Some results and applications of computability theory

By

Ethan McCarthy

A dissertation submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

(Mathematics)

at the

UNIVERSITY OF WISCONSIN—MADISON

2018

Date of final oral examination: April 23, 2018

The dissertation is approved by the following members of the Final Oral Committee:

Professor J. S. Miller, Professor, Mathematics

Professor U. Andrews, Assistant Professor, Mathematics

Professor M. I. Soskova, Assistant Professor, Mathematics

Professor J. Ellenberg, Professor, Mathematics

Abstract

In this work we pursue various directions of research in computability theory, particularly applications to effective symbolic dynamics, algorithmic randomness, and computable analysis.

In our first direction, which we present in the second chapter, we study the enumeration degrees of sets enumeration reducible to their complements. We provide several natural characterizations of this class, including as the e-degrees of complements of maximal anti-chains on the graph of integer strings, and as the e-degrees of enumeration pointed binary trees. As an application, we obtain a characterization of the Turing upper cones of closed sets. Finally, we show how these characterizations can be used to obtain a complete characterization of the Turing degree spectra of minimal subshifts.

In the third chapter, we study the class of strongly jump traceable sets (SJT). This class has been characterized as the MLR diamond classes of superhigh, superlow, and of ω-c.e. We show that for the cases of superlow and ω-c.e., randomness doesn't play an essential role in the characterization: SJT is exactly the class of reals computable in every superlow DNC, equivalently in every ω-c.e. DNC. In the superhigh case, however, the class of reals computable in every superhigh DNC are the computable reals.

In the fourth chapter, we study the class of strong difference randoms (SDR), defined as the class of randoms passing the Solovay condition on all difference tests. Members of SDR satisfy computability properties similar to 2-randoms: array computability, hyperimmunity, bounding 1-generics, etc. We introduce a lowness notion, implicitly present in the work of Nies, which strengthens ω-c.e.-jump domination, and which characterizes

the strong difference randoms among the 1-randoms. We also characterize SDR using strict Demuth tests.

In the final chapter, we study the derivatives of computable functions. A derivative will densely often take computable values, so we look at what complexities can be achieved almost-everywhere. We uncover an interaction between analytic conditions imposed on a function and the possible complexities of its slopes under several computability-theoretic notions of reduction.

Acknowledgements

I would like to thank some of the professors at Madison who taught me mathematical logic: Steffen Lempp, Uri Andrews, Mariya Soskova, Mingzhong Cai, Rutger Kuyper. Each provided me with instruction, mentoring or support far exceededing any expectation placed on them.

In addition, a number of other professors at Madison have provided me with mentoring and support, special thanks to Daniel Erman and Jordan Ellenberg. Outside of Madison, I am particular indebted to Denis Hirschfeldt, Arno Pauly, Reed Solomon, and Andy Lewis-Pye.

There are numerous teachers across the big lake to thank for contributing to my mathematical education, but I would especially like to thank my parents, who would have supported me in any direction I could have pursued. Thanks to my closest friends and my family for their love and support.

For my first several years at Madison I had the privilege of looking up to several brilliant senior students who were always willing to share not only knowledge and experience, but also company and conversation; I am especially thankful to Mushfeq Khan and Ashutosh Kumar. And thanks to all my fellow students in logic throughout the rest of my time at Madison: Turbo, Reese, Paul, Tamvana, Tejas, Iván, James.

There are many people who have made Van Vleck Hall a special place to study as a graduate student, but I was particularly lucky throghout my stay in Madison to share working spaces and friendship with many wonderful office mates: Eric, Christian, Zhou, Ting-Ting, Kejia, Jiaxin, Ernest, Yunbai.

Finally, I would like to thank Joe Miller. Joe's intellect combined with his mathematical taste and instinct would alone make him an invaluable advisor, but I am grateful most of all for his tireless devotion and generosity in advising his students. Joe was there for me every single week, regardless of circumstance, I could not have asked for a better mentor.

Contents

Chapter 1

Introduction

The focus of this work is *computability theory* and its applications. Computability theorists quantify the *computational content* of mathematical objects, constructions, and principles.

Computability theory begins by formalizing notions of computation betweens subsets of natural numbers, which extends to any mathematical objects that can be effectively coded by natural numbers. Throughout this work, we will use standard notation and rely on the basic results and development of computability and Turing reducibility as presented in standard texts such as [56, 62]. We are also particularly influenced by [17].

This work contains results occupying several distinct areas of computability theory: the enumeration degrees and their applications to effective mathematics, algorithmic randomness and its interactions with computability theory, and computable analysis. We will briefly summarize the results of each chapter in this introduction, terms left undefined here will be defined in the chapters where they apply.

1.1 Cototal enumeration degrees

An active area of research in theoretical computer science concerns the computational content of dynamical systems. Of central importance are *effective symbolic systems* such

as Turing machines, cellular automata, and subshifts (see the introduction to [12] for a survey). A *subshift* is a nonempty class of infinite binary sequences, closed topologically as a subset of 2^ω (Cantor space), and closed under the *shift operator* deleting the first bit from each sequence. A subshift is *minimal* if it does not properly contain any other subshift. The *Turing degree spectra* of a subshift is the collection of Turing degrees obtained by ranging over its points, which we consider as providing a measure of its computational content. See [6–8, 18, 29, 32, 33, 61] for the development of this area. In Chapter 2, we give a complete characterization of the Turing degree spectra of minimal subshifts:

Theorem 2.24. A collection of Turing degrees is the spectra of minimal subshift if and only if it is the enumeration cone of a cototal set.

The *enumeration cone of* X, denoted $\mathcal{E}(X)$, is the collection of Turing degrees of enumerations of X. We say that $X \leq_e Y$ if $\mathcal{E}(Y) \subseteq \mathcal{E}(X)$ [59]. The $\leq_e$-equivalence classes form the *enumeration degrees* $\mathcal{D}_e$. Alternatively, an enumeration reduction $X \leq_e Y$ can be viewed as a *partial information* computation which only uses and only computes positive set membership information. Unlike Turing reduction, enumeration reducibility distinguishes between X and its complement $\overline{X}$. A set X is *cototal* if $X \leq_e \overline{X}$ [1]. The *cototal degrees* are the degrees of cototal sets. In Chapter 2, we greatly expand the study of the cototal degrees, giving several characterizations:

Theorem 2.3, 2.15, 2.24. The class of cototal enumeration degrees is given by:

1. The e-degrees of complements of maximal anti-chains on $\omega^{<\omega}$.

2. The e-degrees of languages of minimal subshifts.

3. The e-degrees of uniformly e-pointed trees on 2^ω.

4. The e-degrees of e-pointed trees on 2^ω.

An *e-pointed tree* T is a subtree of the infinite binary tree each path through which is $\leq_e$-above T. In computable model theory, given a countable structure $\mathfrak{M}$, the *degree spectra* of $\mathfrak{M}$ is the collection of Turing degrees of isomorphic copies of $\mathfrak{M}$. We note the following generalization of Knight et. al.'s *No two cones* theorem [45]:

Theorem 1.1 (Montalbán [45]). The degree spectra of a structure is never the Turing upward closure of an F_σ set of sequences, unless it is the enumeration cone of an e-pointed tree.

Little was known, however, about these cones. Even, for example, whether or not they were simply *all* the enumeration cones. In Section 2.4, we show they are only (and precisely) the cototal ones:

Corollary 2.16. The Turing cone of an F_σ set of sequences is the degree spectra of a structure if and only if it is the enumeration cone of a cototal set.

1.2 Randomness

In probability theory, random objects are nondeterministic, viewed as measurable maps from an underlying probability space. This presents a challenge to computability theory, where computations take place between fixed sequences. There are two perspectives we will consider to bring computability theory to bear on "random" sequences: one approach is to develop a theory of effectively measurable reductions, the framework for this perspective is provided by the notion of *layerwise reducibility*. A second approach is to develop a theory of *algorithmic randomness* which can be applied to individual binary sequences.

Both of these perspectives can be pursued by means of statistical tests determining null sets in 2^ω. We restrict ourselves to countable collections of tests which are presented in some effective way. We recommend either of the reference texts [17,49] for background.

The topology on the space of sequences 2^ω is given by the clopen sets $[\sigma] = \{X \in 2^\omega : \sigma \text{ is a prefix of } X\}$. Given $X \subseteq 2^{<\omega}$ a set of finite binary strings, we can define the open class $[X] = \bigcup_{\sigma \in X} [\sigma]$. A class $X \subseteq 2^\omega$ is *effectively open* (Σ^0_1) if $X = [W_e]$ for some computably enumerable (c.e.) set of strings $W_e \subseteq 2^{<\omega}$. A class $P \subseteq 2^\omega$ is Π^0_1 (*effectively closed*) if its complement is Σ^0_1.

A *Martin-Löf test* is a uniform sequence $\{A_n\}_{n<\omega}$ of effectively open classes with $\mu(A_n) \leq 2^{-n}$ for all n. A sequence $X \in 2^\omega$ is said to *pass the Martin-Löf condition* on a test $\{A_n\}_{n\in\omega}$ if $X \notin \bigcap_{n\in\omega} A_n$. The class MLR of *Martin-Löf random sequences* is the class of sequences passing the Martin-Löf condition on every Martin-Löf test.

We lose no generality in assuming that our Martin-Löf tests are *nested*, so that $A_{n+1} \subseteq A_n$ for each n. Then any nested Martin-Löf test $\{A_n\}_{n\in\omega}$ induces a layering $\{2^\omega \setminus A_n\}_{n\in\omega}$ of the measure-one class of sequences passing it.

1.3 Lowness and randomness

We think of sequences which are useful as oracles as being atypical. For example, the Turing cone $\{X : X \geq_T A\}$ above any noncomputable set A has measure zero [11, 58], so we would expect a "random" sequence to not be able to compute a given noncomputable set A. This intuition fails spectacularly for Martin-Löf randoms: given any set A, there exists an MLR sequence X such that $X \geq_T A$ [23,37]. There are several ways to strengthen the notion of Martin-Löf randomness which eliminate some of the

potential for computational utility. A striking dichotomy appears at the degree of the halting problem, $\emptyset'$: an MLR is either complete—strong enough to compute $\emptyset'$—or is computationally too weak even to compute a complete, consistent extension of Peano Arithmetic [64]. The *difference randoms* were introduced by Franklin and Ng as a test notion giving precisely the incomplete Martin-Löf randoms [22].

Definition 1.2. A *difference test* is a sequence $\{U_n \setminus V_n\}_{n\in\omega}$ where $\{U_n\}$ and $\{V_n\}$ are uniformly Σ^0_1 classes of subsets of 2^ω, and each component has measure $\mu(U_n \setminus V_n) \leq 2^{-n}$.

A sequence is *difference random* if it passes the Martin-Löf condition on all difference tests. An even stronger randomness notions is *Demuth randomness*, which ensures even less computational power: every Demuth random is GL_1 [49], i.e., not only is X incomplete, but its jump has lowest possible degree:

Definition 1.3. A sequence $X \in 2^\omega$ is *generalized low (GL_1)* if $X' \equiv_T X \oplus \emptyset'$.

Demuth randomness uses the notion of a *Demuth test*, but also applies a stronger criteria for passing a test. A sequence $X \in 2^\omega$ is said to *pass the Solovay condition* on a test $\{A_n\}_{n\in\omega}$ if $X \in A_n$ for only finitely many n. Recently, Bienvenu and Porter introduced the class of *strong difference randoms* [3]:

Definition 1.4. A sequence $X \in 2^\omega$ is *strong difference random* if it passes the Solovay condition on every difference test.

The strong difference random sequences fit in between the Demuth randoms and the difference randoms, but still exhibit many of the computational properties we would desire of random sequences, including that they are all GL_1 . In Chapter 3, we introduce a lowness notion and show that it characterizes the strong difference random sequences among all ML-randoms:

Definition 3.16. A sequence X is 2^n*-c.e. jump-dominated* if every X-partial computatable function is bounded by an ω*-c.e. function* with a computable approximation $f(n,s)$ with mind-changes bounded by 2^n.

A function $f(n)$ is Δ^0_2 if it is given as $\lim_{s\to\infty} f(n,s)$ of a computable function $f(n,s)$. In this case we call $f(n,s)$ a *computable approximation* to $f(n)$. We say that $f(n)$ is ω-*c.e.* if it has a computable approximation $f(n,s)$ such that the number of mind-changes $|\{s : f(n,s) \neq f(n,s+1)\}|$ is bounded by a computable function $g(n)$.

Theorem 3.18. A Martin-Löf random sequence X is strong difference random if and only if it is 2^n-c.e. jump-dominated.

We also give a characterization of the strong difference randomness in terms of Franklin and Ng's *strict Demuth tests* [22]:

Theorem 3.14. The strong difference randoms are the sequences passing the Solovay condition on every strict Demuth test.

1.4 Eliminating randomness

Randomness can be a powerful hypothesis for taming behavior and coaxing out results. We frequently draw from a deep body of techniques in algorithmic randomness to guide intuition and as a source of examples and counter-examples. However, it is also interesting to ask which results using randomness are fundamentally about randomness, and which can be achieved with weaker, purely combinatorial concepts.

One lowness class historically connected to randomness is the class of *strongly jump traceables*. A sequence $X \in 2^\omega$ is *strongly jump traceable* if given any order function h, every X-partial computable function has a c.e. trace bounded by h (for details, see

[26]). In Chapter 4, we consider the following theorem characterizing the strong jump-traceables:

Theorem 1.5 (Greenberg et. al. [25], Diamondstone et. al. [14]). The class of strongly-jump traceable sequences is:

1. The class of sequences computable in every superlow Martin-Löf random.
2. The class of sequences computable in every ω-c.e. Martin-Löf random.
3. The class of sequences computable in every superhigh Martin-Löf random.

For the two lowness notions, we are able to replace MLR with the weaker combinatorial condition of DNC. A function $f : \omega \to \omega$ is *diagonally non-computable (DNC)* if it differs from the universal Turing machine $\varphi_e(e)$ in every position: $f(e) \neq \varphi_e(e)$ for all $e \in \omega$.

We also show that in the superhigh case, DNC is not enough:

Theorem 4.8. The class of strongly-jump traceable sequences is:

1. The class of sequences computable in every superlow DNC function.
2. The class of sequences computable in every ω-c.e. DNC function.

Theorem 4.10. The class of sequences computable in every superhigh DNC function is the class of computable sequences.

1.5 Slopes of computable real-valued functions

Computable analysis studies continuous functions $f \in \mathcal{C}[0,1]$ via their representions by convergent sequences of polynomial approximations. We code sequences of polynomials

by infinite sequence in ω^ω. Restricting to those functions given by *computable* sequences with computable rates of uniform convergence gives the notion of a *computable real-valued function* introduced by Lacombe and Grzegorczyk [66].

It is known that different randomenss notions capture the common points of differentiability of different classes of computable functions [5,13]. In Chapter 5, we initiate the study of the values of these derivatives. Given a real number $x \in [0,1]$, we can identify almost-everywhere (in fact, off of the dyadic rationals) the real x with its unique binary expansion. Then x' denotes the halting problem relative to the binary expansion of x.

A map from 2^ω is said to be *layerwise recursive* if it is uniformly computable on $2^\omega \setminus A_n$ for some nested Martin-Löf test $\{A_n\}_{n\in\omega}$. Particularly, although there may be no single algorithm computing the function on all of 2^ω, it is computable by algorithms working on subsets of 2^ω of arbitrarily full measure.

We obtain the following results in Chapter 5:

Theorem 1.6.

1. There exists a computable $\mathcal{C}^1$ function $f \in \mathcal{C}^1[0,1]$ for which $f'(x) \geq_T x'$ almost everywhere.

2. There exists a computable function $f \in \mathcal{C}[0,1]$ which is differentiable almost everywhere, and for which $f'(x) \geq_T x'$ uniformly almost everywhere.

3. However, for any function $f \in \mathcal{C}[0,1]$ which is everywhere differentiable, it is not possible even for $f'(x) \geq_T \emptyset'$ uniformly almost everywhere.

4. There exists a computable function $f \in \mathcal{C}[0,1]$ which is everywhere differentiable, for which $f'(x) \geq_T x'$ layerwise uniformly.

5. There exists a computable $\mathcal{C}^1$ function $f \in \mathcal{C}^1[0,1]$ for which $f'(x) \geq_T \emptyset'$ layerwise

uniformly.

6. However, for any function $f \in \mathcal{C}^1[0,1]$, it is not possible for $f'(x) \geq_T x'$ layerwise uniformly.

Chapter 2

Applications of the cototal enumeration degrees

2.1 Introduction

Questions concerning the effective or algorithmic properties of classical mathematical objects have had a significant impact on the development of modern mathematics. In the study of Diophantine sets, we have Matiyasevich's resolution of Hilbert's tenth problem; or on the topic of word problems for groups, the seminal work of Dehn in the early development of geometric group theory. A central goal of applied computability theory is to mathematically formalize this algorithmic perspective by precisely quantifying the computational content of classes of mathematical objects and relations.

The most widely studied measure of such computational content is that given by Turing reducibility, and one approach to quantifying the complexity of a class of objects is to calculate the *degree spectra* of the class, that is, the collection of Turing degrees obtained by the members of the class. Consider for example the characterization of the degrees of the block relations of computable linear orderings as precisely the Σ^0_3 degrees [19], or the Novikov-Boone characterization of the degrees of word problems for finitely presented groups as the computably enumerable degrees [4].

However, applications often must employ reducibilities other than Turing reduction. An early indication that one must consider other reducibilities comes from group theory. A result of Macintyre shows that for finitely generated groups G and H, that if G embeds as a subgroup of every algebraically closed extension of H, then the word problem for G is Turing reducible to the word problem for H [39]. Ziegler showed that the converse fails, but that the result becomes an equivalence if a stronger computability-theoretic reducibility, namely Ziegler reducibility, is substituted for Turing reducibility [69]. Because an algebraically closed group is determined by the finitely generated groups that embed in it, the computability theoretic structure not only provides a complete understanding of the algebraic one, but is in fact implicit within the algebraic structure (see also [2, 28]).

Ziegler's reducibility (which he called $*$-reducibility) was formulated as a strengthening of *enumeration reducibility*, a reducibility important in its own right and the primary reducibility we will use in this chapter. Enumeration reducibility, along with its associated degree structure, measures the relative computational difficulty in producing enumerations of sets of natural numbers.

Enumeration reducibility also revealed its significance early on in the study of *structure spectra*, that is, degree spectra of isomorphism classes of structures. Richter used enumeration reducibility to give sufficient conditions on a first-order theory to ensure it has countable models whose structure spectra has no least element, so that the Turing degrees alone are not sufficient to capture the effective content of the structure considered up to isomorphism [54, 55].

Enumeration reducibility has also been important in applications to computable analysis. Miller, answering a question of Pour-El and Lempp, showed that the Turing degrees

are similarly deficient for quantifying the complexity of continuous real-valued functions, introducing the *continuous degrees*, a subclass of the enumeration degrees which are able to capture the computational content of continuous functions [40]. Kihara and Pauly have extended this connection to associate degree structures to arbitrary quasi-Polish spaces, obtaining the enumeration degrees as the degree structure associated to the universal quasi-Polish space $\mathcal{O}(\mathbb{N})$ [35]. These connections have proven particularly fruitful, resulting in a solution to the general n'th level Borel isomorphism problem.

In this chapter, using a specific subclass of the enumeration degrees: the *cototal* enumeration degrees, we succeed in identifying several degree spectra which have been of interest in the fields of effective structure theory and symbolic dynamics. In Section 2.4, we see that the cototal enumeration degrees can be used to characterize which enumeration cones are obtainable as Turing upward closures of F_σ sets of reals, a question stemming from a result of Montalbán on structure spectra. Then in Section 2.5, we consider an example from symbolic dynamics, that of *subshifts*, and show that the cototal enumeration degrees provide a complete characterization of the Turing degree spectra of their building-blocks: the *minimal* subshifts.

The computational power of subshifts, and in particular of minimal subshifts, has generated interest for several years. From the perspective of computability theory, see Durand, Levin, and Shen [18], Cenzer, Dashti, and King [6], Cenzer, Dashti, Toska, and Wyman [7,8], as well as Simpson [61], Jeandel and Vanier [33], Hochman and Vanier [29], and Jeandel [32]. In computer science, the study of the computational power of simple dynamical systems, especially subshifts, comprises an active body of recent research (see [12] for a summary). Minimal subshifts are also important in studying individual infinite sequences, because measures of sequence complexity that are important in

characterizing the algebraic combinatorics of infinite sequences can be studied on the subshifts they generate (the well-studied Thue-Morse sequence, for example, generates a minimal subshift) [53].

We consider the results of Section 2.5 to constitute an extension of the connection between the enumeration degrees and applications of effective mathematics—already rich in application to group theory and analysis—now to the field of symbolic dynamics. Although these results mark the first use of the cototal enumeration degrees to identify a degree spectrum of independent interest, observations of Jeandel indicate intimate connections between cototality and both simple groups and maximal ideals of rings [32], so we anticipate further interest. Also, in the point-degree spectrum language of Kihara and Pauly, cototality results from topological tameness of the underlying represented spaces, so we believe a better understanding of the cototal degrees will be impactful with regard to those applications as well.

In addition to our applications, some of our results should be useful in pursuing a greater understanding of the cototal enumeration degrees themselves. For example, Theorem 2.3 provides a simple characterization of the cototal enumeration degrees as those e-degrees which contain complements of maximal anti-chains on $\omega^{<\omega}$, and Theorem 2.9 as the enumeration degrees which contain e-pointed trees. This second characterization is particularly useful because of the rich intro-enumerability properties of e-pointed trees. For example, Miller and Soskova have recently used Theorem 2.9 to prove that the cototal enumeration degrees are dense [42]. In the theory of the structure of the enumeration degrees, cototality corresponds to a combinatorial property of *good approximation* that has been essential in establishing structural properties of the enumeration degrees, so we anticipate that further study of the cototal degrees will produce greater

insight into the structure of the enumeration degrees.

2.2 Cototal sets and degrees

Enumeration reducibility, introduced by Friedberg and Rogers in 1959, captures the relative difficulty of producing enumerations of sets of natural numbers. Alternatively, it can be thought of as a notion of computation between sets that uses only the positive portion of their set membership information.

An enumeration functional Γ is a computably enumerable (c.e.) set of pairs $\langle n, F\rangle$ with each $n \in \omega$ and F the canonical code of a finite subset of ω. We think of Γ as reading the positive membership information of X, and, for $\langle n, F\rangle \in \Gamma$, enumerating n upon seeing $F \subseteq X$. Given $X \subseteq \omega$, we define $\Gamma(X) = \{n : \exists F(\langle n, F\rangle \in \Gamma \text{ and } F \subseteq X)\}$.

For sets $A, B \subseteq \omega$, we say that $A \leq_e B$ if there exists an enumeration functional Γ with $\Gamma(B) = A$. Equivalently, $A \leq_e B$ if there is a single Turing functional which, given any enumeration of B, outputs an enumeration of A. The relation $\leq_e$ defines a pre-order on 2^ω, the partial order it induces is called the *enumeration degrees*, or e-degrees, denoted $\mathcal{D}_e$.

Another way to characterize enumeration reducibility was given by Selman [59]. Given a set X, let $\mathcal{E}(X)$ denote the collection of all Turing degrees computing enumerations of X, called the *enumeration cone of* X. Then Selman showed:

Theorem 2.1 (Selman [59]). $A \leq_e B$ if and only if $\mathcal{E}(B) \subseteq \mathcal{E}(A)$.

A set X is *total* if its positive information already suffices to determine its negative information, or precisely: if $\overline{X} \leq_e X$. We call an enumeration degree *total* if it is the e-degree of a total set. The Turing degrees (which we denote by $\mathcal{D}_T$) embed in the

enumeration degrees via the map induced set-wise by $X \mapsto X \oplus \overline{X}$. The image of this embedding is the total e-degrees.

The name "total" is evocative of the following fact: given a total function f, total in the sense of having full domain, the set $\text{graph}(f) = \{\langle n, f(n)\rangle : n \in \omega\}$ is total under enumeration reducibility. In fact, every total set is enumeration-equivalent to the graph of a total function, for example, its characteristic function.

A closely related notion is that of cototality. A set A is *cototal* if $A \leq_e \overline{A}$ that is, if the complement of A is total as a set, and we call an enumeration degree *cototal* if it is the e-degree of a cototal set.

For example, the complements of graphs of total functions are cototal as sets. The e-degrees of such sets are called *graph cototal*. The class of graph cototal degrees has been studied by Solon in [67] and [68]. Several other natural classes of cototal sets were brought to attention by Jeandel in [32], including examples from symbolic dynamics and algebra.

The cototal degrees were studied recently by Andrews et al. in [1]. In addition to showing that the cototal enumeration degrees are a proper subclass of the enumeration degrees (that not every enumeration degree is cototal), they also separate the class of cototal degrees from the class of graph cototal degrees. That is: not every cototal set is enumeration equivalent to a complement of the graph of a total function. It is natural then to look for classes of objects that *do* capture cototality in the enumeration degrees. Andrews et al. show that the complements of maximal independent sets in $\omega^{<\omega}$ (with $\omega^{<\omega}$ considered as an undirected graph) form one such class.

In Section 2.3 we identify another simple class of objects characterizing cototality, showing that the cototal degrees are the degrees of complements of maximal anti-chains

on $\omega^{<\omega}$. In Section 2.4, we show that the e-degrees of enumeration pointed trees are the same as those of the maximal anti-chain complements, providing yet another class of objects whose e-degrees are the cototal degrees. Although we use this section as a stepping stone to approach the class considered in Section 2.5, e-pointed trees are interesting in their own right and provide us with applications to computable structure theory.

Section 2.5 focuses on a particular example of a class of cototal objects identified by Jeandel in [32], namely, the languages of minimal subshifts. Jeandel and Vanier in [33] prove that the Turing degree spectra of a nontrivial minimal subshift is the enumeration cone of its language. We show that the enumeration degrees of languages of minimal subshifts are the same as the enumeration degrees of enumeration pointed trees, providing a characterization of the Turing degree spectra of nontrivial minimal subshifts as precisely the enumeration cones of cototal sets.

2.3 Maximal anti-chain complements

The graph $\omega^{<\omega}$ is the graph of finite strings of integers, ordered by extension.

Theorem 2.2. If A is a maximal antichain on $\omega^{<\omega}$, then $\overline{A}$ is cototal.

Proof. To determine if a string $\sigma \in \omega^{<\omega}$ is in $\overline{A}$, we wait for some element comparable but not equal to σ to enter A. Since A is an antichain, we only enumerate elements of $\overline{A}$ in this way. And by maximality, if $\sigma \in \overline{A}$ then something comparable but not equal to σ must be in A, so our procedure enumerates all of $\overline{A}$. □

Note that every total set is enumeration-equivalent to a maximal antichain. Given a

total set A, consider C given by $\{n : n \in A\} \cup \{n^\frown k \ldots : n \in \overline{A}, k \in \omega\}$. Then $A \equiv_e C$. However, it may be that $\overline{A} \not\equiv_e \overline{C}$, i.e., when $\overline{A}$ is not total itself. Nonetheless, the degrees of cototal sets are, in fact, exactly the degrees of the complements of maximal antichains:

Theorem 2.3. If A is cototal, then $A \equiv_e \overline{C}$ for some C a maximal antichain on $\omega^{<\omega}$.

Contrast this result with the case for function graphs: every total set A is enumeration-equivalent to the graph of a total function, for example the graph of its characteristic function χ_A, but not every cototal degree contains a set of the form $\overline{\text{graph}(f)}$ [1].

Proof of Theorem 2.3. Let $A \leq_e \overline{A}$ via the enumeration operator Γ. Fix a computable listing of Γ to work with. We construct a subset $\overline{C}$ of $\omega^{<\omega}$ as follows:

First, put $\lambda \in \overline{C}$. For the first layer of $\overline{C}$ as a subset of $\omega^{<\omega}$, we enumerate A. That is, $n \in \overline{C} \iff n \in A$.

For each node $\alpha \in \omega^{<\omega} \setminus \{\lambda\}$, we attach some finite set, which we call the *claim* of α. We think of α as claiming this finite set is a subset of $\overline{A}$. A node is filled in, that is, enumerated into $\overline{C}$, when we witness its claim to be false. For the first-level nodes $\alpha = n$, we set their claims to $\{n\}$. So for nodes $\alpha = n$, $\alpha \in \overline{C} \iff \alpha \in A$, that is, if and only if α is wrong about its claim.

Layer by layer, we attach a claim to each node α as follows: each node α of length $|\alpha| = k$ looks at all the $k-1$ nodes below it (apart from λ), and their claims, and chooses one element from each claim to attempt to prove wrong. For each chosen element n, α picks axioms $\langle n, F\rangle$ from Γ. We choose claims in such a way that directly above each node, we attach all possible claims from all possible choices of axioms from Γ that could prove the claims below them wrong.

To do this asignment computably we assume, without loss, that Γ has the following property: for all $n \in \omega$, $\exists F$ such that $\langle n, F\rangle \in \Gamma$. To achieve this we can, for example, add the axioms $\langle n, \{a\}\rangle$ to Γ for some fixed $a \in A$.

We define the claim of α to be the union of the F's from those axioms chosen. A node α is put into $\overline{C}$ when our enumeration of A proves its claim wrong. That is, when some element in the claim of α is enumerated into A.

So $\overline{C}$ is enumeration below A via the construction, and also A is enumeration below $\overline{C}$ because A can be read out explicitely in the first layer of $\overline{C}$.

Claim 2.4. C is a maximal antichain.

First, C is an anti-chain: we must show it is not possible that $\alpha, \beta \in C$ with $\alpha \prec \beta$. Indeed, if $\beta \in C$ with $\alpha \prec \beta$, we have that $\text{claim}(\beta) \subseteq \overline{A}$, but by construction there exists $F \subseteq \text{claim}(\beta)$ so that $\langle n, F\rangle \in \Gamma$ and $n \in \text{claim}(\alpha)$. But then $n \in A$, so that $\text{claim}(\alpha) \not\subseteq \overline{A}$, so we have $\alpha \notin C$. So indeed, C is an antichain.

Now suppose C were not maximal as an antichain. That is, suppose there exists some $\alpha \notin C$ so that $\{\alpha\} \cup C$ were an anti-chain. But under our construction, since $\alpha \notin C$ we have that $\text{claim}(\alpha) \not\subseteq \overline{A}$. Say $n \in \text{claim}(\alpha)$, $n \notin \overline{A}$. And by assumption on Γ, there is an $F \subseteq \overline{A}$ so that $\langle n, F\rangle \in \Gamma$. But also since $\{\alpha\} \cup C$ is an anti-chain, all ancestors of α are not in C, so similarly there are n_i for each ancestor α_i, with $n_i \notin \overline{A}$ and finite sets $F_i \subseteq \overline{A}$ with $\langle n_i, F_i\rangle \in \Gamma$. But then by construction, there is some immediate descendent β of α so that $\text{claim}(\beta)$ is the union of these F_i, so that $\text{claim}(\beta) \subseteq \overline{A}$, so that $\beta \in C$, so that $\{\alpha\} \cup C$ is not an antichain, a contradiction. $\square$

2.4 Enumeration pointed trees

Definition 2.5. An *e-pointed tree* is a tree $T \subseteq 2^{<\omega}$ with no dead-ends, such that $\{\sigma_i\}_{i\in\omega} \geq_e T$ whenever $\{\sigma_i\}_{i\in\omega} \subseteq 2^{<\omega}$ is a path through T.

Notice here that we view a path through a tree as a subset of $2^{<\omega}$, rather than as an element of 2^ω. In the Turing degrees, we are less careful to avoid mixing types: a set $X \in 2^\omega$ and of its collection of prefixes $\{X \upharpoonright i\}_{i\in\omega} \subseteq 2^{<\omega}$ have the same Turing degree. But in terms of enumeration degree we have $\deg_e(\{X \upharpoonright i\}_{i\in\omega}) = \deg_e(X \oplus \overline{X})$. In particular, viewed as subtrees, paths always have total enumeration degree.

Enumeration pointed trees were encountered in work of Antonio Montalbán in computable structure theory:

Theorem 2.6 (Montalbán [45]). The Turing upward closure of an F_σ set of reals in ω^ω cannot be the degree spectra of a structure unless it is an enumeration cone. In fact, for $X \subseteq \mathcal{D}_T$, the following are equivalent:

1. X is the degree spectra of a structure and the Turing upward closure of an F_σ set of reals in ω^ω.

2. X is the enumeration cone of an e-pointed tree.

Recall that the enumeration cone of a set $A \subseteq \omega$ is the collection $\mathcal{E}(A)$ of all Turing degrees computing enumerations of A. In the case that A has total enumeration degree, $\mathcal{E}(A)$ coincides with the Turing cone of A: the upward closure of $\mathbf{d}(A)$ in the Turing degrees. Montalbán notes that there are structure spectra as in Theorem 2.6 which are not Turing cones. However, it was not known precisely which enumeration cones were possible, for example, whether every enumeration cone is realized as the Turing

upward closure of an F_σ set of reals. We provide an answer by showing that the enumeration degrees of e-pointed trees are exactly the cototal degrees. In particular, not every enumeration cone is realized as a structure spectra as in Theorem 2.6.

We first show that e-pointed trees appear in every cototal degree. In fact, every cototal degree contains an e-pointed tree of a particular form.

Definition 2.7. A *uniformly e-pointed tree* is a tree $T \subseteq 2^{<\omega}$ with no dead-ends for which there exists an enumeration functional W so that $W(X) = T$ whenever $X \subseteq 2^{<\omega}$ is a path through T.

Uniformly e-pointed trees have a useful intro-enumerability property:

Theorem 2.8. If T is a uniformly e-pointed tree, then for each n, there exists an m so that $T \upharpoonright n \subseteq W_{|\sigma|}(\sigma)$ for all $\sigma \in T$ with $|\sigma| \geq m$.

When working with a uniformly e-pointed tree T and functional W, we denote the function taking n to the first such m by $s(n) = m$.

Proof of Theorem 2.8. Since every path through T enumerates $T \upharpoonright n$, the collection of basic clopen sets $\{[\sigma] : \sigma \in T \text{ and } W_{|\sigma|}(\sigma) \supseteq T \upharpoonright n\} \cup \{[\sigma] : \sigma \notin T\}$ covers 2^ω. By compactness of 2^ω, finitely many $[\sigma]$ suffice, so by some finite level every path in T has enumerated $T \upharpoonright n$. □

Theorem 2.9. If A is cototal, then $A \equiv_e T$ for some uniformly e-pointed tree T.

Proof. Fix C a maximal antichain on $\omega^{<\omega}$ with $\overline{C} \equiv_e A$. Each nonzero level of $2^{<\omega}$ will be associated to a pair of comparable, unequal strings in $\omega^{<\omega}$.

Put λ in T. If level n is associated with (σ, τ), then every node on level n of T branches left if $\sigma \in \overline{C}$ and right if $\tau \in \overline{C}$.

Since C is an antichain, at most one of σ, τ belong to C, so T has no dead ends. By construction, $\overline{C} \geq_e T$.

To see that $T \geq_e \overline{C}$, we describe an enumeration functional, which we will call V: we enumerate τ when we branch left at the level associated to (τ, σ) and enumerate σ when we branch right. But we have more: we claim that by the same V, each path in T enumerates $\overline{C}$.

Let $\sigma \in \overline{C}$. Then by maximality of C, there must be some element τ comparable to σ with $\tau \in C$. Then the level of $2^{<\omega}$ associated to (τ, σ) can only branch to the right in T, so whichever path we take in T, we must enumerate σ.

So our functional V gives $X \geq_e \overline{C}$ for each path X through T uniformly. Composing this with the reduction $\overline{C} \geq_e T$, we obtain a functional W witnessing that T is a uniformly e-pointed tree. □

One can see, by an application of compactness of 2^ω, that uniformly e-pointed trees are cototal. It is more difficult to see that e-pointed trees themselves are all of cototal degree. To prove this, we first pass through a superclass of uniformly e-pointed trees.

Definition 2.10. A *uniformly e-pointed tree with dead-ends* is a tree $T \subseteq 2^{<\omega}$, possibly with dead-ends, for which there exists an enumeration functional W so that $W(X) = T$ for all paths X through T.

In particular, a uniformly e-pointed tree is a uniformly e-pointed tree with dead-ends.

Theorem 2.11. If T is a uniformly e-pointed tree with dead ends, then T is cototal.

Proof. Let W be the enumeration functional witnessing that T has the e-pointed property uniformly.

Let $n \in T$ and consider the following cover of 2^ω by clopen sets:

$$\{[\sigma] : \sigma \notin T\} \cup \{[\sigma] : W(\sigma) \ni n\}$$

Since 2^ω is compact, finitely many σ suffice to cover the space. But then after enumerating finitely many $\sigma \in \overline{T}$, for any $n \in T$ we witness at some finite stage that every path remaining enumerates n, so we can safely enumerate n into T. This procedure is uniform, so $\overline{T} \geq_e T$. □

Lemma 2.12. If T is a (non-uniformly) e-pointed tree, then there exists some uniformly e-pointed tree T' with dead ends such that $\overline{T'} \geq_e T \geq_e T'$.

Proof. Given T, we attempt to build a sequence of subsets T_i diagonalizing against enumeration functinoals W_i as follows:

$T_0 = T$.

To define T_{n+1}, consider the enumeration functional W_n.

If there exists $\sigma \in T_n$ with $[\sigma] \cap [T_n] \neq \emptyset$ and $W_n(\sigma) \not\subseteq T$, then set $T_{n+1} = T_n \cap [\![\sigma]\!]$. Otherwise, assuming W_n does not enumerate T uniformly on $[T_n]$, there must be some path $X \in [T_n]$ so that $W_n(X) \not\ni \tau_n$ for some $\tau_n \in T$. Then define T_{n+1} by removing any node $\sigma \in T_n$ for which $W_{n,|\sigma|}(\sigma) \ni \tau_n$.

If this procedure continues indefinitely, we have nested sets T_i with $[T_i]$ all nonempty. A nested sequence of compact nonempty sets has nonempty intersection, so we obtain a path $X \in T$ on which no W_n enumerates T.

So this procedure must stop at some finite stage, that is, after intersecting T with some finitely many $[\![\sigma_k]\!]$ and removing all nodes σ such that $W_{n,|\sigma|}(\sigma) \ni \tau_k$ for finitely many k, τ_k, we have that W_n enumerates T uniformly on $[T_{n+1}]$.

Let $T' = T_{n+1}$, $W' = W_n$. Notice that since we intersect with finitely many $[\![\sigma_k]\!]$ and remove only nodes σ such that $W_{k,|\sigma|}(\sigma) \ni \tau_k$ for finitely many k, τ_k, we have that $T \geq_e T'$: as we enumerate T we allow only those nodes lying above or below the finitely many σ_k, and before enumerating a node σ check whether $W_{k,|\sigma|}(\sigma) \ni \tau_k$ for the finitely many k, τ_k.

Claim 2.13. $\overline{T'} \geq_e T$.

Again by a similar compactness argument as in Theorem 2.11, for $n \in T$ we have that

$$\{[\sigma] : \sigma \notin T'\} \cup \{[\sigma] : W'(\sigma) \ni n\}$$

is an open cover of 2^ω. By compactness, in enumerating $\overline{T'}$, by some finite stage we will have enumerated enough of $\overline{T'}$ to see, checking finitely many other σ, that all remaining paths enumerate n. □

Theorem 2.14. If T is an e-pointed tree, then T has cototal degree.

Proof. Let $X = T \oplus T'$, with T' as in Lemma 2.12. Clearly $X \geq_e T$, and since $T \geq_e T'$, we see $T \geq_e X$. So $X \equiv_e T$. And since $\overline{T'} \geq_e T \geq_e T'$, we see that $\overline{X} \geq_e X$, so X is cototal. □

As a corollary:

Corollary 2.15. The following are equivalent of an e-degree $\mathbf{e}$:

1. $\mathbf{e}$ contains a uniformly e-pointed tree.
2. $\mathbf{e}$ contains an e-pointed tree.
3. $\mathbf{e}$ contains a uniformly e-pointed tree with dead-ends.
4. $\mathbf{e}$ is cototal.

We obtain a corollary to Montalbán's Theorem 2.6:

Corollary 2.16. A degree spectrum is the Turing upward closure of an F_σ set of reals in ω^ω if and only if it is the enumeration cone of a cototal set. In fact, for $X \subseteq \mathcal{D}_T$ the following are equivalent:

1. X is the degree spectrum of a structure and the Turing upward closure of an F_σ set of reals.

2. X is the enumeration cone of a cototal set.

In particular, not every enumeration cone may be simultaneously obtained both as the degree spectrum of a structure and as the Turing upward closure of an F_σ set of reals.

2.5 Minimal subshifts

In [32], Emmanuel Jeandel gave several examples of classes of algebraic and combinatorial objects exhibiting cototality. We will consider one such class, the languages of minimal subshifts. More on minimal subshifts can be found in [29]. The *shift operator* on 2^ω is the map taking a real $\alpha \in 2^\omega$ to the unique $\beta \in 2^\omega$ such that $\alpha = n^\frown\beta$ for some $n \in 2$, that is, the operator which erases the first bit of a real. In functional notation, it is the operator $\alpha(n) \mapsto \alpha(n+1)$.

Definition 2.17. A *subshift* is a closed, shift-invariant subspace of 2^ω.

The trivial example of 2^ω itself is called the *full binary shift*. Binary subshifts are thought of as describing the evolution of a symbolic dynamical system taking states in $\{0, 1\}$. More generally, subshifts on n^ω can be defined for any finite set of n states,

with elements of the subshift describing a possible sequence of states taken over some evolution of the system.

Definition 2.18. A subshift X is *minimal* if it satisfies one of the following equivalent conditions:

1. X contains no proper subshifts.
2. X is the shift-invariant closure of any of its points.
3. Every point of X contains the same subwords.

Definition 2.19. The *language* of a subshift X, denoted $\mathcal{L}(X)$, is the collection of all subwords appearing in any of its points. The set $\overline{\mathcal{L}(X)}$ is called the *set of forbidden words*.

The closure condition guarantees that a subshift is characterized by its language, or equivalently by its set of forbidden words. Conversely, designating any collection of words as *forbidden* determines a unique subshift consisting of all infinite strings which avoid the designated forbidden words (i.e., which do not contain any forbidden word as a subword).

Definition 2.20. The Turing degree *spectrum* of a subshift X is the collection of Turing degrees of its points.

Definition 2.21. A subshift is *trivial* (or *periodic*) if it is the shift-invariant closure of a point of the form $X = w^\infty$ for some finite word $w \in 2^{<\omega}$.

The language of a subshift is particularly relevant to us because of the following theorem:

Theorem 2.22 (Jeandel, Vanier [33]). If X is a minimal subshift which is not trivial, the Turing degree spectrum of X is the enumeration cone $\mathcal{E}(\mathcal{L}(X))$.

Note by Theorem 2.1, the set $\mathcal{E}(\mathcal{L}(X))$ of Turing degrees which compute enumerations of $\mathcal{L}(X)$ is characterized by the enumeration degree of $\mathcal{L}(X)$. So understanding the possible Turing degree spectra of minimal subshifts X reduces to understanding what enumeration degrees lie at the base of these enumeration cones $\mathcal{E}(\mathcal{L}(X))$. The cototal degrees enter the picture here:

Theorem 2.23 (Jeandel [32]). If X is a minimal subshift, then $\mathcal{L}(X)$ is cototal.

Degreewise then, an enumeration degree must be cototal to be the enumeration degree of the language of a minimal subshift. We show that this condition is sufficient. That is: each cototal degree contains the language of a minimal subshift.

Our construction is similar to the main construction in [29], in that we build a minimal subshift X as a nested intersection of subshifts X_n generated by languages L_n, with each L_{n+1} built up from concatenations of words in L_n. In [29], minimality is ensured by requiring that each word in L_{n+1} contains all of L_n as subwords. Our main insight is that it is enough that for each n there exists an $m > n$ so each word in L_m contains all of L_n as subwords. This relaxed condition allows us to exploit the intro-enumerability property of e-pointed trees given in Theorem 2.8.

Theorem 2.24. If A is cototal, then $A \equiv_e \mathcal{L}(X)$ for some X a minimal subshift on 2^ω.

Proof. Given A cototal, let $T \in \deg_e(A)$ be a uniformly e-pointed tree with functional W. Fix an enumeration $\{a_k : k \in \omega\}$ of W and put $W_s = \{a_k : k < s\}$.

For each string $\sigma \in 2^{<\omega}$, the set $W_{|\sigma|}(\sigma)$ defines a subtree of $2^{<|\sigma|}$ given by the downward closure of $2^{<|\sigma|} \cap W_{|\sigma|}(\sigma)$, which we will denote by W^σ. Note that W^σ are increasing in σ, i.e., $\sigma \succ \tau \Rightarrow W^\sigma \supseteq W^\tau$, and that without loss of generality the W^σ have the property that $\tau \in W^\sigma \Rightarrow W^\tau \subseteq W^\sigma$. We define levels L_i^σ inductively in i and

σ.

For each σ, define $L_0^\sigma = \{0, 1\}$. Then L_{n+1}^σ consists of words of the form

$$AAAB(AB)^k AA(CDE...Z)B$$

where $\{A, B, ..., Z\} = \bigcup_{\tau \in W^\sigma} L_n^\tau$ with $A, B, \ldots Z$ distinct, and k, thought of as an element of $2^{<\omega}$, is both in W^σ and has length $n + 1$.

We let

$$L_n = \bigcup_{\sigma \in T} L_n^\sigma$$

and define X_n to be the subshift generated by concatenations of the words in L_n. Let $X = \bigcap_{n<\omega} X_n$. This ends the construction. We now verify this X satisfies our conditions:

Claim 2.25. X is a subshift.

Each word in L_{n+1} is made up of concatenations of words in L_n, so we have that $X_{n+1} \subseteq X_n$. Hence X, being a nested intersection of closed, shift invariant subsets of 2^ω, is itself closed and shift invariant.

Notice that since $W^\sigma \subseteq 2^{<|\sigma|}$, and words are put in L_m^σ only to code for nodes in W^σ of length m, we have $L_m^\sigma = \emptyset$ for $m \geq |\sigma|$. Hence we can also write:

$$L_n = \bigcup_{\sigma \in T, |\sigma| > n} L_n^\sigma$$

Claim 2.26. $T \geq_e \mathcal{L}(X)$.

By the construction, we have that $T \geq_e \bigcup L_n$. Then T enumerates the collection of all subwords of $\bigcup L_n$, which we denote by L. We claim that $L = \mathcal{L}(X)$.

First, $\mathcal{L}(X) \subseteq L$: suppose w is a subword of some point in X. Pick n large enough so that the length of w is less than any word in L_n. Then since w appears in X_n it appears

in some concatenation of words in L_n, and by choice of n, in fact w must then appear in a concatenation of at most two words in L_n, say AB with $A \in L_n^{\sigma_1}$ and $B \in L_n^{\sigma_2}$ with $\sigma_1, \sigma_2 \in T$. Let $m > n, |\sigma_1|, |\sigma_2|$ and consider the level $s(m)$, where $s(m)$ is the function from Theorem 2.8. Then if $|\sigma| > s(m)$, we know that $W^\sigma \supseteq T \upharpoonright m$, so that $\bigcup_{\tau \in W^\sigma} L_n^\tau \supseteq L_n^{\sigma_1}, L_n^{\sigma_2}$. In particular, $\bigcup_{\tau \in W^\sigma} L_n^\tau$ contains A and B, so w appears in the word $AAAB(AB)^kAA(CDE...Z)B$ in L_{n+1}^σ.

Secondly, $L \subseteq \mathcal{L}(X)$: given a subword w of some word in L_n^σ for $|\sigma| \geq n$, let $m = s(|\sigma|)$. Then every word in L_m contains w since $\sigma \in W^\tau$ for every $\tau \in T$ with $|\tau| \geq m$. Hence w appears in every point of X_m, so certainly in X.

Claim 2.27. $\mathcal{L}(X) \geq_e T$:

To see that $\mathcal{L}(X) \geq_e T$, recall that $\mathcal{L}(X) = L$. So to enumerate T we re-run the inductive construction of the languages L_n. We can output $L_0 = \{0, 1\}$, then at each stage we look among all the W^σ for $\sigma \in 2^{<\omega}$, and we search our enumeration of L for subwords of the form

$$AAAB(AB)^kCDE...ZB$$

where $k \in W^\sigma$ and if k has length $n+1$ then $\{A, B, C, \ldots, Z\} = \bigcup_{\tau \in W^\sigma} L_n^\tau$. If such a word is found in our enumeration of L, then we output the node coded by k into our enumeration of T.

Before we can see that this procedure enumerates all of T, first note that it does reconstruct the layers L_n^σ at least for $\sigma \in T$. We can proceed by induction: any w in L is a subword of some word in L_n^σ:

For $n = 0$, $L_0^\sigma = \{0, 1\}$ are both outputted.

If w appeared in L_{n+1}^σ, then it was constructed from letters in L_n^τ for $\tau \in W^\sigma$, so by

induction hypothesis on n and τ, we enumerate every letter in L^{σ}_{n+1} for at least every $\sigma \in T$.

Since all of T is in fact coded at some level L^{σ}_{n}, this means the procedure certainly enumerate at least all of T. What we need to see is that in fact only elements of T are enumerated. The worry is that that by searching among all possible W^{σ} for all $\sigma \in 2^{<\omega}$, rather than just $\sigma \in T$, we may have accidently enumerated nodes corresponding to k's found that were not actually ever in T.

To verify that we only enumerate σ corresponding to k for $\sigma \in T$, we again proceed by induction. Suppose we have found k in some word w in L. Now any w of the appropriate form that we do find must be found as some concatenation of words in some L_j (for some $L^{\sigma_i}_{j}$, but we can forget which σ_i). Let j be smallest such: i.e., w does not appear strictly within any of the words $A, B, C, \ldots, Z$ in L_j. Then to have seen w and accepted it as in L_j, we must have seen a subword made up of these letters and of the form $AAAB(AB)^kAA(CDE...Z)B$, but that was not put into L to code for $k \in T$. But then in particular, we found w while searching for subwords of concatenations of L_j that begin with a word in L_j repeated three times. But there is no way in L_j to concatenate words of the correct form to obtain any new word of this form: the only way to even obtain a new subword of the form AAA is to concatenate a word ending in A, say $PPPA(PA)^k(QR..Z)A$ with a word starting in A, say $AAAV(AV)^j(WX..Z)V$ but then the only new sequence of three As is followed by another A, not some distinct letter B, so it is not of the correct form either.

Claim 2.28. X is minimal.

Suppose p appears in some point on X. Then p appears in some word $w \in L_n$ so in

some $w \in L_n^\sigma$ for some $\sigma \in T$.

Let $m = s(|\sigma|)$. Then for $|\tau| \geq m$, since $W^\tau \supseteq T \upharpoonright |\sigma| \ni \sigma$, we know w appears in L_{n+1}^σ. So taking $k \geq \max(m, n+1)$ we see that w appears in every word in L_k. So all the points in X, being points in X_k, contain p.

Since p was arbitrary, all points of X contain the same subwords. □

Theorem 2.22 makes our result of particular interest, as it allows us to find examples of degree spectra of minimal subshifts using known examples of cototal sets. We close with one such application:

Definition 2.29. The *Turing co-spectrum* of a minimal subshift X is the collection of all lower-bounds of the degree spectra of X, i.e. $\{\mathbf{d} \in \mathcal{D}_T : \mathbf{d} \leq_T \mathbf{b}, \mathbf{b} \in \mathrm{Spec}_T(X)\}$.

Gutteridge shows that there is a quasiminimal cototal degree $\mathbf{q}$ [27]. That is, a cototal degree $\mathbf{q}$ which is nonzero, and bounds no non-zero total e-degree. Taking X a minimal subshift with $\deg_e(\mathcal{L}(X)) = \mathbf{q}$, we obtain the following:

Theorem 2.30. There exists a minimal subshift with no computable points, but whose Turing co-spectrum is $\{\mathbf{0}\}$.

Chapter 3

The Strong Difference Randoms

3.1 Introduction

The study of algorithmic randomness concerns notions of randomness which can be applied to individual subsets of ω. For an overview of the basic ideas and results in algorithmic randomness, we recommend [17, 49]. The most widely studied notions of algorithmic randomness are given in terms of passing a countable collection of statistical tests, each of which describe null subsets of 2^ω.

Definition 3.1. A *test* is a sequence $\{A_n\}_{n\in\omega}$ of subsets of 2^ω with $\mu(A_n) \to 0$. A real $X \in 2^\omega$ is said to *pass the Martin-Löf condition* on a test $\{A_n\}_{n\in\omega}$ if $X \notin \bigcap_{n\in\omega} A_n$, and to *pass the Solovay condition* on a test $\{A_n\}_{n\in\omega}$ if $X \in A_n$ for only finitely many n.

Test-based algorithmic randomness notions are typically obtained by restricting to a countable class of tests, and considering either the class of reals which pass the Martin-Löf condition on all those tests, or the class of reals which pass the Solovay condition on all those tests. Note that the Solovay condition is a stronger condition: a real which passes the Solovay condition on a test already passes the Martin-Löf condition on that test. When applied to the same class of tests, imposing the Solovay condition gives a randomness notion at least as strong as imposing the Martin-Löf condition, and often a strictly stronger randomness notion.

Solovay introduced the Solovay condition in conjunction with the notion of a *Solovay test*:

Definition 3.2. A *Solovay test* is a sequence $\{A_n\}_{n<\omega}$ of uniformly Σ^0_1 classes in 2^ω, the sum of whose measures is finite: $\sum_{n<\omega} \mu(A_n) < \infty$. A real is *Solovay random* if it passes the Solovay condition on all Solovay tests.

In this sense, the class of randoms obtained by imposing the Solovay condition over the class of Solovay tests is the class of reals that obey the Borel-Cantelli lemma on all Σ^0_1-presented statistical tests. More well-known is the definition of Martin-Löf randomness:

Definition 3.3. A *Martin-Löf test* is a sequence $\{A_n\}_{n<\omega}$ of uniformly Σ^0_1 classes in 2^ω with $\mu(A_n) \leq 2^{-n}$ for all n. A real is *Martin-Löf random* if it passes the Martin-Löf condition on all Martin-Löf tests.

In this sense, the Martin-Löf randoms are those which are not covered by an effective sequence of covers with measure that can effectively be made arbitrarily small. At its face, Solovay randomness seems a stronger notion than Martin-Löf randomness. However, in a short and slick argument, Solovay observed that a real is Solovay random if and only if it is Martin-Löf random [63].

Although these definitions conform well to our intuition about what an effective version of randomness should look like, under either the probability paradigm or the measure-theoretic paradigm, the class of Martin-Löf randoms fails to satisfy some of the computational conditions we might expect of a randomness notion. For example, given a noncomputable set A, the collection of reals $\{X : X \geq_T A\}$ has measure zero. Under the informal notion of randomness, we would say that, given any non-computable set A, a "typical" or "random" set does not compute it [11, 58]. Of course, no randomness

notion will guarantee this sort of typicality, since a non-computable set A will always compute itself, but the class of Martin-Löf randoms fails this entire intuition even more spectacularly: given any set A, there exists a Martin-Löf random X such that $X \geq_T A$ [23, 37].

There are many ways to strengthen the notion of Martin-Löf randomness which eliminate some of the potential for a random to be computationally useful. For example, the notion of weak 2-randomness, although originally defined as the collection of reals which belong simultaneously to every Σ^0_2 class of measure one [38], can be obtained by imposing the Martin-Löf condition over a superclass of Martin-Löf tests:

Definition 3.4. A *generalized Martin-Löf test* $\{A_n\}_{n<\omega}$ is a nested sequence of uniformly Σ^0_1 classes in 2^ω with $\mu(A_n) \to 0$.

Theorem 3.5 (Wang [65]). A real is weak 2-random if and only if it passes the Martin-Löf condition on every generalized Martin-Löf test.

Downey, Nies, Weber and Yu showed that each weak 2-random forms a minimal pair with the halting problem $\emptyset'$ [16]: i.e., for X a weak 2-random, and any set Y, if $Y \leq_T X$ and $Y \leq \emptyset'$, then $Y \leq \emptyset$. Hirschfeldt and Nies showed the converse, so the property of forming a minimal pair with $\emptyset'$ characterizes the weak 2-randoms among the Martin-Löf randoms[1]. For example, no weak 2-random is Δ^0_2, i.e. no weak 2-random is computable from $\emptyset'$.

Another strengthening of Martin-Löf randomness is *Demuth randomness*, obtained by imposing the Solovay condition over the class of *Demuth tests*. Imposing the Martin-Löf condition gives a weaker notion, aptly named *weak Demuth randomness*.

[1]Hirschfeldt and Nies' result is unpublished, the proof appears in [51].

Definition 3.6. A *Demuth test* is a class $\{A_n\}_{n\in\omega}$ of Σ^0_1 classes in 2^ω with $\mu(A_n) \leq 2^{-n}$, whose components are given by some ω-c.e. function $g(n)$ as $A_n = [W_{g(n)}]$ for all n.

Recall that a function $g(n)$ is ω-c.e. if it has a computable approximation $g(n,s)$ such that the number of mind-changes $|\{s : f(n,s) \neq f(n,s+1)\}|$ is bounded by a computable function $f(n)$. Given $e \in \omega$, W_e denotes the e'th computably enumerable set under a fixed effective listing of all c.e. sets.

Definition 3.7. A real $X \in 2^\omega$ is *Demuth random* if it passes the Solovay condition on every Demuth test, and *weak Demuth random* if it passes the Martin-Löf condition on every Demuth test.

Again, Demuth randomness ensures some degree of computational weakness: every Demuth random is GL_1 [49].

A particularly striking dichotomy appears at the level of computing $\emptyset'$. By a result of Stephan, a Martin-Löf random real is either computationally strong enough to compute the halting problem (that is, it is complete), or is computationally too weak to compute any complete extension of Peano Arithmetic [64]. The typical case is the latter, since the class of reals computing $\emptyset'$ (or any fixed non-computable set) is of measure 0. Remarkably, this class of incomplete Martin-Löf randoms can be characterized by a natural test notion, introduced by Franklin and Ng [22].

Definition 3.8. A *difference test* is a sequence $\{U_n \setminus V_n\}_{n\in\omega}$ where $\{U_n\}$ and $\{V_n\}$ are uniformly Σ^0_1 classes of subsets of 2^ω, and each component has measure $\mu(U_n \setminus V_n) \leq 2^{-n}$.

Definition 3.9. A real $X \in 2^\omega$ is *difference random* if it pases the Martin-Löf condition on every difference test.

Then Franklin and Ng showed:

Theorem 3.10 (Franklin and Ng [22]). A Martin-Löf random real X is difference random if and only if $X \not\geq \emptyset'$.

Bienvenu and Porter considered imposing the strong condition on these tests [3]:

Definition 3.11 (Bienvenu and Porter). A real $X \in 2^\omega$ is *strong difference random* if it passes the Solovay condition on every difference test.

The strong difference randoms fit in between the Demuth randoms and the difference randoms, incomparable with the weak Demuth randoms and the weak 2-randoms [3]. Bienvenu and Porter were also able to show that the strong difference randoms satisfy many of the computational properties we would desire of a randomness notion, including that they are all GL_1 [3].

We continue the study of strong difference randomness along the lines of Franklin and Ng's study of the difference randoms. Franklin and Ng characterized the difference randoms by imposing the Martin-Löf condition on a subclass of Demuth tests which they called the *strict Demuth tests*. We present a characterization of strong difference randomness as that class obtained by imposing the Solovay condition over the same class of strict Demuth tests. Then, we present a computability-theoretic characterization of the strong difference randoms among the Martin-Löf randoms as those Martin-Löf randoms which are array computable in a very nice way, a notion of computational weakness which we will make precise and explore in its own right.

3.2 Strict Demuth tests

Franklin and Ng characterized the difference randoms in terms of *strict Demuth tests*:

Definition 3.12. A Demuth test $\{[W_{g(n)}]\}$ is *strict* if $g(n)$ has a computable approximation $g(n,s)$ such that for every n and s, if $g(n,s) \neq g(n,s+1)$, then $[W_{g(n,s+1)}] \cap \left[\bigcup_{t \leq s} W_{g(n,t)}\right] = \emptyset$.

Theorem 3.13 (Franklin and Ng). A real $X \in 2^\omega$ is difference random if and only if it passes the Martin-Löf condition on every strict Demuth test.

We provide a similar characterization for the strong difference randoms. Namely, the strong difference randoms are those passing the strong (Solovay) condition on all strict Demuth tests. Our proof is similar to that of Franklin and Ng's theorem above.

A key step in our proof is a simple combinatorial observation: each infinite set of integers contains either infinitely many even integers, or infinitely many odd integers. We apply this observation non-uniformly to speed up the rate at which the size of the components of our difference tests converge to zero. When applied to difference tests with the Martin-Löf condition, this step can be done uniformly, and followed mutatis mutandis provides a simplified proof of Franklin and Ng's theorem.

Theorem 3.14. A real $X \in 2^\omega$ is strong difference random if and only if it passes the Solovay condition on every strict Demuth test.

Proof. If $\{[W_{g(n)}]\}_{n \in \omega}$ is a strict Demuth test, then setting $U_n = \bigcup_s [W_{g(n,s)}]$ and $V_n = \bigcup \{[W_{g(n,s)}] : g(n,s) \neq g(n,s+1)\}$ gives a difference test with $U_n \setminus V_n = [W_{g(n)}]$ for each n, so if X is in infinitely many components of a strict Demuth test, then it is not strong difference random.

In the other direction, suppose that X is not strong difference random, and take $\{U_n \setminus V_n\}$ a difference test for which X belongs to infinitely many components. By speeding up the enumeration of V_n, we may assume without loss of generality that for

every n and s, $\mu\{U_{n,s} \setminus V_{n,s}\} \leq 2^{-n}$. Since X belongs to infinitely many components of $\{U_n \setminus V_n\}$, it belongs to infinitely many odd components, or infinitely many even components. Knowing which (non-uniformly), culling appropriately either all of the even or all of the odd components from our test and re-indexing, we may assume without loss of generality that X is captured in infinitely many components of a difference test $\{U_n \setminus V_n\}$ and that for every n and s we have $\mu\{U_{n,s} \setminus V_{n,s}\} \leq 4^{-n}$.

Largely following Franklin and Ng, we now build a strict Demuth test $\{W_{g(n)}\}_{n\in\omega}$ and a Solovay test E (note that a Solovay test is a strict Demuth test, with static components) so that X fails the Solovay condition on at least one of them.

We build $W_{g(i)}$ as follows: initially, $W_{g(i,s)}$ copies U_i until the measure of $U_{i,s}$ exceeds 2^{-i}. Then we enumerate $U_{i,s} \setminus V_{i,s}$ into E. On the k'th attempt to build $W_{g(i,s)}$, for $W_{m_1}, \ldots, W_{m_{k-1}}$ the abandoned versions of $W_{g(i,s)}$, we let our current version of $W_{g(i,s)}$ follow $U_{i,s} \setminus \big([W_{m_1} \cup \ldots \cup W_{m_{k-1}}]\big)$, again until its measure exceeds 2^{-i}, at which point we abandon it and again throw $U_{i,s} \setminus V_{i,s}$ into E. For each i, we can only abandon a version of $W_{g(i,s)}$ at most 2^i times before running out of measure, so the $g(i,s)$ so constructed eventually settles with $\leq 2^i$ changes. And since each set enumerated into E on behalf of i contributes only 4^{-i} much weight, and this is done at most 2^i times, E is a Solovay test.

Suppose X is in the component $U_n \setminus V_n$, if $X \notin [W_{g(n)}]$, then since $[W_{g(n)}]$ eventually settles to some $U_n \setminus ([W_{m_1} \cup \ldots \cup W_{m_k}])$, it must belong to one of the $[W_{m_i}]$, which means it appeared in $U_{n,s}$ for some early s, so was in $U_{n,s} \setminus V_{n,s}$ which was put in E on behalf of W_{m_i}.

So for each of the infinitely many components of $\{U_n \setminus V_n\}_{n\in\omega}$ that contain X, either X belongs to the n'th component of E, or to $W_{g(n)}$. So either X belongs to infinitely

many components of $\{W_{g(n)}\}_{n\in\omega}$, or infinitely many components of E. In either case, X is witnessed to fail a strict Demuth test. □

3.3 Jump domination

The following lowness notion was introduced in [20]:

Definition 3.15. A set A is *ω-c.e.-jump dominated* if for each A-partial computable function Θ^A, there exists an ω-c.e. function $f(n)$ so that $\Theta^A(n)$ is dominated by $f(n)$.

Note that since the jump J^A is universal for A-partial computable functions, it is enough that J^A be dominated by an ω-c.e. function, which is indeed the definition given in [20]. Among the well-known lowness notions, ω-c.e. jump domination is implied by jump traceability and implies array computable, but also implies GL_1 [20]. For more background on lowness notions, including definitions of these classes, we recommend [49] Chapter 8, although they will not be needed for the results of this chapter.

Given a computable approximation $f(i,s)$ to an ω-c.e. function $f(i)$, we often consider the *number of mind-changes* $m(i) = |\{s : f(i,s) \neq f(i,s+1)\}|$. We introduce the following notion:

Definition 3.16. *A is 2^n-c.e.-jump dominated* if for each A-partial computable function Θ^A, there exists an ω-c.e. function $f(n)$ so that $\Theta^A(n)$ is dominated by $f(n)$, and $f(n)$ has a computable approximation $f(n,s)$ such that for all n, $|\{s : f(n,s) \neq f(n,s+1)\}| \leq 2^n$.

It should be noted that the specific choice of 2^n is not essential. We could similarly define a^n-c.e.-jump dominated for any fixed constant $a > 1$. Then given any A-partial computable function Θ^A, by splitting Θ^A into $\Psi^A_j(n) = \Theta^A(j+nk)$ for $j = 1, \ldots, k-1$,

with $k = \lceil \log_2 a \rceil$, then bounding each $\Psi_j^A(n)$ with $\leq a^n$ mind-changes and combining these bounds gives a bound on Θ^A with mind-changes eventually bounded by 2^n.

We do insist, however, that the bound on mind-changes be given uniformly by a fixed exponential gauge: if we allow a to vary, then again dominating J^A with exponentially-bounded mind-changes would be enough to bound any A-partial computable function with exponentially-bounded mind-changes (with exponential bounds of varying bases), as we can write a pairing function $\langle i, j \rangle$ which grows linearly in the second coordinate.

In [49], Theorem 3.6.26, Nies shows that all Demuth randoms are GL_1 by showing that they are in fact ω-c.e.-jump dominated. He proceeds by setting $\Theta^A(i) = \mu s : J_s^A(i) \downarrow$, then Nies builds $f(i, s)$ with mind-changes bounded by 2^i, together with a Demuth test (in fact, a difference test), so that any random passing with Solovay condition has $\Theta^A(i)$ bounded by $f(i)$. Given an arbitrary A-partial computable function Ψ^A, by replacing J^A by the functional which converges on input i (say to 1) at stage $\geq \Psi^A(i)$, the exact same argument goes through giving an ω-c.e. bound now on $\Psi^A(i)$, with the same 2^i bound on mind-changes. As a result:

Corollary 3.17. Each strong difference random is 2^n-c.e.-jump dominated.

Among the MLR reals, the converse of this theorem also holds. Hence, the strong difference randoms are precisely those randoms which are 2^n-c.e.-jump dominated.

Theorem 3.18. If A is MLR and 2^n-c.e.-jump dominated, then A is strong difference random.

Proof. Suppose that A is not SDR, then A is in infinitely many components of $\{W_{g(i)}\}_{i \in \omega}$, where $W_{g(i,s)}$ is a strict Demuth test.

Hence A is in infinitely many components of $\{W_{g(2i)}\}_{i \in \omega}$ or $\{W_{g(2i+1)}\}_{i \in \omega}$. So without

loss of generality, we may assume A is in some test $\{W_{g(i)}\}_{i\in\omega}$ where $|W_{g(i)}| \leq 4^{-i}$.

Suppose moreover now that A is 2^n-c.e.-jump dominated. Applying this condition to $\Theta^A(m) = \mu s : A \searrow W_{g(m,s),s}$, we obtain $f(m,s)$ with $\leq 2^m$ mind-changes, dominating $\Theta^A(m)$.

We now build a Solovay test E capturing A. Each time $f(i,s)$ gives output, for $t = f(i,s)$ enumerate $W_{g(i,t),t}$ into E for up to 4^{-i} much measure. Since $f(i,s)$ changes only up to 2^i times, E is a Solovay test. And since $f(i,s)$ dominates $\Theta^A(i)$, we know that A is captured by E. $\square$

Chapter 4

Strong Jump-Traceability and Diagonal Non-Computability

4.1 The strong jump-traceables and randomness

The notion of strong jump-traceability was introduced by Figueira, Nies and Stephan [21] as a modification of Nies' notion of jump-traceability [46] in an attempt to capture the class of K-trivials using traditionally combinatorial ideas of computability theory, rather than the analytic ideas characteristic of algorithmic randomness. Although we will not explore the class of K-trivials directly in this chapter, it suffices to say that their definition arises from Kolmogorov complexity, and they have many characerizations intimately related to algorithmic randomness. Therefore, a purely combinatorial characterization of the class would be highly desirable (see also [41, 48]). Instead of producing such a characterization, the strong jump-traceables emerged as a sub-ideal of the K-trivials, with weaker notions of jump-traceability fleshing out a hierarchy of classes within the K-trivials [14, 15].

A set A is *strongly jump-traceable (SJT)* if for every *order function* h, every A-partial computable function has a *trace* bounded by h. Definitions of all these terms can be found in [21], but it suffices to remark that they are of a combinatorial flavor.

In contrast to this combinatorial definition, the strong jump-traceables also have an characerization in terms of the more analytic notion of *cost functions*, which we will be using in this chapter. This alternative analytic perspective made the strong jump-traceables a compelling, albeit unsuccessful, candidate for characterizing the K-trivials.

Definition 4.1. A *monotone cost function* is a computable function c of two variables. We say that it assigns a *cost* $c(x, s)$ to changing a Δ^0_2 approximation at position x at stage s. We require that, for each fixed x, $c(x, s)$ is non-decreasing in s and converges to a limit, $c(x)$, and that for each fixed s the cost $c(x, s)$ is non-increasing in x (in particular, $c(x)$ is non-increasing).

Definition 4.2. A Δ^0_2 approximation $\langle A_s \rangle$ *obeys* a cost function $c(x, s)$ if the sum of all $c(x_s, s)$ is finite, where x_s range over those $x = x_s$ which are least such that $A_s(x) \neq A_{s+1}(x)$.

Definition 4.3. A cost function $c(x, s)$ is *benign* if there is a computable function g so that for all positive rational ϵ, the size $|\mathfrak{I}|$ of any collection $\mathfrak{I}$ of pairwise disjoint intervals $[x, s)$ with $c(x, s) \geq \epsilon$ is bounded by $g(\epsilon)$: i.e. $|\mathfrak{I}| \leq g(\epsilon)$.

Then we have the following theorem, which for our purposes in this chapter we will take as a definition:

Theorem 4.4 (Greenberg, Nies [26]). A Δ^0_2 set A is *strongly jump-traceable* if it obeys every benign cost function. That is, if for any benign cost function c, there is a Δ^0_2 approximation of A obeying c.

Of particular interest in the body of results concerning the K-trivials have been various characterizations of the K-trivials by different notions of computational lowness. One notion is that of being computed by "many" oracles: a set A is K-trivial if and

only if it is computed by some MLR^A set: i.e. if A is computed by a set of oracles that is not effectively small (in A) [47].

Similar characterizations of the SJTs have been found characterizing them as the reals computable from some "large" class of oracles. Strikingly, the theorems split into two archetypes: if a set is computed by random and computationally weak oracles, it is strongly jump-traceable, but the same holds if a set is computed by random and computationally strong oracles.

We say that a set A *is weak truth-table reducible to a set* B, written $A \leq_{wtt} B$, if there is a Turing reduction from B to A with computably bounded use.

A set A is *superlow* if $A' \leq_{wtt} \emptyset'$, i.e. if A' is ω-c.e., equivalently, if every set c.e. relative to A is ω-c.e. [44]. In particular, A superlow implies that A itself is ω-c.e. A set A is *superhigh* if $\emptyset'' \leq_{wtt} A'$.

Theorem 4.5 (Greenberg, Hirschfeldt and Nies [25]). For each of the following classes, the strong jump-traceables are precisely the collection of degrees computable in each of their members:

(1) The MLR superlows.

(2) The MLR ω-c.e. sets.

(3) The MLR superhighs.

These theorems were proven first for the c.e. strong jump-traceables by Greenberg, Hirschfeldt, and Nies [25]. Of fundamental importance but established later by Diamondstone, Greenberg, and Turetsky, strong jump-traceability is inherently enumerable: every strong jump traceable can be bounded by a c.e. strong jump-traceable [14].

In the cases (1) and (2), there is reason to suspect that the results are not essentially related to randomness of the class of oracles. Indeed, [25] point out a corollary to cases

(1) and (2) in the PA degrees. Recall that a set A is of *PA degree* or is *PA-complete* if A computes a path through every non-empty Π^0_1 class, if and only if A computes a 0–1 valued DNC function, if and only if A computes a complete, consistent extension of Peano Arithmetic [34].

Theorem 4.6 (Greenberg, Hirschfeldt and Nies [25]). A set is strongly jump traceable if and only if it is computable from every superlow (or from every ω-c.e.) PA degree.

In the following section, we will show that in Theorem 4.5, for (1) and (2), MLR can be replaced by the weaker combinatorial property of DNC. A sequence $X \in \omega^\omega$ is diagonally non-computable (DNC) if it differs in each diagonal position from the universal Turing machine: $X(e) \neq \phi_e(e)$ for each e. For example, each Martin-Löf random computes a DNC sequence [37].

Since in the nontrivial forward direction, this replaces the MLRs with the larger class of DNCs, we have not only reason to suspect, but grounds to assert, that the low cases (1) and (2) are in fact not essentially related to randomness of the oracles.

To us, this suggests that the characterization of SJTs as computable from every computationally low random are not intimately related to the randomness of the oracle: they only esssentially use a combinatorial property which is true of randoms. We will however note that characterization (3) cannot be improved in this way—the collection of sets computable from every DNC superhigh (in fact every DNC LR-hard) is precisely the collection of computable sets—leaving open the possibility that (3) is indeed deeply connected to randomness.

4.2 Strong jump-traceables and DNCs

Theorem 4.7. If A is strongly jump-traceable, then A is computable from every ω-c.e. DNC real.

Before proving the theorem, a corollary:

Corollary 4.8. A is strongly jump-traceable if and only if A is computable from every superlow (or from every ω-c.e.) DNC real.

Every ML-random has DNC degree [37], so the "if" direction follows directly from Theorem 4.5, the "only if" direction follows from the main theorem, and the fact previously observed that superlow implies ω-c.e.

To prove the theorem, we adapt the proof of Greenberg and Nies in Proposition 5.1 of [26].

Proof. Let $Y \in \omega^\omega$ be Δ^0_2 and DNC, and fix $\langle Y_s \rangle$ a Δ^0_2 approximation.

Given f computable, increasing, we define a cost function $c(x,t) = c_f(x,t)$ as follows:

Let $c(x,0) = 2^{-x}$. For $t > 0$, put $n = n_t$ the least n such that $Y_t \upharpoonright f(n) \neq Y_{t-1} \upharpoonright f(n)$, and set $c(x,t) = \max\{c(x,t-1), 2^{-n}\}$ for $x < t$.

Claim: if $\langle Y_s \rangle$ is an ω-c.e. approximation, then c is benign.

Proof: Certainly c is monotone and has the limit condition. If Y is ω-c.e. via $\langle Y_s \rangle$ with witness g, that is, g is computable and $Y_s(m) \neq Y_{s-1}(m)$ on at most $g(m)$ many stages, then given $\mathfrak{I}$ a set of pairwise disjoint intervals with each $[x,s) \in \mathfrak{I}$ yielding $c(s,x) \geq 2^{-n}$, then for all $[x,s) \in \mathfrak{I}$ with $x > n$, there is a $t \in (x,s]$ such that $Y_t \upharpoonright f(n) \neq Y_{t-1} \upharpoonright f(n)$. But then $|\mathfrak{I}| \leq (n+1) + \Sigma_{m<f(n)} g(m)$, which proves the claim.

We will choose $f(n)$ so that effectively during the construction, we control $n \cdot 2^n$ places below $f(n)$ in the diagonal function: we allot 2^n places for each index $e < n$. We will see

our construction is computable, so we can write an auxilliary partial computable function that diagonalizes its reflected places in the diagonal function against the corresponding position of Y_t at the current stage. Then $f(n)$ total computable is chosen so that we have control over sufficiently many positions. Because f is computable, by the recursion theorem we have access to its index in the construction following.

Let $\langle A_{e,t} \rangle$ list all Δ^0_2 approximations. If $A_{e,t} \to A_e$ obeys $c(x,t)$, we claim that $Y \geq_T A_e$.

We introduce some notation which will be crucial to the proof: for each n and $t \geq n$, let $s_n(t)$ be least $s \leq t$ such that $\forall r \in [s,t]$ $Y_r \upharpoonright f(n) = Y_t \upharpoonright f(n)$. So $s_n(t)$ is the time of the last change in $Y_s \upharpoonright f(n)$ before stage t.

At stage t, if n is the least such that $A_{e,t} \upharpoonright s_n(t) \neq A_{e,t-1} \upharpoonright s_n(t)$, then converge the diagonal function to a position of $Y_t \upharpoonright f(n) - 1$. Do this for each $e < n$ that have not yet met their allotment.

If $\langle A_{e,t} \rangle$ obeys c, then it cannot infinitely often make 2^n changes at cost $\geq 2^{-n}$. So eventually, any change in $\langle A_{e,t} \rangle$ is responded to by a convergence agreeing with Y_t. That is more precisely, a change $A_{e,t} \upharpoonright s_n(t) \neq A_{e,t-1} \upharpoonright s_n(t)$ means (by defintion of c) a change of cost $\geq 2^{-n}$.

So after some finite stage, we have enough changes allotted to e to respond to any change in $A_{e,t}$. So for fixed e, wait until $\langle A_{e,t} \rangle$ has settled in any place where it changes at least 2^n times at cost $\geq 2^{-n}$. Say by stage s'.

Claim: if $t > s'$, then the following implication holds:

$$Y \upharpoonright f(n) - 1 = Y_t \upharpoonright f(n) - 1 \Longrightarrow A_e \upharpoonright s_n(t) = A_{e,t} \upharpoonright s_n(t).$$

From this implication we have $Y \geq_T A_e$ for Y DNC, since we will certainly have $s_n(t)$

unbounded by definition of $s_n(t)$, and each time we see $Y_t \upharpoonright f(n)-1 = Y \upharpoonright f(n)-1$ we can obtain $A_e \upharpoonright s_n(t)$.

Proof of claim: in fact, we claim $A_s \upharpoonright s_n(t)$ never changes after stage t. Suppose $u > t$ is a stage where $A_{e,u} \upharpoonright s_n(t) \neq A_{e,u-1} \upharpoonright s_n(t)$. Let m be least such that $A_{e,u} \upharpoonright s_m(t) \neq A_{e,u-1} \upharpoonright s_m(t)$.

Since s_n is non-decreasing, $s_n(u) \geq s_n(t)$, so $A_{e,u} \upharpoonright s_n(u) \neq A_{e,u-1} \upharpoonright s_n(u)$. So by minimality, $m \leq n$.

Since $u > s'$, we converge a position of the diagonal function to agree with $Y_u \upharpoonright f(m)-1$. So $Y_u \upharpoonright f(m)-1$ cannot agree with Y. Minimality of m implies

$$A_{e,u} \upharpoonright s_{m-1}(u) = A_{e,u-1} \upharpoonright s_{m-1}(u),$$

so by assumption on u, $s_{m-1}(u) < s_n(t)$ and $s_n(t) \leq t$ by definition of s_n. Hence:

$$Y_u \upharpoonright (f(m)-1) = Y_t \upharpoonright (f(m)-1) \prec Y_t \upharpoonright (f(n)-1).$$

But $Y_t \upharpoonright (f(n)-1)$ agrees with $Y \upharpoonright (f(n)-1)$ by assumption, a contradiction. □

Recall that any MLR A computes a DNC real via a finite variation of the function $n \mapsto A \upharpoonright n$ [37], notice that this map is indeed a wtt reduction (the use is bounded by n), and by definition wtt reductions preserve the properties of ω-c.e. and superlow.

A set which wtt-computes a DNC is called wtt-DNC, e.g. any MLR is wtt-DNC. Under this terminology, we note that our proof gives the following:

Corollary 4.9 (Of proof). *A is strongly jump-traceable if and only if A is computable from every superlow (or from every ω-c.e.) wtt-DNC set.*

We see that this statement improves (in the nontrivial forward direction) the theorems for superlow and ω-c.e. MLRs, while replacing the randomness content with the

purely combinatorial property of wtt-DNC.

4.3 The superhigh case

An analogous characterization to Theorem 4.5 (3) fails entirely. The only sets computable in every superhigh DNC are the computable sets. In fact, following Nies [47], we say a set A is *LR-hard* if $MLR(A) \subseteq MLR(\emptyset')$. In general, $A \leq_{LR} B$ if $MLR(B) \subseteq MLR(A)$. In particular, $A \leq_T B$ implies that $A \leq_{LR} B$ ($\leq_{LR}$ is a "weak reducibility"), and the notion that A is LR-hard means precisely that $\emptyset' \leq_{LR} A$. Simpson proved that every LR-hard set is superhigh [60], and we have:

Theorem 4.10. The collection of sets computable from every LR-hard PA degree is the collection of computable sets.

Proof. Suppose $Z \not\geq_T A$. By the relativized Cone Avoidance Basis Theorem [24], for any nonempty $\Pi^0_1[Z]$ class $\mathcal{P}$, there is an $x \in \mathcal{P}$ that does not compute A. And there is a nonempty $\Pi^0_1[Z]$ class consisting only of PA$[Z]$-degree sets (in fact, of sets which are all 0–1 valued DNC$[Z]$ sets [34]), therefore there is an X of PA$[Z]$-degree, in particular an X which is of PA degree and for which $X \geq_T Z$, such that also $X \not\geq_T A$.

Considering Z an LR-hard set, then X is of PA degree and LR-hard. So we see the collection of sets computable from every LR-hard PA degree is all of the sets computable from every LR-hard degree. But this collection is known to be the computable sets by a result of Cholak, Greenberg and Miller [9]: they show that the property of almost-everywhere domination, which is equivalent to LR-hardness [36], can be forced by a notion that admits cone avoidance. That is, for any noncomputable set A, there is a uniformly almost-everywhere dominating f that does not compute A. □

Chapter 5

Slopes of computable functions

5.1 Introduction

This chapter concerns real-valued functions which are computable in the classical sense of Lacombe and Grzegorczyk. For an exposition of the basic notions of computable analysis, we recommend [52, 66]. Informally, a real valued function is computable if there is an effective algorithm which, given an effective approximation to an input real x, produces an effective approximation to the output value $f(x)$.

Precisely, we fix a standard effective listing $\{p_n\}_{n\in\omega}$ of all the rational polygonal functions on $[0,1]$—i.e. the polygonal functions made of segments whose endpoints with rational coordinates—then a function $f(x) \in \mathcal{C}[0,1]$ is *computable* if there is a total computable function $\lambda : \omega \to \omega$, called a *representation* of f, such that $\max_{x\in[0,1]}|f(x) - p_{\lambda(n)}(x)| \leq 2^{-n}$. In other words, $f(x)$ is an effective uniform limit of rational polygonal functions. The same class of computable real-valued functions is obtained if, rather than the standard listing of rational polygonal functions, we use a standard effective listing of the polynomials with rational coefficients [52].

Our intuition from classical real analysis is that functions which are analytically tame (e.g. smooth, smoothly differentiable, etc.) ought to behave well in a computational sense. This intuition is confirmed for example by a theorem of Pour-El and

Richards, from early in the development of computable analysis, which established that if a computable function f is twice continuously differentiable, the derivative of f is itself a computable function [52].

Classical theorems in computable analysis concerning differentiability consider the behavior of f and its derivatives as global functions given by effective representations. More recent work (and some rediscovered work in the language of constructivism) has turned toward the points of differentiability themselves, yielding a connection between classical differentiability theorems and algorithmic randomness. For example, the common Lebesgue points of all L^1 computable functions are the Schnorr randoms [50] [57], an effective analogue of the Lebesgue density theorem. Similarly, the common points of differentiability of all monotone computable functions are the computable randoms, an effective analogue of the monotone differentiation theorem [5]. The points of differentiability of all computable functions of bounded variation are precisely the Martin-Löf randoms [5, 13]. A similar theorem characterizes the Kurtz randoms [43].

With a newfound interest in particular points of differentiability it is natural to ask about the corresponding values of $f'(x)$, that is, how computationally complicated are the slopes of computable functions? Given any computable function f, it is easy to see that the value of $f'(x)$, whenever it exists, is uniformly computable in the jump of x. Pour-El and Richards' result tells us that, given a $\mathcal{C}^2$ computable function, the values $f'(x)$ are uniformly computable in x. We pursue this perspective further, asking questions of the following form: given some degree of good analytic behavior for f, how computationally complicated are the values of the derivative $f'(x)$? We will see that even at the level of $\mathcal{C}^1$, there is a computable function f so that the values of $f'(x)$ compute the jump of x almost everywhere: that is, they have the highest degree possible.

At this point, we should clarify a notational issue: throughout this chapter, when f is a function, we will write $f'(x)$ to denote the value of the derivative of the function f at the point x. When x denotes a number, we will write x' to denote the Turing jump of the real x, i.e. the halting set relative to (the binary expansion) of x. We will also use $\emptyset'$ to denote the halting set itself. So for example, when we write $f'(x) \geq_T x'$ we mean that the value of the derivative of the function f at the point x computes the halting set relative to the real x, when we write $f'(x) \geq_T \emptyset'$ we mean that the value of the derivative of the function f at the point x computes the halting set.

In addition to constructions of $\mathcal{C}^1$ functions, we explore what happens as we vary the smoothness and differentiability criteria for f, for example we consider what happens when f is asked only to be merely differentiable almost everywhere, or when f is asked to be differentiable everywhere. In these cases we will see that the possible computational complexity of $f'(x)$ interacts with different computational paradigms, both uniform and non-uniform. We also construct several examples of functions where $f'(x) \geq_T \emptyset'$. Although it is not possible for a $\mathcal{C}^1$ function f to give $f'(x) \geq_T x'$ uniformly almost everywhere, or even $f'(x) \geq_T \emptyset'$ uniformly almost everywhere, it is possible for the latter condition (and not the former) to be satisfied with layerwise uniformity, a concept from algorithmic randomness.

5.2 A first construction

We should first note that the mean-value theorem guarantees that the values of a derivative cannot all be computationally complicated.

Remark 5.1. If f is a differentiable, computable function, then $f'(x)$ is computable on

a dense set of points.

Proof. By the mean value theorem, between any two computable reals a, b, there exists a point $c \in [a, b]$ for which $f'(c) = \frac{f(b)-f(a)}{b-a}$, and hence for which $f'(c)$ is computable. □

This means that we cannot hope to construct any computable function $f(x)$ for which $f'(x)$ is of high complexity everywhere. In light of this restriction, we will constrain ourselves to almost-everywhere results. We start by constructing a computable function f with $f'(x) \geq \emptyset'$ almost everywhere:

Theorem 5.2. There exists a computable function f for which $f'(x) \geq_T \emptyset'$ uniformly in x for almost all x.

Proof. We build $g(x)$ (which we will want to look like our $f'(x)$) on $[0, 1]$ as a sum of waveforms. Given a partition $\mathcal{P}$ of $[0, 1]$, by an *alternating step function of depth a on* $\mathcal{P}$ we mean a piecewise-constant function alternating between a and $-a$ on the intervals defined by $\mathcal{P}$.

Let K be the halting set (we can also do the same with any c.e. set). Fix a list of dyadic meshes $\left\{\frac{i}{2^n} : 0 \leq i \leq 2^n\right\}$, in order of fineness. Fix a computable listing $\{a_n\}$ of K. We start with $g_0(x) = 1/2$ (so the ternary expansion of g_0 is all 1s).

At stage $n + 1$, we obtain $g_{n+1}(x)$ by summing $g_n(x)$ with another step function as follows: when a_n enters K, we add an alternating step function of depth 3^{-a_n} on the first unused diadic mesh whose fineness δ satisfies $3^{-a_n}\delta < 2^{-n-1}$ (that is, whose steps have areas each $< 2^{-n-1}$).

Let $g_n(x)$ be the sum of the first n waveforms so enumerated, and $g(x) = \lim_{n\to\infty} g_n(x)$ pointwise. Summing the first n waveforms, and integrating up to some fixed $x \in [0, 1]$, we miss at most one step of each remaining waveform. So $|\int_0^x g(t)dt - \int_0^x g_n(t)dt| \leq$

$\sum_{i=n}^{\infty} 2^{-i-1} = 2^{-n}$, hence the $\int_0^x g_n(t)dt$ converge uniformly and effectively to $\int_0^x g(t)dt$. Morevoer, $\int_0^x g_n(t)dt$ are uniformly computable, therefore $f(x) := \int_0^x g(x)dx$ is computable.

We claim that $f'(x) = g(x)$ for x not dyadic-rational. Let $\tilde{g}_n$ denote the sum of the first n step functions by decreasing depth. Note $\tilde{g}_n$ then approximates $g(x)$ uniformly within $\sum_{i=n}^{\infty} 3^{-i} = \frac{1}{2 \cdot 3^{n-1}}$, so the convergence $\tilde{g}_n \to g(x)$ is uniform. Consider the difference quotient for f:

$$f'(x) = \lim_{h \to 0} \frac{\int_x^{x+h} g(t)\ dt}{h} = \lim_{h \to 0} \frac{\int_x^{x+h} \tilde{g}_n(t) + R_n(t)\ dt}{h}$$

where $R_n(x) = g(x) - \tilde{g}_n(x)$ is bounded in absolute value by $\frac{1}{2 \cdot 3^{n-1}}$. Splitting the sum and taking the first limit, we get, for x not dyadic-rational:

$$f'(x) = \tilde{g}_n(x) + \lim_{h \to 0} \frac{\int_x^{x+h} R_n(t)\ dt}{h}$$

and applying our bound on R_n yields

$$|f'(x) - \tilde{g}_n(x)| \leq \frac{\int_x^{x+h} 2^{-1}3^{1-n}dt}{h} = \frac{h2^{-1}3^{1-n}}{h} = 2^{-1}3^{1-n}$$

so $\tilde{g}_n(x) \to f'(x)$, hence $f'(x) = g(x)$.

Moreover, for any point x, $g(x)$ computes K: as $a_n \in K$ if and only if the n'th term in the ternary expansion of $g(x)$ is not 1. □

5.3 Smoothing things out

So far we have claimed no analytic conditions on the functions we have constructed, other than that they are differentiable almost everywhere. To achieve good analytic behavior, we must at some point sacrifice computational uniformity:

Remark 5.3. There does not exist any function f smooth of class $\mathcal{C}^1$ such that $f'(x) \geq_T \emptyset'$ uniformly almost everywhere.

Proof. By way of contradiction, let f be $\mathcal{C}^1$ and Φ_e a functional with $\emptyset' = \Phi_e(f'(x))$ almost everywhere. Then Φ_e itself computes $\emptyset'$ from some subset of $[0,1]$. We claim that this reduction fails on densely many intervals of $[0,1]$.

Given any interval I of $[0,1]$, consider any subinterval of the form $(\sigma^\frown 0, \sigma^\frown 1)$ in I, with σ a finite binary expansion. We can then write a program i which continues testing, under Φ_e, extensions of σ until it finds an extension $\sigma^\frown\tau$ where Φ_e converges to 0 on i, and then halts. So Φ_e fails on some subinterval $(\sigma^\frown\tau, \sigma^\frown\tau^\frown 1)$ of I.

Now $f'(x)$ is continuous, and clearly cannot be constant (or else, applying the mean value theorem on two computable reals, it would be computable), so the image of $f'(x)$ passes through some interval where the reduction Φ_e fails. The inverse image of this interval then itself contains an interval where the uniform reduction $f'(x) \geq_T \emptyset'$ fails. □

We are able, however, to achieve a weaker notion of uniformity. In the papers [30,31], Hoyrup and Rojas introduced the framework of *layerwise computability*, which formalize versions of effective measurability corresponding to different randomness notions. We are particularly interested in the version corresponding to Schnorr randomness. For an introduction to the basic notions of algorithmic randomness, see [17] or [49], also recall Definition 3.3.

Definition 5.4. A Martin-Löf test $\{U_n\}$ is a *Schnorr test* if $\mu(U_n)$ are uniformly computable.

Definition 5.5. $f(x)$ is *Schnorr layerwise computable* if there is a Schnorr test (U_n) such that $f(x)$ is uniformly computable on $X \setminus U_n$.

This definition inspires a parameterized version, where the oracle in the computation depends on x.

Definition 5.6. A reduction $f(x) \geq_T g(x)$ is *Schnorr layerwise uniform in the deficiency of* x if there is a Schnorr test (U_n) and a computable sequence of indices e_n such that $\Phi_{e_n}(f(x)) = g(x)$ for all $x \in X \setminus U_n$.

Note the use of x in the oracle below. We will be able to eliminate the use of x in the oracle in Section 5.5.

Theorem 5.7. There exists a $\mathcal{C}^1$, computable function f for which $x \oplus f'(x) \geq_T \emptyset'$ Schnorr-layerwise uniformly in x.

Proof. We follow the same construction as in Theorem 5.2, but with continuous wedge-shaped waveforms.

That is, for each depth a and partition $\mathcal{P} = \{p_0, p_1, \ldots, p_n\}$, we replace our step functions with the piecewise linear functions interpolating:

$$(p_m, 0) \text{ to } \left(p_m + \tfrac{p_{m+1}-p_m}{4^{n+1}}, (-1)^m a\right) \text{ to } \left(p_{m+1} - \tfrac{(p_{m+1}-p_m)}{4^{n+1}}, (-1)^m a\right) \text{ to } (p_{m+1}, 0),$$

for $0 \leq m \leq n-1$. Summing these waveforms in reverse order of depth, we see that convergence $\tilde{g}_n \to g$ is absolutely uniform, and so g inherits continuity from the $\tilde{g}_n$.

Consider the intervals where the waveforms have positive slope, which we will call the *smoothing sets*. On these intervals we are not coding for $\emptyset'$, as we are not controlling the height of the waveform. However, the smoothing sets form a Schnorr test: the smoothing set for a step function on a dyadic mesh of length n have uniformly computable rational sizes bounded by $\frac{2n}{4^{n+1}}$, and we only add one step function for each partition, so we can even throw all the potential smoothing sets we could ever use into a Schnorr test U_n.

Therefore, each Schnorr random x only falls on the smoothed sections of finitely

many waveforms. But given x and a bound on the number of waveforms it lies on, we can compute which smoothing intervals x beongs to, and subtract off the value of the corresponding waveforms from $f'(x)$. Now we have succeeding in coding all but a finite initial segment of $\emptyset'$, whose length we know: but by the padding lemma, if we know any tail end of $\emptyset'$ we can always uniformly recover any initial segment of $\emptyset'$. □

5.4 Coding locally

So far, our theorems only yield $f'(x)$ of some fixed complexity for almost all x. We could also consider varying the complexity requirements on $f'(x)$ with the value of x. Again, we begin with a remark:

Remark 5.8. For computable f, $x' \geq_T f'(x)$ uniformly in all x at which f is differentiable.

Proof. Since f is computable, x' can compute difference quotients on rational intervals closing about x, and find a sequence of such converging appropriately to (i.e. giving a Cauchy name for) $f'(x)$. □

So the strongest complexity we could demand from $f'(x)$ is that $f'(x) \geq x'$. We can upgrade Theorem 5.2 to obtain such f, our only innovation is to code locally as we witness convergences on individual dyadic rational intervals. We recommend the reader consult the construction of Theorem 5.2 before this local construction.

Theorem 5.9. There exists a computable function f for which $f'(x) \geq_T x'$ uniformly in x for almost every x.

Proof. We reference the proof of Theorem 5.2, but code locally on dyadic rational intervals. As before, $g_n(x)$ will be the sum of the first n waveforms enumerated in a

computable procedure, but each waveform we be added on a dyadic rational interval as follows: let $\{\sigma_n\}_{n\in\omega}$ list all finite binary strings in $2^{<\omega}$, which we again identify with the binary expansions of dyadic rationals in $[0,1]$, and consider each $\Phi_e^{\sigma_n}(e)$. When we witness a convergence $\Phi_{|\sigma_n|,e}^{\sigma_n}(e)\downarrow$, this means that $x'(e)=1$ for each $x\in(\sigma_n,\sigma_n{}^\frown 1)$. So, with the same coding and same fineness requirements as in Theorem 5.2, we add a wavefunction of depth 3^{-e} on a partition of the interval $(\sigma_n,\sigma_n{}^\frown 1)$.

Again, with $g_n(x)$ the sum of the first n waveforms (by enumeration), since we choose our partitions small enough we will still obtain $\int_0^x g_n(t)dt$ converging uniformly effectively to $\int_0^x g(t)dt$. But we cannot take $\tilde{g}_n$ the sum of the first n step functions by decreasing depth, since we have added step functions of the same height on densely many dyadic rational intervals. Instead, we set $\tilde{g}_n$ to be the sum of all the step functions of depth down to 3^{-n}.

Now the $\tilde{g}_n$ may not be continuous off of dyadic rationals, but they are still certainly integrable by Lebesgue dominated convergence. So we still have our derivatives:

$$\lim_{h\to 0}\frac{\int_x^{x+h}\tilde{g}_n(t)dt}{h}=\tilde{g}_n(x)$$

almost everywhere by the Lebesgue differentiation theorem. Since we also still have the same bound on the remainder term in our difference quotient analysis, we obtain again that $f'(x)=g(x)$ almost everywhere. □

Note that we must temper any hope to upgrade Theorem 5.7 in this local way. That is, we cannot obtain a Schnorr-layerwise reduction $x\oplus f'(x)\geq_T x'$ for a $\mathcal{C}^1$ computable function f:

Remark 5.10. If f is $\mathcal{C}^1$ and computable, then $\emptyset'\oplus x\geq_T f'(x)$.

Proof. Since f is $\mathcal{C}^1$, for each n there exists a rational mesh $\{x_1, \ldots, x_k\}$ of sufficient fineness so that on each sub-interval $[x_i, x_{i+1}]$, all rational secant lines (and therefore all tangent lines) have slope within 2^{-n} of $\frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i}$. Since f is computable, $\emptyset'$ can find these meshes uniformly in n. From this sequence of meshes, access to x allows us to compute a Cauchy name for $f'(x)$. □

Therefore, for f $\mathcal{C}^1$ and computable, existence of a reduction $x \oplus f'(x) \geq_T x'$ at a point x witnesses that x is GL_1. But there exist Schnorr randoms which are not GL_1 (indeed, there are 1-randoms which are not GL_1).

If we drop all uniformity requirements, the f of Theorem 5.7 is almost sufficient to obtain $f'(x) \geq_T x'$ almost everywhere, since almost every x is GL_1. See Theorem 5.17 below.

A naive attempt to adapt our proof of Theorem 5.7 in a similar way will run into the following problem: although the convergence $\tilde{g}_n \to g$ is uniform, each $\tilde{g}_n$ need not be (and indeed is not) continuous, so we would need not obtain a continuous g.

Although we cannot obtain a Schnorr layerwise reduction $f'(x) \geq_T x'$ for any continuously differentiable f, we can still obtain one for an f which is everywhere differentiable:

Theorem 5.11. There exists a function f which is computable, differentiable everywhere, such that $x \oplus f'(x) \geq_T x'$ Schnorr-layerwise uniformly in x.

Proof. We follow the proof of Theorem 5.7, but we modify our meshes.

In Theorem 5.7, whenever we witness convergence of a functional on some interval, we wish to code on this subinterval by locally adding a waveform. Instead of choosing a fresh uniform mesh of fineness δ satisfying $3^{-a_n}\delta < 2^{-n-1}$ and defining our waveform on this mesh, we ask as another uniform requirement that $3^{-a_n}\delta < 2^{-n-1}l$, where l is the

length of the interval we are coding on. Moreover, we ask that the fineness of the mesh increase quadratically in the distance from each endpoint.

To achieve this last requirement, suppose we are working on an interval (a, b), where $b - a = l$. We refine our mesh by adding the points $(\{a + 1/n : n \in \mathbb{N}\} \cup \{b - \frac{1}{n} : n \in \mathbb{N}\}) \cap (a, b)$. To define the step function, we begin at the centermost intervals, cutting them in half and alternating (continuously, as in Theorem 5.7) between a and $-a$ (so that the integral over each interval of the mesh is 0), and continue alternating out, defining our waveform in this way on the whole open interval.

Our meshes are still computable, and note that the fineness of the mesh at each stage has only been increased. Now if we integrate a waveform so defined from a to $a + 1/n$, note that we miss at most the interval from $a + 1/(n + 1)$ to $a + 1/n$, which is of width $\frac{1}{n} - \frac{1}{n+1} = \frac{1}{n(n+1)} < \frac{1}{n^2}$. More generally integrating from a to $a + h$ we claim we obtain a value which is $\leq 2^{1-i}h^2$. To see this, find n so that $\frac{1}{n+1} < h < \frac{1}{n}$. Then integrating up to h, we obtain a value of at most $2^{-i}\left(\frac{1}{n} - \frac{1}{n+1}\right) = 2^{-i}\frac{1}{n(n+1)} = 2^{-i}\frac{n+1}{n(n+1)^2} \leq 2^{1-i}\frac{1}{(n+1)^2} \leq 2^{1-i}h^2$.

Now consider our analysis of the difference quotients. Let $\tilde{g}_n(x) = \sum_{i=0}^{n} \gamma_i(x)$ where each γ_i is the sum of all the local waveforms coding for the i'th bit of x' on some interval. When x lies in the coding set of a γ_i (that is, in an interval where we are coding for the i'th bit of x'), we have that γ_i is continuous at x, and so $\lim_{h\to 0} \frac{\int_x^{x+h} \gamma_i(t)dt}{h} = \gamma_i(x)$ as desired.

For x that do not lie in the coding set of some γ_i, consider the integral $\int_x^{x+h} \gamma_i(t)dt$. If $x + h$ lies on the support of γ_i, then since x does not lie on the support of γ_i, if $x + h$ lies in a coding set for γ_i then it lies in a coding interval (a dyadic interval where we have witnessed convergence of x' at poisition i) that starts, at worst, at x.

So $\int_x^{x+h} \gamma_i(t)dt \leq 2^{1-i}h^2$. (Note that if $x+h$ does not lie in a coding set for γ_i, then $\int_x^{x+h} \gamma_i(t)dt = 0$). So

$$\lim_{h\to 0} \frac{\int_x^{x+h} \tilde{g}_n(t)dt}{h} = \lim_{h\to 0} \frac{\sum_{i=0}^{n} \int_x^{x+h} \gamma_i(t)dt}{h} \leq \lim_{h\to 0} \frac{4 \cdot h^2}{h} = 0$$

so that $\tilde{g}_n$ is differentiable at x as well.

We have the same bound on the remainder term, so we obatain $f(x)$ differentiable everywhere, as desired. □

We have seen this level of uniformity cannot be improved to u.a.e. for f C_1. But neither can it for f differentiable everywhere, by the following Baire category result:

Theorem 5.12 (Clarkson [10]). If f is differentiable, then for each interval I, either $(f')^{-1}(I)$ is empty, or is of positive measure.

This allows us to extend our nonexistence result to the following:

Theorem 5.13. There does not exist f differentiable such that $f'(x) \geq_T \emptyset'$ uniformly a.e.

Proof. Any uniform reduction $y \geq_T \emptyset'$ fails on densely many intervals of y. By Darboux's theorem, f' has the intermediate value property. We may assume f' is nonconstant, so the values of f' cover a whole interval, of which we may find a subinterval where the reduction $y \geq_T \emptyset'$ fails. By Clarkson's theorem, f' pulls back this interval to a non-null set. □

5.5 Removing the point from the oracle

Our definition of computation *Schnorr-layerwise uniform in* x uses x as an oracle, so it might be reasonable to ask if a reduction of the form $x \oplus f'(x) \geq_T \emptyset'$ can be obtained u.a.e.

Theorem 5.14. There does not exist f differentiable almost everywhere such that $x \oplus f'(x) \geq_T \emptyset'$ uniformly almost everywhere.

Proof. Let Ψ be a functional witnessing the uniform computation of $\emptyset'$. Viewed as a functional on $\mathbb{R} \times \mathbb{R}$, this gives a reduction $y_1 \oplus y_2 \geq \emptyset'$ that is valid on some subset of $\mathbb{R} \times \mathbb{R}$. We write a program i searching for open boxes $I_1 \times I_2$ where Ψ converges to 0 and on which there exists a rational secant line of f over I_1 whose slope is contained in I_2. When such a point is found, i converges to 1. Then at some point within I_1 the value of $f'(x)$ lands in I_2. Restricting f to I_1, the preimage of I_2 under f' is then a nonempty subset of I_1, and hence of positive measure. So on a positive measure set we have succeeded in diagonalizing against the reduction $x \oplus f'(x) \geq \emptyset'$. □

In fact, our use of x in our Schnorr-layerwise existence results is somewhat superficial. Our reductions use the deficiency of x to bound the number of smoothing sets x lies in, but x itself is only used in order to reconstruct the waveforms where coding on x failed—in order to subtract away the smoothing waveforms and recover the coding of the remaining bits of our jump. We can actually avoid the need for this subtraction entirely by modifying our waveforms.

Both of the following theorems are obtained by modifying previous constructions:

Theorem 5.15. There exists a computable function f, differentiable everywhere, for which $f'(x) \geq x'$ Schnorr-layerwise uniformly in the deficiency of x.

Theorem 5.16. There exists a $\mathcal{C}^1$, computable function f for which $f'(x) \geq_T \emptyset'$ Schnorr-layerwise uniformly in the deficiency of x.

Proof. We follow the same proofs as before, but instead of smoothing by connecting our piecewise constant segments with linear interpolation, we use a Cantor function (Devil's

staircase). Note that in the previous verifications, we only use that the smoothing pieces are uniformly computable and continuous, so we obtain the same properties on f as before.

Observe that in the step-by-step construction of the Cantor function, off of the null Cantor set, at each step n we define values of the Cantor function which are uniformly away from 0 in n: that is, if we only consider the part of the domain of the Cantor function defined up to stage n in the construction, any value of the Cantor function so far defined has a finite ternary expansion of computably bounded length. In particular, when values of the Cantor function are added to our coding, we have an upper bound on the length of the initial segment of bits in the coding that they are able to affect.

Now we rewrite our Schnorr test to cover our tracks. Interweaving the nested pieces of the Cantor sets within our Schnorr test, the deficiency of x alone determines how many of the smoothing sets could have contained x and which piece of the smoothing set it is in: not only do we know which bits of the coding of $f'(x)$ were missed, but also which bits could possibly have been lost in the smoothing.

So we can ignore these bits and recover all the coding uniformly off each layer of the Schnorr test again by the padding lemma. □

This construction now also allows us to obtain our $\mathcal{C}^1$ function f with $f'(x) \geq_T x'$ non-uniformly almost-everywhere, as promised before:

Theorem 5.17. There exists a $\mathcal{C}^1$, computable function f for which $f'(x) \geq_T x'$ almost everywhere (non-uniformly).

Proof. Almost every x is GL_1, so for almost every x, $x \oplus \emptyset' \geq_T x'$. We can modify the construction of f in Theorem 5.7 to code x locally into the sequence of even ternary bits

(that is, the sequence obtained by looking at every other bit in the ternary expansion of $f'(x)$). Note we can always do this much local coding continuously, because there are no late entries in x: coding for a bit in x, we add only 1 linear function of height $a = 3^{-2n}$. So $x \oplus f'(x) \geq_T x \oplus \emptyset' \geq_T x'$ almost everywhere. We see in Theorem 5.16 that the use of x is not necessary, and so we can obtain f $\mathcal{C}^1$ with $f'(x) \geq_T x'$ almost everywhere. $\square$

5.6 Summary

Although the individual results and the techniques used are of interest in themselves, taken together they form a striking picture of a three-way interaction between analytic conditions, computational complexity, and paradigms of computation. We considered questions of the form: "does there exist a computable real-valued function f with a given smoothness property such that, with a given level of computational uniformity, $f'(x) \geq_T \emptyset'$, or $f'(x) \geq_T x'$?"

The smoothness classes we considered were the class of almost-everywhere differentiable functions, the class of functions which are differentiable everywhere, and finally the class $\mathcal{C}^1$ of smooth functions. The notions of computational uniformity we considered were reductions given uniformly almost everywhere, and reductions which were Schnorr-laywerwise uniform.

The preceeding results, together with some of the observations, answer all the questions raised by combining these notions. By Theorem 5.9, there exists f computable such that $f'(x) \geq_T x'$ uniformly almost everywhere. Such an f cannot be made to be differentiable everywhere by Theorem 5.13. And by Remark 5.1 on the mean value theorem, such a reduction cannot be improved to a uniform reduction.

However, by Theorem 5.15, there exists f computable and differentiable everywhere such that $f'(x) \geq_T x'$ Schnorr-layerwise in the deficiency of x. This cannot be improved to a uniformly almost-everywhere reduction by our remark on Clarkson's theorem.

By Theorem 5.16, there exists f computable and $\mathcal{C}^1$ such that $f'(x) \geq_T \emptyset'$ Schnorr-layerwise in the deficiency of x. This cannot be improved to $f\ \mathcal{C}^2$ by the result of Pour-El and Richards, cannot be improved to uniformly almost everywhere by Theorem 5.13, and cannot be improved to $\geq_T x'$ by the remark on non-GL_1 Schnorr-randoms.

Another way to understand how these results fit together is to consider the "best" reduction achievable for a computable function with a given analytic property, where we consider uniform almost everywhere (u.a.e.) "better" than Schnorr-layerwise in the deficiency of x better than (non-uniform) almost-everywhere reductions (a.e.): then for f differentiable almost everywhere, the best we can do for either $f'(x) \geq_T \emptyset'$ or $f'(x) \geq_T x'$ is a u.a.e. reduction. For f differentiable everywhere, the best we can do for either $f'(x) \geq_T \emptyset'$ or $f'(x) \geq_T x'$ is a layerwise reduction. For $f\ \mathcal{C}^1$, the best we can do for $f'(x) \geq_T \emptyset'$ is a layerwise reduction, and the best we can do for $f'(x) \geq_T x'$ is an a.e. reduction.

In what we've said above, note that Schnorr-layerwise and u.a.e. reductions are incomparable: Schnorr layerwise reductions are not uniform on any particular set of full measure, but at each Schnorr-random x we eventually obtain a reduction for $f'(x)$, whereas u.a.e. reductions may entirely fail at some Schnorr randoms. However, note that in each case where we obtain a u.a.e reduction, we separately obtain a Schnorr layerwise reduction by virtue of our stronger results constructing f differentiable everywhere, and hence certainly differentiable almost everywhere. In this sense, none of these "best" types of reductions can be improved to a stronger uniformity notion.

To go from f differentiable almost everywhere to f differentiable everywhere, we must sacrifice uniformity in the complexity of f' for some sort of layerwise uniformity. To go from f differentiable everywhere to f $\mathcal{C}^1$, we must further sacrifice local complexity $f'(x) \geq_T x'$ for a lower fixed complexity $f'(x) \geq_T \emptyset'$.

It remains open what happens when f is required to be twice-differentiable, but not twice continuously differentiable. We know that when a computable f is $\mathcal{C}^2$, f' is computable, so it is not possible to obtain $f'(x) \geq_T \emptyset'$ for any positive measure set of x. But we wonder what happens to Theorem 5.16 in the weaker case where f is required only to be twice differentiable, not $\mathcal{C}^2$.

Bibliography

[1] Uri Andrews, Hristo A. Ganchev, Rutger Kuyper, Steffen Lempp, Joseph S. Miller, Alexandra A. Soskova, and Mariya I. Soskova, *On cototality and the skip operator in the enumeration degrees*, Submitted.

[2] Oleg V. Belegradek, *On algebraically closed groups*, Algebra i Logika **13** (1974), no. 3, 813–816.

[3] Laurent Bienvenu and Christopher P. Porter, *Strong difference randomness*, Informal presentation, Computability in Europe 2013.

[4] William W. Boone, *Certain simple, unsolvable problems of group theory I-IV*, Journal of Symbolic Logic **22** (1957), no. 4, 372–373.

[5] Vasco Brattka, Joseph S. Miller, and André Nies, *Randomness and differentiability*, Trans. Amer. Math. Soc. **368** (2016), no. 1, 581–605.

[6] Douglas Cenzer, Ali Dashti, and Jonathan L. F. King, *Computable symbolic dynamics*, Mathematical Logic Quarterly **54** (2008), no. 5, 460–469.

[7] Douglas Cenzer, Ali Dashti, Ferit Toska, and Sebastian Wyman, *Computability of countable subshifts*, Programs, Proofs, Processes: 6th Conference on Computability in Europe (Berlin, Heidelberg), CiE'10, Springer-Verlag, 2010, pp. 88–97.

[8] ———, *Computability of countable subshifts in one dimension*, Theory of Computing Systems **51** (2012), no. 3, 352–371.

[9] Peter Cholak, Noam Greenberg, and Joseph S. Miller, *Uniform almost everywhere domination*, The Journal of Symbolic Logic **71** (2006), no. 3, 1057–1072.

[10] J. A. Clarkson, *A property of derivatives*, Bull. Amer. Math. Soc. **53** (1947), no. 2, 124–125.

[11] Karel de Leeuw, Edward F. Moore, Claude F. Shannon, and Norman Shapiro, *Computability by probabilistic machines*, Annals of Mathematical Studies, no. 34, pp. 183–212, Princeton University Press, 1956.

[12] Jean-Charles Delvenne, Petr Kůrka, and Vincent Blondel, *Decidability and universality in symbolic dynamical systems*, Fundamenta Informaticae **74** (2006), no. 4, 469–490.

[13] Osvald Demuth, *The differentiability of constructive functions of weakly bounded variation on pseudo numbers*, Comment. Math. Univ. Carolin. **16** (1975), no. 3, 583–599, Russian.

[14] David Diamondstone, Noam Greenberg, and Daniel D. Turetsky, *Inherent enumerability of strong jump-traceability*, Transactions of the American Mathematical Society **367** (2015), no. 3, 1771–1796.

[15] Rod Downey and Noam Greenberg, *Strong jump-traceability II: K-triviality*, Israel Journal of Mathematics **191** (2012), no. 2, 647–666.

[16] Rod Downey, Andre Nies, Rebecca Weber, and Liang Yu, *Lowness and Π^0_2 nullsets*, Journal of Symbolic Logic **71** (2006), no. 3, 1044–1052.

[17] Rodney G. Downey and Denis R. Hirschfeldt, *Algorithmic randomness and complexity*, Theory and Applications of Computability, Springer, 2010.

[18] Bruno Durand, Leonid A. Levin, and Alexander Shen, *Complex tilings*, J. Symbolic Logic **73** (2008), no. 2, 593–613.

[19] Lawrence Feiner, *The strong homogeneity conjecture*, J. Symbolic Logic **35** (1970), no. 3, 375–377.

[20] Santiago Figueira, Denis R. Hirschfeldt, Joseph S. Miller, Keng Meng Ng, and André Nies, *Counting the changes of random Δ^0_2 sets*, Journal of Logic and Computation **25** (2015), no. 4, 1073–1089.

[21] Santiago Figueira, André Nies, and Frank Stephan, *Lowness properties and approximations of the jump*, Annals of Pure and Applied Logic **152** (2008), no. 1, 51–66.

[22] Johanna N. Franklin and Keng Meng Ng, *Difference randomness*, Proceedings of the American Mathematical Society **139** (2011), no. 1, 345–360.

[23] Péter Gács, *Every sequence is reducible to a random one*, Information and Control **70** (1986), no. 2–3, 186–192.

[24] Robin O. Gandy, Georg Kreisel, and William W. Tait, *Set existence*, Bulletin de l'Académie Polonaise des Sciences, Série des Sciences Mathématiques, Astronomiques et Physiques **8** (1960), 577–582.

[25] Noam Greenberg, Denis R. Hirschfeldt, and André Nies, *Characterizing the strongly*

jump-traceable sets via randomness, Advances in Mathematics **231** (2012), no. 3–4, 2252–2293.

[26] Noam Greenberg and André Nies, *Benign cost functions and lowness properties*, The Journal of Symbolic Logic **76** (2011), no. 1, 289–312.

[27] Lance Gutteridge, *Some results on enumeration reducibility*, Ph.D. thesis, Simon Fraser University, 1971.

[28] Graham Higman and Elizabeth Scott, *Existentially closed groups*, London Mathematical Society Monographs, vol. 3, Oxford University Press, New York, 1988.

[29] Mike Hochman and Pascal Vanier, *A note on Turing degree spectra of minimal subshifts*, CoRR **abs/1408.6487** (2014).

[30] Mathieu Hoyrup and Cristobal Rojas, *An application of Martin-Löf randomness to effective probability theory*, Mathematical Theory and Computational Practice. CiE 2009, Lecture Notes in Computer Science, vol. 5635, CiE, Springer Berlin, Heidelberg, 2009, pp. 260–269.

[31] ———, *Applications of effective probability theory to Martin-Löf randomness*, 36th International Colloquium on Automata, Languages and Programming, Lecture Notes in Computer Science, vol. 5555, ICALP, Springer Berlin, Heidelberg, 2009, pp. 549–561.

[32] Emmanuel Jeandel, *Enumeration in closure spaces with applications to algebra*, CoRR **abs/1505.07578** (2015).

[33] Emmanuel Jeandel and Pascal Vanier, *Turing degrees of multidimensional SFTs*, CoRR **abs/1108.1012** (2011).

[34] Carl G. Jockusch and Robert I. Soare, *Degrees of members of* Π^0_1 *classes*, Pacific Journal of Mathematics **40** (1972), no. 3, 605–616.

[35] Takayuki Kihara and Arno Pauly, *Point degree spectra of represented spaces*, CoRR **abs/1405.6866** (2014).

[36] Bjørn Kjos-Hanssen, Joseph S. Miller, and Reed Solomon, *Lowness notions, measure and domination*, Journal of the London Mathematical Society **85** (2012), no. 3, 869–888.

[37] Antonín Kučera, *Measure,* Π^0_1 *classes and complete extensions of PA*, Recursion Theory Week, Lecture Notes in Mathematics, no. 1141, Springer-Verlag, 1985, pp. 245–259.

[38] Stuart Alan Kurtz, *Randomness and genericity in the degrees of unsolvability*, Ph.D. thesis, University of Illinois, 1981.

[39] Angus Macintyre, *Omitting quantifier-free types in generic structures*, J. Symbolic Logic **37** (1972), no. 3, 512–520.

[40] Joseph S. Miller, *Degrees of unsolvability of continuous functions*, J. Symbolic Logic **69** (2004), no. 2, 555–584.

[41] Joseph S. Miller and André Nies, *Randomness and computability: Open questions*, Bulletin of Symbolic Logic **12**, no. 3, 390–410.

[42] Joseph S. Miller and Mariya I. Soskova, *Density of the cototal enumeration degrees*, To appear in Ann. Pure Appl. Logic.

[43] Kenshi Miyabe, *Characterization of Kurtz randomness by a differentiation theorem*, Theory of Computing Systems **52** (2013), no. 1, 113–132.

[44] Jeanleah Mohrherr, *A refinement of low n and high n for the r.e. degrees*, Z. Math. Logik Grundlag. Math. **32** (1986), no. 1, 5–12.

[45] Antonio Montalbán, *Computable structure theory: Part 1*, Unpublished manuscript dated 2016.

[46] André Nies, *Reals which compute little*, Logic Colloquium '02, Lecture Notes in Logic, Springer–Verlag, 2002, pp. 260–274.

[47] ———, *Lowness properties and randomness*, Advances in Mathematics **197** (2005), no. 1, 274–305.

[48] ———, *Eliminating concepts*, Computational Prospects of Infinity. Part II. Presented talks (Hackensack, NJ), Lecture Notes Series. Institute for Mathematical Sciences. National University of Singapore, vol. 15, World Sci. Publ., 2008, pp. 225–247.

[49] ———, *Computability and randomness*, Oxford University Press, Inc., New York, NY, USA, 2009.

[50] Noopur Pathak, Cristbal Rojas, and Stephen G. Simpson, *Schnorr randomness and the Lebesgue differentiation theorem*, Proceedings of the American Mathematical Society **142** (2014), no. 1, 335–349.

[51] Christopher P. Porter, *Mathematical and philosophical perspectives on algorithmic randomness*, Ph.D. thesis, University of Notre-Dame, 2012.

[52] Marian B. Pour-El and J. Ian Richards, *Computability in analysis and physics*, Springer-Verlag, Berlin, 1989.

[53] Gwenaël Richomme, Kalle Saari, and Luca Q. Zamboni, *Abelian complexity of minimal subshifts*, J. London Math. Soc **83** (2011), no. 2, 79–95.

[54] Linda Jean Richter, *Degrees of unsolvability of models*, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1977.

[55] ———, *Degrees of structures*, J. Symbolic Logic **46** (1981), no. 4, 723–731.

[56] Hartley Rogers Jr., *Theory of recursive functions and effective computability*, McGraw–Hill Book Company, New York, 1967.

[57] Jason Rute, *Topics in algorithmic randomness and computable analysis*, Ph.D. thesis, Carnegie Mellon University, 2013.

[58] Gerald E. Sacks, *Degrees of unsolvability*, Annals of Mathematics Studies, no. 55, Princeton University Press, 1963.

[59] Alan L. Selman, *Arithmetical reducibilities I*, Mathematical Logic Quarterly **17** (1971), no. 1, 335–350.

[60] Stephen G. Simpson, *Almost everywhere domination and superhighness*, Math. Logic Quart. **53** (2007), no. 4–5, 462–482.

[61] ———, *Medvedev degrees of two-dimensional subshifts of finite type*, Ergodic Theory and Dynamical Systems **34** (2012), no. 2, 679–688.

[62] Robert I. Soare, *Recursively enumerable sets and degrees: A study of computable functions and computably generated sets*, Perspectives in Mathematical Logic, Springer, Berlin, 1987.

[63] Robert M. Solovay, *Draft of a paper (or series of papers) on Chaitin's work*, Unpublished manuscript dated May 1975.

[64] Frank Stephan, *Martin-Löf random and PA-complete sets*, Technical Report 58 (Heidelberg), Matematisches Institut, Universität Heidelberg, 2002.

[65] Yongge Wang, *Randomness and complexity*, Ph.D. thesis, University of Heidelberg, 1996.

[66] Klaus Weihrauch, *Computable analysis: An introduction*, Springer-Verlag New York, Secaucus, NJ, USA, 2000.

[67] Boris Ya. Solon, *Total and co-total enumeration degrees (Russian)*, Izv. Vyssh. Uchebn. Zaved. Mat. **49** (2005), no. 9, 56–64.

[68] Boris Ya. Solon, *Co-total enumeration degrees*, Proceedings of the Second Conference on Computability in Europe: Logical Approaches to Computational Barriers (Berlin, Heidelberg), CiE'06, Springer-Verlag, 2006, pp. 538–545.

[69] Martin Ziegler, *Algebraisch abgeschlossene Gruppen*, Word Problems II (S.I. Adian, W.W. Boone, and G. Higman, eds.), Studies in Logic and the Foundations of Mathematics, vol. 95, North-Holland, 1980, pp. 449–576.