, for every position d in D_ν, we can define a residual game E^d_ν, as being the game started at position d and having all the edges in D_ν removed, but with the order of the players switched. We can use this construction, and the finiteness of the game, to construct an *effective* winning strategy for the game. (All this is well-known, and we refer the reader to (e.g.) Khoussainov-Nerode [142] for more details on finite games.)

Returning to the proof, let K denote any version of prefix-free Kolmogorov complexity. We will construct a computably enumerable function F and define a function H, with $H(\sigma) = \min\{K(\sigma) + 2, F(\sigma)\}$ which will be a prefix-free Kolmogorov complexity. Additionally, we will construct a computably enumerable set U such that

$$U \not\leq_{tt} M_H.$$

For technical reasons, we assume that $\sum_x 2^{-K(x)} < \frac{1}{4}$, and will ensure that $\sum_x 2^{-F_0(x)} < \frac{1}{4}$. Let Γ_n denote the n-th partial truth table reduction. We will construct the diagonalizing set U as a set of triples (n, i, j). We will use those with first coordinate n to diagonalize Γ_n, and consider each n infinitely often using the order $1, 1, 2, 1, 2, 3, 1, 2, 3, 4, \ldots$.

In the construction, if we are already processing n, we assign two numbers i_n and j_n, and for some of the n's we will have assigned a finite game G_n. Each time n is considered, we can either let one of the players play the game $G_n(= G^s_n)$, or we can change both the game G_n and the numbers i_n and j_n. The positions of the game are determined by a finite set A_n (fixed for the game), a set of functions $h_n : A_n \mapsto \mathbb{N}$, and sets of pairs of rational numbers q_n and p_n. Here will will allow h_n, p_n, q_n all to change without any move.

Moves of the first player will determine the value of F_s, and the second player's moves will be determined by the decrease in the estimate of K_s.

When n is processed for the first time, we set $i_n = j_n = 0$. Each time we re-consider n, we will increase the value of i_n by 1 if

- there is a number $m < n$ such that the position of G_m changed, or G_m itself changed since the last time we considered n (priority).

If a change in the estimate of K_s violates the rules of G_n we will increase the value of j_n by 1.

Construction, stage s

At stage s, assume that we are processing n. Using the rules above, we determine the values of i_n and j_n. If there is no change, and G_n is already defined, then play the next move of the game G_n.

If G_n is not defined or either of the values i_n or j_n changed, run s steps in the computation of the table $\gamma_n(n, i_n, j_n)$ for the reduction Γ_n with argument $\langle n, i_n, j_n\rangle$.

If no table is produced, then $G_n \uparrow$.

If a table is produced we define define the game G_n. The oracle will be querying M and hence asking question like "Is $z \in M_s$?." By definition of M, such question are of the form "Is $(x, r) \in M_s$?", and hence asking "Is $H(x) < r$?".

Let B_n be the set of first coordinates for such queries. Let $A_n = B_n - (\cup_{m<n} A_m)$. Then the set of functions h_n will be defined on the set A_n so that they are those bounded above by the function $\min\{K_s + 2, F_s\}$. That is, for all $x \in A_n$, $h_n(x) \leqslant \min\{K_s(x) + 2, F_s(x)\}$. Any such function has range within $\mathbb{N}$ and hence there are only finitely many such functions. The numbers q_n and p_n will be rationals less than 2^{-n-i_n-2}.

In the initial position d_0, we will simply define $h_n(x) = \min\{K_s(x) + 2, F_s(x)\}$, for all $x \in A_n$. To specify the initial position, we also ask that $q_n = p_n = 0$. Now we are ready to be able to specify the rationals p'_n and q'_n which we can also do at the same time as specifying the rules of movement.

The rules of play are that player 1 can move from a position (h_n, q_n, p_n) to (h'_n, q'_n, p'_n) such that $h'_n \leqslant h_n$, $q'_n - q_n = \sum_x 2^{-h'_n(x)} - \sum_x 2^{-h_n(x)}$, and $p'_n = p_n$.

Here the sums are taken as $\sum_{x\in A_n}$. In the construction, if we make this play, then for each x where $h'_n(x) < h_n(x)$ we will revise the value of $F_s(x)$ accordingly. This is possible as the range of the finite functions h'_n are in $\mathbb{N}$. The key idea is that this move will specify an answer to the oracle for M_{s+1}.

Similarly, player 2 can move from (h_n, q_n, p_n) to (h'_n, q'_n, p'_n) such that $h'_n \leqslant h_n$, $p'_n - p_n = \sum_x 2^{-h_n`(x)} - \sum_x 2^{-h_n(x)}$, and $q'_n = q_n$.

Notice that in this case, it our opponent who has dropped the value of $h'_n(x)$ and because the value of $K_s(x)$ changed, and hence some oracle question has chnaged its answer.

Since there are only finitely many finite function h_n, this will generate a finite set of rules and positions for the game.

To each function from B_n to $\mathbb{N}$, the table assigns an answer of the algorithm Γ_n to the question of whether the triple $\langle n, i_n, j_n\rangle$ belongs to U.

Now, we know that either the first player has an effective winning strategy forcing a "yes" answer (for "$\langle n, i_n, j_n\rangle \in U$") or the second has one forcing a "no" answer.

In the former case, we do nothing to U and in the latter, we put $\langle n, i_n, j_n\rangle$ into U_s.

End of Construction

We assume that for even s, we have $H_s = \min\{K_{\frac{s}{2}} + 2, F_{\frac{s}{2}}\}$, and for odd s, $H_s = \min\{K_{\frac{s-1}{2}} + 2, F_{\frac{s+1}{2}}\}$ Suppose that game G_n was defined at stage s. Then the v-th move of the game is determined by the value H_{2s+v}, for arguments on the set A_n. If the games on arguments different from n don't interfere with those for n, and the second player does not violate the prohibition to change the value $\sum_x 2^{-h_n(x)}$ too much, then Γ_n will be diagonalized on (n, i_n, j_n).

First, the games on values $m > n$ won't interfere, since A_m does not intersect B_n by construction. By induction, we can assume that the games for $m < n$ have ceased activity, and hence we have reached a stable value for i_n.

Thereafter, any change to j_n is as a result of an increase of $\sum_x 2^{-K_s(x)-2}$ by at least 2^{-n-i_n-2}. Since i_n is fixed, the value of j_n stabilizes. Once i_n and j_n are fixed, the table produced by the algorithm Γ_n is now fixed, and hence G_n is fixed. Thus we will diagonalize Γ_n.

Finally, we need to prove that H is a prefix-free Kolmogorov complexity. It is enough to show that $H \leqslant K + c$ for some constant c, as K is universal, and to show that H is computably enumerable, and $\sum_x 2^{-H(x)} \leqslant 1$.. The first two are immediate.

Note that $\sum_x 2^{-H(x)} = \sum_x 2^{-H_0(x)} + \sum_x \sum_s (2^{-H_{s+1}(x)} - 2^{-H_s(x)})$. Since $\sum_x (2^{-H_0(x)}) \leqslant \sum_x 2^{-F_0(x)}$, we see $\sum_x (2^{-H_0(x)}) \leqslant \frac{1}{4}$. Every increase in $\sum_s (2^{-H_{s+1}(x)} - 2^{-H_s(x)})$ occurs because either because of a decrease in the value of $K_s(x)$ or due to a move of player 1 in one of the games. For any n and i there may be several games in which player 2 has violated the game prohibition, but at most one game where the prohibition was not broken (by priority).

Muchnik's proof finishes by breaking the the pairs (x, s) into three parts. The first part have odd s, the seond even s where the prohibition is unbroken, and the third where the prohibition is violated.

Consider the sum $(2^{-H_{s+1}(x)} - 2^{-H_s(x)})$ from the first part. This is $\leqslant \sum_{x,s} (2^{-K_{s+1}(x)-2} - 2^{-K_s(x)-2})$. From the second part, this sum cannot exceed $\sum_{n,i} 2^{-n-i-2} = \frac{1}{4}$. From the third part, is again no greater than $\leqslant \sum_{x,s} (2^{-K_{s+1}(x)-2} - 2^{-K_s(x)-2})$. Finally, we note that

$$\leqslant \sum_{x,s} (2^{-K_{s+1}(x)-2} - 2^{-K_s(x)-2}) \leqslant \sum_x 2^{-K(x)-2} \leqslant \frac{1}{4},$$

we get $\sum_x 2^{-H(x)} \leqslant 1$. □

We remark that it is presently unknown whether there is a universal prefix-free machine relative to which $\overline{R}_K$ is tt-complete. It is also unknown if there is a universal machine relative to which the overgraph for monotone complexity is not tt-complete.

We remark that one of the original uses Muchnik made of the constructions above was to show that for each d there are strings σ and τ such that $K(\sigma) > K(\tau) + d$ and $C(\tau) > C(\sigma) + d$, Theorem 7.4.1.

19.4.3 The conditional case

The final c.e. set we examine is the one generated by conditional complexity. We consider the following overgraph.

$$M = \{(x, y, n) : Q(x|y) < n\},$$

for any Q such as K, Km or C.

Theorem 19.4.11 (Muchnik, An. A., [221])**.** *For any choice of Q, M is m-complete.*

Proof. Muchnik's proof is modeled upon the proof of Kummer's Theorem that $\overline{R}_C$ is tt-complete. We fix Q as C but the proof works with easy modifications for other complexities.

The construction will have a parameter d which can be worked out in advance, and known by the recursion theorem. For our purposes think of d in the following big enough to make everything work. We will construct a series of m-reductions g_x for $x \in [1, 2^d]$. Then for each z either we will know that z enters $\emptyset'$ computably, or there will be a unique y such that $g_x(z) = (x, y, d)$ and $x \in \emptyset'$ iff $g_x(z) \in M$. For some maximal x chosen infinitely often, this will then give the m reduction since on those elements which don't computably enter $\emptyset'$, g_x is (computable) and defined.
Construction For each active $y \leqslant s$, find the least $q \in [1, 2^p]$ with

$$(q, y, d) \notin M_s.$$

(Notice that such an x needs to exist since $\{q : (q, y, d) \in M\} < 2^d$.)

If q is new, (that is, $(q', y, d) \in M_s$ for all $q' < q$), find the least z with $z \notin \emptyset'[s + 1]$ and define

$$g_q(z) = (q, y, d).$$

Now for any v, if v enters $\emptyset'[s + 1]$, find the largest r, if any, with $g_r(v)$ defined. If one exists Find $\widehat{y}$ with $g_r(v) = (r, \widehat{y}, d)$. Declare that $\widehat{y}$ is no longer active.

(Therefore, we will do this exactly once for any fixed $\widehat{y}$ so the cost is modest, and known by the Recursion Theorem.)
End of Construction

Note that there must a largest $x \leqslant 2^d$ such that $\exists^\infty v(g_x(v) \in M)$. Call this x. We claim that g_x is the required m-reduction. Work in stages after which g_{x+1} enumerates nothing into M.

Given z, since g_x is defined on infinitely many arguments and they are assigned in order,we can go to a stage s where either z has entered $\emptyset'[s]$, or

$g_x(z)$ becomes defined, and $g_x(z) = (x, y, d)$ for some active y. $g_x(z)$ will be put into M should z enter $\emptyset'$ after s. The result follows. $\square$

References

[1] Allender, E., Buhrman, H., Koucký, M., van Melkebeek, D., and Ronneburger, D., *Power from Random Strings,* in *FOCS 2002,* IEEE (2002), 669-678.

[2] Allender, E., H. Buhrman, and M. Koucký, *What Can be Efficiently Reduced to the Kolmogorov-Random Strings?,* Annals of Pure and Applied Logic, 2006 to appear.

[3] Ambos-Spies, K. *On the Structure of the Polynomial Degrees of Recursive Sets,* Habilitationschrift, Universitat Dortmund, 1984.

[4] Ambos-Spies, K., *Antimotitic recursively enumerable sets,* Z. Math. Logik Grundlagen Math., Vol. 31 (1985) 461-477.

[5] K. Ambos-Spies, *Algorithmic randomness revisited*, in *Language, Logic and Formailzation of Knowledge,* Bibliotheca, 1998, 33-52.

[6] Ambos-Spies K., C. Jockusch, R. Shore, and R. Soare, *An algebraic decomposition of recursively enumerable degrees and the coincidence of several degree classes with the promptly simple degrees,* Trans. Amer. Math. Soc., Vol. 281 (1984), 109-128.

[7] Ambos-Spies, K., B. Kjos-Hanssen, S. Lempp and T. Slaman, *Comparing DNR and WWKL*, to appear.

[8] Ambos-Spies K., and A. Kučera, *Randomness in computability theory*, in *Computability Theory and its Applications*, (Cholak, Lempp, Lerman, Shore, eds) Contemporary Mathematics Vol. 257 (2000), 1-14.

[9] Ambos-Spies K., and E. Mayordomo, *Resource bounded measure and randomness*, in *Complexity, Logic and Recursion Theory,* (A. Sorbi, ed.) Marcel-Decker, New York, 1997, 1-48.

[10] Ambos-Spies, K., E. Mayordomo, Y. Wang, and X. Zheng, *Resource-bounded dense genericity, stochasticity and weak randomness*, Proceedings of the Thirteenth Symposium on Theoretical Aspects of Computer Science (STACS'96). Springer-Verlag Lecture Notes in Computer Science 1046, 63-74 (1996)

[11] Ambos-Spies K. and A. Nies, *The Theory of The Polynomial Time Many-One Degrees is Undecidable,* to appear.

[12] Ambos-Spies, K., W. Merkle, J. Reimann, and F. Stephan, *Hausdorff Dimension in Exponential Time*, Computational Complexity 2001, 210-217, IEEE Computer Society, 2001.

[13] Ambos-Spies, K., A. Nies, and R.A. Shore, *The Theory of the Recursively Enumerable Weak Truth Table Degrees is Undecidable,* Journal of Symbolic Logic, Vol. 57 (1992), 864-874.

[14] Ambos-Spies, K., K. Weihrauch and X. Zheng, *Weakly computable real numbers,* J. Complexity, Vol. 16 (4)(2000), 676-690.

[15] Artheya, K., J. Hitchcock, J. Lutz, and E. Mayordomo, *Effective strong dimension, algorithmic information, and computational complexity,* Proceedings of the Twenty-First Symposium on Theoretical Aspects of Computer Science (Montpellier, France, March 25–27, 2004), Springer-Verlag, 2004, pp. 632-643.

[16] Arslanov, M., R. F. Nadyrov, and V. D. Solov′ev. *A criterion for the completeness of recursively enumerable set s, and some generalizations of a fixed point theorem*, Izv. Vysš. Učebn. Zaved. Matematika, (4 (179)):3– 7, 1977.

[17] Arslanov, M., *On some generalizations of the fixed-point theorem*, Soviet Math. Vol. 25 (1981), 1-10.

[18] Arslanov M., *Degree structures in local degree theory*, in A. Sorbi, editor, *Complexity, logic, and recursion theory*, volume 187 of *Lecture Notes Pure Applied Mathematics*, 49-74. Marcel Dekker INC, New York, 1997.

[19] Ash, C. J. and J. F. Knight, *Computable Structures and the Hyperarithmetical Hierarchy,* Springer-Verlag, 2000.

[20] J. Balcazaar,J. Diaz, and J. Gabarro, *Structural Complexity*, Volumes 1 and 2, Springer Verlag (1987,1989).

[21] Barmpalias, G., and A. Lewis, *New proof of the Yu-Ding Theorem*, e-mail communication, May, 2005.

[22] Barmpalias, G., and A. Lewis, *On a question of Yu and Ding*, submitted.

[23] Barzdins, J., *Complexity of programs to determine whether natural numbers not greater than n belong to a recursively enumerable set,* Soviet Mathematics Doklady vol. 9 (1968), 1251-1254.

[24] V. Becher and G. Chaitin, *Another example of higher order randomness*, Fund. Inform. 51 (2002), 325–338.

[25] V. Becher, S. Daicz, and G. Chaitin, *A highly random number*, in *Combinatorics, Computability and Logic (Constanţa, 2001)* (C. S. Calude, M. J. Dinneen, and S. Sburlan, eds.), Springer Ser. Discrete Math. Theor. Comput. Sci., Springer, London, 2001, 55–68.

[26] V. Becher and S. Grigorieff, *Random reals and possibly infinite computations part I: randomness in* $\emptyset'$, J. Symbolic Logic 70 (2005), 891–913.

[27] Bedregal, B., and A. Nies, Lowness properties of reals and hyper-immunity. In *WOLLIC 2003*, volume 84 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2003. http://www.elsevier.nl/locate/entcs/volume84.html.

[28] Belegradek, O., *On algebraically closed groups*, Algebra i Logika, **13** No. 3 (1974), 813-816.

[29] Buhrman, H., T. Jiang, M. Li, and P. Vitanyi, *New applications of the incompressibility method: Part II,* Theoretical Computer Science, Vol. 235 (2000), 59–70.

[30] Bienvenu, L., and W. Merkle, *Reconciling data compression and algorithmic randomness,* to appear.

[31] Burgin, M, *Super-recursive Algorithms*, Speinger-Verlag Monographs in Computer Science, 2005.

[32] Cai, J. and J. Hartmanis, *On Hausdorff and topological dimensions of the Kolmogorov complexity of the real line,* Journal of Computing and System Sciences, Vol. 49 (1994), 605-619.

[33] Calude, C., *Information Theory and Randomness, an Algorithmic Perspective,* Springer-Verlag, Berlin, 1994. Second revised edition 2002.

[34] Calude, C., and Coles, R. *Program size complexity of initial segments and domination relation reducibility,* in *Jewels are Forever*, (J. Karhümaki, H. Mauer, G. Paŭn, G. Rozenberg, eds.) Springer-Veralg, New York, 1999, 225-237.

[35] Calude, C., Coles, R., Hertling, P., Khoussainov, B., *Degree-theoretic aspects of computably enumerable reals*, in *Models and Computability,* (ed. Cooper and Truss) Cambridge University Press, 1999.

[36] Calude, C., Hertling, P., Khoussainov, B., Wang, Y., *Recursively enumerable reals and Chaitin's* Ω *number*, in STACS '98, Springer Lecture Notes in Computer Science, Vol 1373, 1998, 596-606.

[37] Calude, C., and A. Nies, *Chaitin's* Ω *numbers and strong reducibilities*, Journal of Universal Computer Science, 1997.

[38] Calude, C., L. Staiger, S. A. Terwijn, *On partial randomness*, in preparation.

[39] Cantor, G., *Beiträge zur begründung transfiniten mengenlehre,* Math. Ann., Vol. 96, (1895), 481-512.

[40] Carathéodory, C., *Über das lineare massvon punktmengen- eine verallgemeinerung des längerbegriffs,* Nachrichten der K. Gesellschaft der wissenachaften zu Göttingen, Mathematisch-physicalische Klasse (1914), 404-426.

[41] Cenzer, D., Π^0_1 *classes in computability theory,* in *Handbook of Computability Theory*, volume 140 of *Stud. Logic Found. Math.*, pages 37–85. North-Holland, Amsterdam, 1999.

[42] Chaitin, G., *A theory of program size formally identical to information theory*, Journal of the Association for Computing Machinery 22 (1975), pp. 329-340.

[43] Chaitin, G. *Information-theoretical characterizations of recursive infinite strings,* Theoretical Computer Science, vol. 2 (1976), 45-48.

[44] Chaitin, G. *Incompleteness theorems for random reals,* Adv. in Appl. Math Vol. 8 (1987), 119-146.

[45] Chaitin, G., *Information, Randomness & Incompleteness*, 2nd edition, Series in Computer Science 8, World Scientific, River Edge, NJ, 1990.

[46] Chernov, A., An. A. Muchnik, A. E. Romashchenko, A. Shen, N. K. Vereshchagin, *Upper semi-lattice of binary strings with the relation "x is simple conditional to y",* Theor. Comput. Sci. 271(1-2): 69-95 (2002)

[47] Cholak, P, R. Coles, R. Downey, and E. Herrmann, *Automorphisms of the lattice of* Π^1_0 *classes: perfect thin classes and anr degrees*, *Trans. Amer. Math. Soc.* Vol. 353 (2001), 4899-4924.

[48] Cholak P., and Harrington, L., *Definable encodings in the computably enumerable sets*, Bull. Symbolic Logic, 6(2) (2000), 185-196.

[49] Cholak, P, R. Downey, and M. Stob, *Automorphisms of the lattice of recursively enumerable sets: promptly simple sets,* Trans. American Math. Society, **332** (1992) 555-570.

[50] Downey, R. and Chong, C., *Degrees bounding minimal degrees,* Math.Proc.Cambridge Phil. Soc., **105** (1989) 211-222.

[51] Downey, R. and Chong, C., *Minimal degrees recursive in 1-generic degrees,* Annals Pure and Appl. Logic, **48** (1990) 215-225.

[52] , Church, A., *On the concept of a random sequence*, Bulletin Amer. Math. Soc., Vol. 46 (1940), 130-135.

[53] Cohen, P., *The independence of the continuum hypothesis I,* Proc. Nat. Acad. Sci. A. S. A., Vol. 50, (1963), 1143-1148.

[54] Cooper, S. B., *Degrees of unsolvability complementary between recursively enumerable degrees,* Ann. Math. Logic, Vol. 4 (1972), 31-73.

[55] Cooper, S. B., *Minimal degrees and the jump operator,* J. Symbolic Logic 38 (1973) 249-271.

[56] Cooper, S. B., *A characterisation of the jumps of minimal degrees below 0'*, in *Computability, Enumerability, Unsolvability: Directions in Recursion Theory*, London Mathematical Society Lecture Notes, Cambridge University Press, 1996, pp. 81-92.

[57] B. F. Csima, *Applications of Computability Theory to Prime Models and Differential Geometry*, Ph.D. Dissertation, The University of Chicago, 2003.

[58] Csima, B. and A. Montalbán, *A minimal pair of K-degrees*, to appear in Proceedings Amer. Math. Soc.

[59] Daley,

[60] de Leeuw, K., Moore, E. F., Shannon, C. E., and Shapiro, N. *Computability by probabilistic machines,* Automata studies, pp. 183–212. Annals of mathematics studies, no. 34. Princeton University Press, Princeton, N. J., 1956.

[61] G. Davie, *Characterising the Martin-Löf random sequences using computably enumerable sets of measure one*, Inform. Process. Lett. 92 (2004), 157–160.

[62] Demuth, O., *On constructive pseudonumbers,* Comment. Math. Carolinae, Vol. 16 (1975), 315-331 (Russian).

[63] Demuth, O., *On some classes of arithmetical real numbers,* Comment. Math. Carolinae, Vol. 23 (1982), 453-465 (Russian).

[64] Demuth, O., *Reducibilities of sets based on constructive functions of real variables,* Comment. Math. Carolinae, Vol. 29 (1988), 143-156.

[65] Demuth, O., *Remarks on the structure of tt-degrees based on constructive measure theory,* Comment. Math. Carolinae, Vol. 29 (1988), 233-247.

[66] Demuth, O., *Remarks on Denjoy sets,* in *Proceedings of Summer School Conference Dedicated to the 90th Aniversay of Heyting,* Plenum Press, New York, 1990, 267-280.

[67] Downey, R., *Subsets of hypersimple sets,* Pacific J. Math., **127** (1987), 299-319.

[68] Downey, R., *On the universal splitting property,* Mathematical Logic Quarterly, vol. 43 (1997) 311-320.

[69] Downey, R., *Computability, definability, and algebraic structures*, in [73], 2003, 63-102.

[70] Downey, R., *Some recent progress in algorithmic randomness,* in *Proceedings of the 29 th Annual Conference on Mathematical Foundations of Computer Scence, Prague, August 2004* (invited paper) (J. Fiala, V. Koubek and J. Kratochvil eds), Springer-Verlag, Lecture Notes in Computer Science, Vol. 3153 (2004), 42-81.

[71] Downey, R., *Some Computability-Theoretical Aspects of Reals and Randomness,* in *The Notre Dame Lectures* (P. Cholak, ed) *Lecture Notes in Logic* Vol. 18, Association for Symbolic Logic, 2005, 97-148.

[72] Downey, R. and E. Griffiths, *On Schnorr randomness*, Journal of Symbolic Logic, Vol 69 (2) (2004), 533-554.

[73] Downey, R., Ding Decheng, Tung Shi Ping, Qiu Yu Hui, Mariko Yasuugi, and Guohua Wu (eds), *Proceedings of the 7th and 8th Asian Logic Conferences*, World Scientific, 2003, viii+471 pages.

[74] Downey, R., E. Griffiths and G. LaForte, *On Schnorr and computable randomness, martingales, and machines*, Mathematical Logic Quarterly, Vol. 50, No 6 (2004), 613-627.

[75] Downey, R., E. Griffiths, and S. Reid, *On Kurtz randomness*, Theoretical Computer Science Vol. 321 (2004), 249-270.

[76] Downey, R. and D. Hirschfeldt, *Aspects of Complexity* (Short courses in complexity from the New Zealand Mathematical Research Institute Summer 2000 meeting, Kaikoura) Walter De Gruyter, Berlin and New York, 2001.

[77] Downey, R., D. Hirschfeldt, and G. LaForte, *Randomness and reducibility*, in *Mathematical Foundations of Computer Science 2001* (Sgall, Pultr, P. Kolman, eds.), Lecture Notes in Computer Science 2136, Springer, 2001, 316–327. Final version in Journal of Computing and System Sciences, Vol. 68 (2004), 96-114..

[78] Downey, R., D. Hirschfeldt, and G. LaForte, *Undecidability of Solovay and other degree structures for c.e. reals*, in preparation.

[79] Downey, R., D. Hirschfeldt, and A. Nies, *Randomness, computability and density* SIAM Journal on Computing, 31 (2002) 1169–1183 (extended abstract in proceedings of STACS 2001 January, 2001. Lecture Notes in Computer Science, Springer-Verlag, (Ed A. Ferriera and H. Reichel, 2001, 195-201)).

[80] Downey, R., D. Hirschfeldt, J. Miller, and A. Nies, *Relativizing Chaitin's halting probability*, submitted.

[81] Downey, R., D. Hirschfeldt, A. Nies, and F. Stephan, *Trivial reals*, extended abstract in *Computability and Complexity in Analysis* Malaga, (Electronic Notes in Theoretical Computer Science, and proceedings, edited by Brattka, Schröder, Weihrauch, FernUniversität, 294-6/2002, 37-55),July, 2002. Final version appears in [73] 2003, 103-131.

[82] Downey, R., D. Hirschfeldt, A. Nies, and S. Terwijn, *Calibrating randomness,* Bulletin Symbolic Logic. Vol. 12 (2006), 411-491.

[83] Downey R., and C. Jockusch, *T- degrees, jump classes and strong reducibilities,* Trans. Amer. Math. Soc., Vol.. 301 (1987) 103-136.

[84] Downey, R., C. Jockusch, and M. Stob, *Array nonrecursive sets and multiple permitting arguments,* in *Recursion Theory Week* (Ambos-Spies, Muller, Sacks, eds.) Lecture Notes in Mathematics 1432, Springer-Verlag, Heidelberg, 1990, 141–174.

[85] Downey, R., C. Jockusch, and M. Stob, *Array nonrecursive degrees and genericity*, in *Computability, Enumerability, Unsolvability* (Cooper, Slaman, Wainer, eds.), London Mathematical Society Lecture Notes Series 224, Cambridge University Press (1996), 93–105.

[86] Downey, R. and G. LaForte, *Presentations of computably enumerable reals,* Theoretical Computer Science, Vol. 284 (2002), 539-555.

[87] Downey, R., G. LaForte, and A. Nies, *Enumerable sets and quasi-reducibility,* Annals of Pure and Applied Logic, Vol. 95 (1998), 1-35.

[88] Downey, R., S. Lempp, and R. Shore, *Jumps of minimal degrees below* $\mathbf{0}'$, J. London Math. Soc., **54**, No. 2 (1996), 417-439

[89] Downey, R., W. Merkle, and J. Reimann, *Schnorr Dimension,* to appear.

[90] Downey, R., and J. Miller, *A basis theorem for* Π^0_1 *classes of positive measure and jump inversion for random reals*, Proc. Amer. Math. Soc., Vol. 134 (2006), 283-288.

[91] Downey, R., J. Miller, and J Reimann, *Finite randomness,* in preparation.

[92] Downey, R., J. Miller, and L. Yu, *On the quantity of* K*-trivial reals,* in preparation.

[93] Downey, R., J. Reimann, and W. Merkle, *Schnorr dimension*, preliminary version appeared in *Computability in Europe, 2005*, Springer-Verlag Lecture Notes in Computer Science Vol. 3526 (2005) 96-105. Final version in *Mathematical Structures in Computer Science* Vol. 16 (2006), 789-811.

[94] Downey, R., and Remmel, J. B., *Classification of degree classes associated with r.e. subspaces*, Ann. Pure and Appl. Logic, **42** (1989) 105–125

[95] Downey R., and R. A. Shore, *Degree theoretical definitions of the* low$_2$ *recursively enumerable sets,* J. Symbolic Logic., Vol. 60 (1995) 727-756.

[96] Downey, R., and M. Stob, *Splitting theorems in recursion theory*, Annals of Pure and Applied Logic, **65** (1)((1993) 1-106).

[97] Downey R., and S. A. Terwijn, *Computably enumerable reals and uniformly presentable ideals*, Mathematical Logic Quarterly 48(1) (2002) 29–40.

[98] Downey R., and L. V. Welch, *Splitting properties of r. e. sets and degrees,* J. Symbolic Logic, Vol. 51 (1986) 88-109.

[99] Downey, R., G. Wu and X. Zheng, *Degrees of d.c.e. reals,* Mathematical Logic Quarterly. Vol. 50 (2004), 345-350.

[100] Epstein, R., R. Haas and R. Kramer, *Hierarchies of sets and degrees below* $\mathbf{0}'$, in *Logic Year 1979-1980* (ed. M. Lerman, J. Schmerl, and R. Soare) Lecture Notes in Mathematics, Vol. 859, Springer-Verlag, New York, 1981, 49-62.

[101] Ershov, Y., *A hierarchy of sets, Part I*, Algebra i Logika **7** (1968), 47-73 (Russian); Algebra and Logic **7** (1968), 24-43 (English translation)

[102] Ershov, Y., *A hierarchy of sets, Part II*, Algebra i Logika **7** (1968) 15-47 (Russian), Algebra and Logic **7** (1968), 212-232 (English Translation).

[103] Falconer, K., *Fractal geometry, mathematical foundations & applications*, Wiley & Sons, 1992.

[104] Feller, W., *Introduction to Probability Theory and its Applications,* Vol. 1, Wiley, London, 1950.

[105] Figueira, S., A. Nies, and F. Stephan, *Lowness properties and approximations of the jump,* in *Proceedings of the Twelfth Workshop of Logic, Language, Information and Computation (WoLLIC 2005),* Electronic Lecture Notes in Theoretical Computer Science, Vol. 143 (2006), 45-57.

[106] Figueira, S., F. Stephan, and G. Wu, *Randomness and universal machines*, to appear.

[107] Fortnow, L., *Kolmogorov complexity,* in [76], 73-86.

[108] L. Fortnow, R. Freivalds, W. Gasarch, M. Kummer, S. Kurtz, C. Smith, and F. Stephan, *On the relative size of learnable sets*, Theoretical Computer Science, 197 (1998), 139-156.

[109] Freidzon, R., *Families of recursive predicates of measure zero*, J. Soviet Math. 6 (1976), 449-455. Translated from Zap. Naučn. Sem. Leningrad. Otdel. Mat. Inst. Steklov. (LOMI) 32, (1972) 121-128.

[110] Friedberg, R., *Two recursively enumerable sets of incomparible degrees of unsolvability,* Proc. Nat. Acad. Sci. U. S. A., Vol. 43, (1957), 236-238.

[111] Friedberg, R., *A criterion for completeness of degrees of unsolvability,* J. Symb. Logic, Vol. 22 (1957), 159-160.

[112] Friedberg, R., *Three theorems on recursive enumeration,* J. Symb. Logic, Vol 23 (1958), 309-316.

[113] Friedberg, R. and H. Rogers, Jr., *Reducibility and completeness for sets of integers,* Z. Math. Logik Grund. Math., Vol. 5 (1959), 117-125.

[114] Gács, P., *On the symmetry of algorithmic information*, Soviet Mat. Dolk., Vol. 15 (1974), 1477-1480.

[115] Gács, P., *Exact Expressions for some Randomness Tests* Z. Math. Log. Grdl. M., Vol. 26 (1980), 385-394. (Short version: Springer Lecture Notes in Computer Science 67 (1979) 124-131)

[116] Gács, P., *On the relation between descriptional complexity and algorithmic probability,* Theoretical Computer Science, Vol. 22 (1983), 71-93.

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

[118] Gács, P., *Lecture Notes on Descriptional Complexity and Randomness*, Boston University, 1993-2005.

[119] Gaifmann, H., and M. Snir, *Probabilities over rich languages*, J. Symb. Logic, Vol. 47, (1982), 495-548.

[120] Garey M., and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness* (Freeman, San Francisco, 1979).

[121] Groszek, M., and T. Slaman, Π^0_1 *classes and minimal degrees*, Ann. Pure Appl. Logic, 87(2) (1997), 117-144

[122] Hanf, W., *The boolean algebra of logic*, Bull. Amer. Math. Soc., Vol. 81 (1975), 587-589.

[123] Harrington, L., and R. Soare, *Post's Program and incomplete recursively enumerable sets*, Proc. Natl. Acad. of Sci. USA 88 (1991), 10242-10246.

[124] Hausdorff, F., *Dimension und äußeres Maß*, Mathematische Annalen 79 (1919) 157–179.

[125] Hirschfeldt, D, A. Nies, and F. Stephan, *Martin-Löf oracles*, in preparation.

[126] Hirschfeldt, D., personal communication, Feb, 2001.

[127] Ho, Chun-Kuen, *Relatively recursive reals and real functions*, Theoretical Computer Science, Vol. 219 (1999), 99-120.

[128] Hitchcock, J., J. H. Lutz and S. A. Terwijn, *The arithmetical complexity of dimension and randomness*, manuscript, January 2003.

[129] Ishmukhametov, S., *Weak recursive degrees and a problem of Spector,* in *Recursion Theory and Complexity*, (ed. M. Arslanov and S. Lempp), de Gruyter, (Berlin, 1999), 81-88.

[130] Jockusch C. G., *The degrees of bi-immune sets,* Z. Math. Logik. Grundlagen Math., Vol. 15 (1972), 135-140.

[131] Jockusch C. G., *Three easy constructions of recursively enumerable sets,* in *Logic Year 1979-80* (Lecture Notes in Math, Vol. 859, (M. Lerman, J. H. Schmerl, and R. I. Soare, eds) Springer Verlag, Berlin, Heidelberg, New York, (1981), 83-91.

[132] Jockusch C. G., Fine degrees of word problems in cancellation semigroups, *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, Vol. 26 (1980), 93-95.

[133] Jockusch, C. G., *Degrees of generic sets,* in *Recursion Theory: its Generalizations and Applications,* (ed. Drake and Wainer), Cambridge University Press, (1980).

[134] Jockusch, C. G. , M. Lerman, R. I. Soare, and R. Solovay, *Recursively enumerable sets modulo iterated jumps and extensions of Arslanov's completeness criterion,* J. Symb. Logic, Vol. 54 (1989), 1288-1323.

[135] Jockusch, C. G. , and D. Posner, *Double jumps of minimal degrees,* J. Symb. Logic, Vol. 43 (1978), 715-724.

[136] Jockusch, C. G. and R. I. Soare, *Degrees of members of* Π_1^0 *classes*, Pacific J. Math. Vol. 40, (1972), 605-616.

[137] Jockusch, C. G. and R. I. Soare, Π_1^0 *classes and degrees of theories,* Trans. Amer. Math. Soc., Vol. 173, (1972), 33-55.

[138] Kamae, T., *Subsequences of normal sequences*, Israel J. Math. 16 (1973) 121–149.

[139] Katseff, H. P., *Complexity dips in random infinite binary sequences*, Information and Control 38 (1978) 258–263.

[140] Kautz, S., *Degrees of Random Sets,* Ph.D. Diss. Cornell, 1991.

[141] Khoussainov, B., *Randomness, computability, and algebraic specifications*, Annals of Pure and Applied Logic 91 (1998) 1–15.

[142] Khoussainov, B. and A. Nerode, *Automata Theory and its Applications*, Birkhäuser, Berlin, 2001.

[143] H. Ki and T. Linton, *Normal numbers and subsets of* $\mathbb{N}$ *with given densities*, Fundamenta Mathematicae 144 (1994) 163–179.

[144] Kjos-Hanssen, B., F. Stephan, and A. Nies. *On a question of Ambos-Spies and Kučera*, to appear.

[145] Kleene, S. C., *The origins of recursive function theory,* Ann. Hist. Comput., Vol. 3, (1981), 52-67.

[146] Kleene S. C., and E. Post, *The uppersemilattice of degrees of recursive unsolvability,* Ann. Math., Vol. 59, (1954), 379-407.

[147] Ko, Ker-I, *On the notion of infinite pseudo-random sequences*, Theoretical Computer Science, A, vol. 48 (1986), 9-33.

[148] Ko, Ker-I, *On the continued fraction representation of computable real numbers*, Theoretical Computer Science, A, 47 (1986), 299-313,

[149] Ko, Ker-I, *Complexity of Real Functions*, Birkhäuser, Berlin, 1991.

[150] Ko, Ker-I, and H. Friedman, *On the computational complexity of real functions,* Theoretical Computer Science, 20 (1982), 323-352.

[151] Kolmogorov, A.N., *Three Approaches to the Quantitative Definition of Information*, in *Problems of Information Transmission (Problemy Peredachi Informatsii)*, 1965, Vol. 1, 1-7.

[152] Kolmogorov, A.N. and V.A. Uspensky, *Algorithms and Randomness*, SIAM J. Theory Probab. Appl., Vol. 32 (1987), 389-412.

[153] Kraft, L. G., *A device for quantizing, grouping, and coding amplitude modulated pulses*, M.Sc. Thesis, MIT, 1949.

[154] Kripke, S., *"Flexible" predicates in formal number theory,* Proc. Amer. Math. Soc., Vol. 13 (1962), 647-650.

[155] Kučera, A., *Measure,* Π_1^0 *classes, and complete extensions of PA,* in *Springer Lecture Notes in Mathematics,* Vol. 1141, 245-259, Springer-Verlag, 1985.

[156] Kučera, A., *An alternative priority-free solution to Post's Problem, Proceedings, Mathematical Foundations of Computer Science,* (Ed. J. Gruska, B. Rovan, and J. Wiederman), Lecture Notes in Computer Science, Vol. 233, Springer-Verlag, (1986).

[157] Kučera, A., *On the use of diagonally nonrecursive functions,* in *Logic Colloquium, '87*, North-Holland, Amsterdam, 1989, 219-239.

[158] Kučera, A., *Randomness and generalizations of fixed point free functions,* in *Recursion Theory Week, Proceedings Oberwolfach 1989,* (K. Ambos-Spies, G. H. Müller and G. E. Sacks, eds.) Springer-Verlag, LNMS 1432 (1990), 245-254.

[159] Kučera, A., *On relative randomness,* Annals of Pure and Applied Logic, vol. 63, (1993), 61-67.

[160] Kučera, A., and S. Terwijn, *Lowness for the class of random sets,* Journal of Symbolic Logic, vol. 64 (1999), 1396-1402.

[161] Kučera, A. and T. Slaman, *Randomness and recursive enumerability,* SIAM Journal of Computing, Vol. 31 (2001), 199-211.

[162] Kučera, A. and T. Slaman, *Ideals in the Δ_2^0 degrees,* in preparation.

[163] Kummer, M., *Kolmogorov complexity and instance complexity of recursively enumerable sets,* SIAM Journal of Computing, Vol. 25 (1996), 1123-1143.

[164] Kummer, M., *On the complexity of random strings*, (extended abstract) in *STACS, '96*, Springer-Verlag Lecture Notes in Computer Science, Vol 1046 (11996), 25-36.

[165] Kurtz, S., *Randomness and Genericity in the Degrees of Unsolvability*, Ph. D. Thesis, University of Illinois at Urbana, 1981.

[166] Lachlan, A., *Lower bounds for pairs of recursively enumerable degrees,* Proceedings of the London Math. Society, Vol. 16, (1966) 537-569.

[167] Lachlan, A., *A recursively enumerable degree which will not split over all lesser ones,* Ann. Math. Logic, Vol. 9, (1975), 307-365.

[168] Lachlan, A., *Bounding minimal pairs,* J. Symb. Logic, Vol. 44, (1979), 626-642.

[169] Ladner, R., *A completely mitotic nonrecursive recursively enumerable degree, Trans. Amer. Math. Soc.*, Vol. 184, (1973), 479-507.

[170] Ladner, R., and L. Sasso, Jnr., *The weak truth table degrees of recursively enumerable sets,* Annals of Mathematical Logic, **8** (1975), 429-448.

[171] Lathrop J., and J. H. Lutz, *Recursive computational depth*, Information and Computation 153 (1999) 139–172.

[172] Lebesgue, H, *Lecons sur L'intégration et la Recherché des Fonctions Primitives,* Gauthier-Vallars, 1904.

[173] Lempp, S. and M. Lerman, *Priority arguments in iterated trees of strategy,* in *Recursion Theory Week* (Proceedings Oberwolfach (1989)), Springer–Verlag, New York (1990).

[174] Lempp, S. and M. Lerman, *The decidability of the existential theory of the recursively enumerable degrees with jump relations,* Advances in Mathematics, Vol 120 (1996), 1-142.

[175] Lerman, M., *Admissible ordinals and priority arguments,* in *Cambridge Summer School in Mathematical Logic* (A. Mathias and H. Rogers (eds.)) Lecture notes in Math. **337**, Springer-Verlag New York (1973) 311-344.

[176] Lerman, M., *Degrees of Unsolvability*, Springer–Verlag, New York (1983).

[177] , L. A. Levin, *Some Theorems on the Algorithmic Approach to Probability Theory and Information Theory*, Dissertation in Mathematics, Moscow, 1971.

[178] Levin. L., *On the notion of a random sequence*, Soviet Math. Dokl. 14 (1973) 1413–1416.

[179] Levin, L., *Laws of information conservation (non-growth) and aspects of the foundation of probability theory,* Problems Informat. Transmission, Vol. 10 (1974), ‘ 206-210.

[180] Levin, L., *Measures of complexity of finite objects (axiomatic description),* Soviet Mathematics Dolkady, vol. 17 (1976), 552-526.

[181] Levy, P., *Theorie de l'Addition des Variables Aleantoires,* Gauthier-Villars, 1937.

[182] Lewis, A., *Minimal complements for degrees below 0'*, J. Symb. Logic, to appear.

[183] Lewis, A., *A single minimal complement for the computably enumerable degrees,* Trans. Amer. Math. Soc, to appear.

[184] A. E. M. Lewis and G. Barmpalias, *Randomness and the Lipschitz degrees of computability*, to appear.

[185] Li, Ming and Vitanyi, P., *Kolmogorov Complexity and its Applications*, Springer-Verlag, 1993.

[186] Li, Ming and Vitanyi, P., *A new approach to formal language theory by Kolmogorov complexity,* SIAM J. Comput., 24:2(1995), 398-410.

[187] Li, Ming and Vitanyi, P., *Statistical properties of finite sequences with high Kolmogorov complexity,* Math. System Theory, 27(1994), 365-376.

[188] Lieb, E., D. Osherson, and S. Weinstein, *Elementary proof of a theorem of Jean Ville,* to appear, manuscript September 6, 2005.

[189] Loveland, D. *A variant of the Kolmogorov concept of complexity*, Information and Control, vol. 15 (1969), 510-526.

[190] Lutz, J., *Category and measure in complexity classes*, SIAM J. Comput. 19 (1990) 1100–1131.

[191] Lutz, J., *Almost everywhere high nonuniform complexity*, J. Comput. System Sci. 44 (1992) 220–258.

[192] Lutz, J., *Dimension in complexity classes*, Proc. 15th Conference on Computational Complexity, 158–169, IEEE Computer Society, 2000.

[193] Lutz, J., *The dimensions of individual strings and sequences*, Information and Computation, Vol 187 (2003), pp. 49-79. (Preliminary version: *Gales and the constructive dimension of individual sequences*, in: U. Montanari, J. D. P. Rolim, E. Welzl (eds.), Proc. 27th International Colloquium on Automata, Languages, and Programming, 902–913, Springer, 2000.)

[194] Lutz, J., *Effective fractal dimensions,* Mathematical Logic Quarterly Vol. 51 (2005), pp. 62-72.

[195] Maass, W., *Characterization of the recursively enumerable sets with supersets effectively isomorphic to all recursively enumerable sets,* Trans. Amer. Math. Soc., Vol. 279 (1983), 311-336.

[196] Macintyre, A., *Omitting quantifier free types in generic structures*, Journal of Symbolic Logic, **37** (1972), pp. 512-520.

[197] Martin, D., *Classes of recursively enumerable sets and degrees of unsolvability,* Z. Math. Logik Grundlag. Math. 12 (1966) 295-310.

[198] Martin-Löf, P., *The definition of random sequences*, Information and Control, 9 (1966), 602-619.

[199] Martin-Löf, P., *Complexity oscillations in infinite binary sequences,* Z. Wahrscheinlichkeittheorie verw. Gebiete, Vol. 19 (1971), 225-230.

[200] E. Mayordomo, *A Kolmogorov complexity characterization of constructive Hausdorff dimension.*, Information Processing Letters 84 (2002) 1–3.

[201] Medvedev, Y., *On nonisomorphic recursively enumerable sets,* Dokl. Acad. Nauk., Vol. 102 (1955), 211-214.

[202] Merkle W., *The complexity of stochastic sequences,* Preliminary version in CCC2003, final version to appear.

[203] Merkle W., *The Kolmogorov-Loveland stochastic sequences are not closed under selecting subsequences,* J. Symbolic Logic, Vol. 68 (2003), 1362-1376.

[204] Merkle W., and N. Mihailovic, *On the construction of effective random sets*, in Proceedings MFCS 2002, (ed. Diks and Rytter) Springer-Verlag LNCS 2420.

[205] Merkle, W., N. Mihailovic, and T. Slaman, *Some results on effective randomness*, to appear.

[206] Mihailović, N., personal communication, November, 2006.

[207] Merkle, W., J. Miller, A. Nies, J. Reimann, and F. Stephan, *Kolmogorov-Loveland randomness and stochasticity*, to appear, Annals of Pure and Applied Logic.

[208] Miller, J., *Deriving Muchnik's Theorem from Solovay's manuscript,* unpublished notes.

[209] Miller, J., *Kolmogorov random reals are 2-random*, J. Symbolic Logic, Vol. 69, (3) (2004), 907-913

[210] Miller, J., *Solution to the question of whether* $\{x : K(x) < |x| + K(|x|) - c\}$ *is* Σ^0_1, e-mail communication, 13th January, 2005.

[211] Miller, J., *A hyperimmune-free weakly 2 random real*, e-mail communication, 14th April, 2005.

[212] Miller, J., *The K-degrees, low for K degrees, and weakly low for K oracles,* in preparation.

[213] Miller, J. *Contrasting plain and prefix-free Kolmogorov complexity,* in preparation.

[214] Miller, J., *The K-degrees, low for K-degrees and weakly low for K sets,* in preparation.

[215] J. S. Miller and A. Nies, *Randomness and computability: open questions*, Bull. Symb. Logic. 12 no 3 (2006) 390-410.

[216] J. S. Miller and L. Yu, *On initial segment complexity and degrees of randomness*, to appear, Trans. Amer. Math. Soc..

[217] J. S. Miller and L. Yu, *Oscillation in the initial segment complexity of random reals*, to appear.

[218] Miller, W., and D. A. Martin. The degree of hyperimmune sets. *Z. Math. Logik Grundlag. Math.*, 14:159–166, 1968.

[219] Mostowski, A., *A generalization of the incompleteness theorem,* Fund. Math., Vol. 49 (1961), 205-232.

[220] Muchnik, A. A., *On the unsolvability of the problem of reducibility in the theory of algorithms,* Dokl. Akad. Nauk. S. S. S. R., Vol. 106, (1956), 194-197.

[221] Muchnik, An. A., and S. P. Positelsky, *Kolmogorov entropy in the context of computability theory*, Theor. Comput. Sci., Vol. 271 (2002), 15-35.

[222] Muchnik, An. A., A. Semenov, and V. Uspensky, *Mathematical metaphysics of randomness,* Theor. Comput. Sci., Vol 207(2), (1998), 263-317.

[223] Matijacevic, Y., *Enumerable sets are Diophantive,* Dokl. Akad. Nauk. S. S. S. R., Vol. 191 (1970), 279-282.

[224] Ng Keng Meng, MSc Thesis, National University of Singapore, in preparation.

[225] Nies, A., *Reals which compute little*, to appear.

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

[227] Nies, A., *Each Low(CR) set is computable*, typeset manuscript, January 2003.

[228] Nies, A., *Ideals in the recursively enumerable degrees*, in preparation.

[229] Nies, A., *Lowness for computable and partial computable randomness,* in preparation.

[230] Nies A., and F. Stephan, *Degrees of Recursively Random Reals*, unpublished manuscript, February 2003.

[231] Nies, A., *Computability and Randomness*, Oxford University Press, in preparation.

[232] Nies, A., F. Stephan and S. A. Terwijn, *Randomness, relativization, and Turing degrees*, Journal of Symbolic Logic, Vol. 70(2) (2005), 515-535.

[233] Odifreddi, P., *Classical Recursion Theory,* North-Holland, 1990.

[234] Odifreddi, P., *Classical Recursion Theory, Vol. 2,* North-Holland, North-Holland, 1999.

[235] Oxtoby, J., *Measure and Category*, 2nd edition, Springer 1980.

[236] Paris, J., *Measure and minimal degrees,* Annals of Mathematical Logic, Vol. 11 (1977), 203-216.

[237] Paris, J., *Survey of Results on Measure and Degrees,* unpublished notes.

[238] E. Post, *Recursively enumerable sets of integers and their decision problems,* Bulletin Amer. Math. Soc., Vol. 12, (1944), 248-316.

[239] Pour-El, M., *From axiomatics to intrinsic characterization, some open problems in computable analysis*, Theoretical Computer Science, A., to appear.

[240] Pour-El, M. and I. Richards, *Computability in Analysis and Physics*, Springer-Verlag, Berlin, 1989.

[241] Raichev, A., Ph.D. thesis, University of Wisconsin-Madison, 2005.

[242] Raichev, A., *Relative randomness and real closed fields*, J. Symbolic Logic, Vol. 70 (2005), 319-330.

[243] Raisonnier, J., *A mathematical proof of S. Shelah's theorem on the measure problem and related results,* Israel Journal of Mathematics, vol. 48 (1984), 48-56.

[244] Reimann, J., *Computability and Dimension,* unpublished notes, University of Heidelberg, 2004

[245] Reimann, J. and T. Slaman, *Measures and their random reals,* to appear.

[246] Reimann, J. and T. Slaman,

[247] Reimann, J. and S. Terwijn, *A lower cone of broken constructive Hausdorff dimension,* unpublished notes, June 2003.

[248] Rettinger, R., X. Zheng, R. Gengler and B. von Braunmühl, *Weakly computable real numbers and total computable real functions*, COCOON01, 2001, Guilin, China. Lecture Notes in Computer Science **2108**, 586-595.

[249] Rice, H., *Classes of recursively enumerable sets and their decision problems,* Trans. Amer. Math. Soc., Vol. 74, (1953) 358-366.

[250] Rice, H., *Recursive real numbers*, Proceedings of the American Mathematical Society, 5 (1954), 784-791.

[251] Reid, S., *A Class of Algorithmically Random Reals*, MSc Thesis, Victoria University, Wellington, 2004.

[252] Robinson, R. W., *Jump restricted interpolation in the recursively enumerable degrees,* Annals of Math., Vol 93 (1971), 586-596.

[253] Rogers, H. Jnr., *Theory of Recursive Functions and Effective Computability,* McGraw Hill, New York, 1968.

[254] Royden, H., —em Real Analysis, Macmillan, New York, 1968.

[255] Ryabko, B. Ya., *Coding of combinatorial sources and Hausdorff dimension,* Soviet Math. Doklady, Vol. 30 (1984), 219-222.

[256] Ryabko, B. Ya., *Noiseless coding of combinatorial sources,* Problems of Information Transmission, Vol. 22 (1986), 170-179.

[257] Ryabko, B. Ya., *Algorithmic approach to the prediction problem,* Problems of Information Transmission, Vol. 29 (1993), 186-193.

[258] Ryabko, B. Ya., *The complexity and effectiveness of prediction problems,* J. Complexity, Vol. 10 (1994), 281-295.

[259] Sacks, G., *On the degrees less than* $\mathbf{0}'$*,* Annals of Math., Vol. 77, (1963) 211-231.

[260] G. Sacks, *Degrees of Unsolvability*, Princeton University Press, 1963.

[261] Sacks, G., *The recursively enumerable degrees are dense,* Annals of Math., Vol. 80 (1964) 300-312.

[262] Sacks, G., *A maximal set that is not complete,* Michigan Math. J., Vol. 11 (1964), 193-205.

[263] Salomaa, A., *Computation and Automata*, Cambridge Univ. Press (1985).

[264] Schnorr, C. P., *A unified approach to the definition of a random sequence,* Mathematical Systems Theory, 5 (1971), 246-258.

[265] Schnorr, C. P., *Zufälligkeit und Wahrscheinlichkeit*, Springer-Verlag Lecture Notes in Mathematics, Vol 218, 1971, Springer-Verlag, New York.

[266] Schnorr, C. P., *Process complexity and effective random tests*, Journal of Computer and System Sciences, vol. 7 (1973), 376-388.

[267] Scott, D., *Algebras of sets binumerable in complete extensions of arithmetic,* in Proc. Symp. Pure and Appl. Math., Vol. 5 (1962), 357-366.

[268] Scott, D. and S. Tennenbaum, *On degrees of complete extensions of arithmetic,* Not. A. M. S., Vol. 7 (1960), 242-243.

[269] Shannon, C. E., *The mathematical theory of communication*, Bell System Tech. J. 27 (1948) 379–423, 623–656.

[270] Shoenfield, J. *On the degrees of unsolvability*, Annals of Math., Vol. 69 (1959), 644-653.

[271] Shoenfield, J., *Undecidable and creative theories,* Fund. Math., Vol. 49, (1961), 171-179.

[272] Shoenfield, J., *Application of model theory to degrees of unsolvability,* in *Symposium on the Theory of Models* (J. Addison, L. Henkin, A. Tarski, (eds.)), North-Holland, Amsterdam (1965) 359-363.

[273] Shoenfield, J., *Degrees of Unsolvability*, Mathematical Stuidies 2, North-Holland, Amsterdam, 1971.

[274] Silver, J.,

[275] Simpson, S., *Mass problems and randomness*, Bull. Symb. Logic, Vol. 11 (2005), 1-27.

[276] Soare, R., *Recursion theory and Dedekind cuts*, Transactions of the American Mathematical Society, 140 (1969), 271-294.

[277] Soare, R., *Cohesive sets and recursively enumerable Dedekind cuts*, Pacific Journal of Mathematics, vol. 31, no. 1 (1969), 215-231.

[278] Soare, R., *Automorphisms of the lattice of recursively enumerable sets I: maximal sets,* Annals of Math. (2), 100 (1974) 80-120.

[279] Soare, R., *Tree arguments in recursion theory and the* $\mathbf{0}'''$ *priority method,* in: *Recursion Theory,* (Ed. A. Nerode and R.A. Shore), Proc. Symp. Pure Math., American Mathematical Society, Providence, (1985), 53-106.

[280] Soare, R., *Recursively enumerable sets and degrees* (Springer, Berlin, 1987).

[281] R. Soare, *Computability Theory and Differential Geometry*, bulletin Symbolic Logic, to appear.

[282] Solomonoff, R., *A formal theory of inductive inference, part 1 and part2,* Information and Control, 7 (1964), 224-254.

[283] Solovay, R., *A model of set theory in which every set of reals is Lebesgue measurable*, Annals of Mathematics 92 (1970) 1–56.

[284] Solovay, R., *Draft of paper (or series of papers) on Chaitin's work*, unpublished notes, May, 1975, 215 pages.

[285] Solovay, R., *On Random r.e. sets*, in *Non-Classical Logic, Model Theory and Computability*, (A. I. Arruda et al., eds), North Holland, 1977, 283-307.

[286] Spector, C., *On the degrees of recursive unsolvability,* Annals of Math., Vol. 64 (1956), 581-592.

[287] Spector, C., *Measure-theoretic construction of incomparible hyperdegrees,* em J. Symb. Logic, vol 23 (1958), 280-288.

[288] Staiger, L., *Kolmogorov complexity and Hausdorff dimension,* Information and Computation, Vol. 103 (1993), 159-194.

[289] Staiger, L., *A tight upper bound on Kolmogorov complexity and uniformly optimal prediction,* Theorey of Computing Sciences, Vol. 31 (1998), 215-229.

[290] Staiger, L., *Constructive dimension equals Kolmogorov complexity*, Research Report CDMTCS-210, University of Auckland, January 2003.

[291] Stephan, F., personal communication.

[292] Stephan, F., *Martin-Löf random sets and PA-complete sets,* ASL Lecture Notes in Logic, Vol. 27 (2006), 342-348.

[293] Stephan, F. and G. Wu, *Presentations of K-Trivial Reals and Kolmogorov Complexity*, in Proceedings of *New Computational Paradigms: First Conference on Computability in Europe, CiE 2005*, Springer LNCS Vol. 3526 (2005), 461-469.

[294] Stephan, F.

[295] Stephan, F. and Yu, L., *Lowness for weakly 1-generic and Kurtz-random* in Proceedings of *Theory and Applications of Models of Computation: Third International Conference, TAMC 2006*, Beijing, China, May 15-20, 2006. Springer LNCS, Vol. 3959 (2006), 756764.

[296] Stillwell, J., *Decidability of "almost all" theory of degrees,* J. Symb. Logic, Vol. 37 (1972), 501-506.

[297] Stob, M., *Index sets and degrees of unsolvability,* Pacific J. Math., Vol. 100 (1982), 241-248.

[298] Sullivan, D., *Entropy, Hausdorff measures old and new, and limit sets of geometrically finite Kleinian groups,* Acta Mathematica, Vol. 153 (1984), 259-277.

[299] Tadaki, K., *A generalization of Chaitin's halting probability* Ω *and halting self-similar sets,* Hokkaido Math. J., Vol. 32 (2002), 219-253.

[300] Terwijn, S. *Computability and Measure*, Ph. D. Thesis, University of Amsterdam, 1998.

[301] Terwijn, S., *Computability and measure*, PhD thesis, University of Amsterdam/ILLC, 1998.

[302] Terwijn, S., *On the quantitative structure of* Δ^0_2, in: U. Berger, H. Osswald, and P. Schuster (eds.), Reuniting the Antipodes — Constructive and Nonstandard Views of the Continuum, Symposium Proceedings, Italy, 1999. Synthèse Library, Kluwer, 2000, 271–284.

[303] Terwijn, S., *Complexity and Randomness,* Notes for a course given at the University of Auckland, March 2003. report CDMTCS-212, University of Auckland.

[304] Terwijn, S., and D. Zambella, *Algorithmic randomness and lowness,* Journal of Symbolic Logic, vol. 66 (2001), 1199-1205.

[305] Tricot, P., *Two definitions of fractional dimension,* Math. Proc. Cambridge Phil. Soc., Vol. 91 (9182), 57-74.

[306] Turing, A., *On computable numbers with an application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society, 42 (1936), 230-265, correction in Proceedings of the London Mathematical Society, 43 (1937), 544-546.

[307] Uspensky, V., *Some remarks on r.e. sets,* Z. Math. Logik. Grund. Math., Vol. 3 (1957), 157-170.

[308] Uspensky, V., *Complexity and entropy: an introduction to the theory of Kolmogorov complexity,* in *Koomogorov Complexity and Computational Complexity,* (O. Watanabe, ed.), Springer-Verlag, New York, 1992.

[309] Uspensky, V., A. Semenov and A.Kh. Shen, *Can an Individual Sequence of Zeros and Ones be Random?*, Russian Math. Surveys, Vol. 45 (1990), 121-189.

[310] Uspensky, V., and A.Kh. Shen, *Relations between varieties of Kolmogorov complexities*, Math. Systems Theory, Vol. 29 (1996), 271-292.

[311] Vereshchagin, N., *A computably enumerable undecidable set with low prefix complexity: a simplified proof*, Electronic Colloquium on Computational Complexity, Revision 01 of Report TR01-083.

[312] Ville, J., *Ĕtude critique de la concept du collectif*, Gauthier-Villars, 1939.

[313] Vitanyi, P., *Randomness*, In: *Matematica, Logica, Informatica*, Volume 12 of the Storia del XX Secolo, to be published by the Instituto della Enciclopedia Italiana.

[314] van Lambalgen, M., *Random Sequences,* Ph. D,. Diss. University of Amsterdam, 1987.

[315] van Lambalgen, M., *The axiomatization of randomness,* JSL.

[316] Volchan, S. *The Algorithmic Theory of Randomness*, manuscript, 2001.

[317] von Mises, R., *Grundlagen der Wahrscheinlichkeitsrechnung*, Math. Z., vol. 5 (1919), 52-99.

[318] Wang, Y., *Randomness and Complexity*, PhD Diss, University of Heidelberg, 1996.

[319] Wang, Y., *A separation of two randomness concepts*, Information Processing Letters 69(3) (1999) 115–118.

[320] Weihrauch, K., *Computability*, Springer-Verlag, Berlin, 1987.

[321] Weihrauch, K., and Zheng, X., *Arithmetical hierarchy of real numbers*, MFCS'99, Sklarska Poreba, Poland, September, 1999, 23-33.

[322] Wu, Gouhua, *Presentations of computable enumerable reals and the initial segment of computable enumerable degrees,* in Proceedings of COCOON' 01, Guilin, China, 2001.

[323] Yates, C. E. M., *A minimal pair of recursively enumerable degrees,* J. Symbolic Logic, Vol. 31 (1966), 159-168.

[324] Yates, C. E. M., *The degrees of index sets II,* Trans. Amer. Math. Soc., Vol. 135, (1969), 249-266.

[325] Yates, C. E. M., *Initial segments of the degrees of unsolvability, Part II: Minimal degrees*, J. Symbolic Logic, Vol 35 (1970), 243-266.

[326] Yu Liang and Ding Decheng, *There is no sw-complete c.e. real*, Journal of Symbolic Logic, to appear.

[327] Yu Liang and Ding Decheng, *There are* $2^{\aleph_0}$ *many* H*-degrees in the random reals*, Proc. Amer. math. Soc. Vol. 132 (2004), 2461-2464.

[328] Yu Liang, Ding Decheng, and R. Downey, *The Kolmogorov complexity of the random reals*, Annals of Pure and Applied Logic, Volume 129, Issues 1-3 , (2004), 163-180

[329] Zambella, D., *On sequences with simple initial segments*, ILLC technical report, ML-1990-05, University of Amsterdam, 1990.

[330] Zheng, X., *On the Turing degrees of weakly computable real numbers*, Journal of Logic and Computation **13** (2003), 159-172.

[331] Ziegler, M., *Algebraisch abgeschlossen gruppen* in *Word Problems II, The Oxford Book*, ed. S.I. Adian, W.W. Boone, and G. Higman, North Holland, 1980, pp.449-576.

[332] Zvonkin A. K., and L.A. Levin, *The complexity of finite objects and the development of concepts of information and randomness by the theory of algorithms*, Russian Math. Surveys, 25(6):83-124 (1970).

Index